

## **Anexo IX – Código do modelo de simulação em SIMAN**

```

Model statements for module: Station 5
4$ STATION, Estacoes Y Set.Station
1308$ ASSIGN: A Pos=MEMIDX(Estacoes Y Set.Station,M)
1310$ DELAY: 0.0,,VA:NEXT(159$)
Model statements for module: Decide 255
159$ BRANCH, 1:
If,A Ultrapassa==0 .AND. V Destino ( A PtAtual,1 + (A
Sentido==0)) == 0,1311$,Yes:
Else,1312$,Yes
1311$ ASSIGN: Nao esta fazendo ultrapassagem ?.NumberOut True=Nao esta
fazendo ultrapassagem ?.NumberOut True + 1
:NEXT(200$)
1312$ ASSIGN: Nao esta fazendo ultrapassagem ?.NumberOut False=
Nao esta fazendo ultrapassagem ?.NumberOut False +
1:NEXT(166$)
Model statements for module: Decide 296
200$ BRANCH, 1:
If,Temp3 == 8300,1313$,Yes:
Else,1314$,Yes
1313$ ASSIGN: Patio de entroncamento ?.NumberOut True=Patio de
entroncamento ?.NumberOut True + 1:NEXT(150$)
1314$ ASSIGN: Patio de entroncamento ?.NumberOut False=Patio de
entroncamento ?.NumberOut False + 1:NEXT(223$)
Model statements for module: Assign 283
150$ ASSIGN: IndColuna=(1*(A Tipo<4) + 2*(A Tipo==4) + 3*(A Tipo==5))
+ 3*(A Sentido==0):
A Dist=A Dist + V Trecho (A Trecho, 10):
StatusPP(A PtAtual,Indcoluna)=1:
NumPP(A PtAtual,1 + (A Sentido==0))=A IDTrem:NEXT(639$)
Model statements for module: Decide 481
639$ BRANCH, 1:
If,A Tipo < 4 .AND. A RetornoH <> 0 .AND. A SouHelper ==
0,631$,Yes:
If,A RetornoH <> 0 .AND. A SouHelper == 0,640$,Yes:
Else,92$,Yes
Model statements for module: Decide 479
631$ BRANCH, 1:
If,A PtAtual==136 .AND. A Sentido == 1,632$,Yes:
If,A PtAtual==7 .AND. (A RetornoH==143 .OR. A
RetornoH==144),636$,Yes:
If,A PtAtual==3 .AND. A RetornoH==1,643$,Yes:
If,A PtAtual==21 .AND. A RetornoH==7,640$,Yes:
If,A PtAtual==33,646$,Yes:
If,A PtAtual==179,649$,Yes:
Else,92$,Yes
Model statements for module: Process 3
632$ ASSIGN: 1 Desanexa helpers em FMS.NumberIn=1 Desanexa helpers em
FMS.NumberIn + 1:
1 Desanexa helpers em FMS.WIP=1 Desanexa helpers em
FMS.WIP+1
1322$ QUEUE, 1 Desanexa helpers em FMS.Queue
1321$ SEIZE, 2,VA:
AnimaH1,1:NEXT(1320$)
1320$ DELAY: E Helpers(2,1),,VA
1319$ RELEASE: AnimaH1,1
1367$ ASSIGN: 1 Desanexa helpers em FMS.NumberOut=1 Desanexa helpers
em FMS.NumberOut + 1:
1 Desanexa helpers em FMS.WIP=1 Desanexa helpers em
FMS.WIP-1:NEXT(633$)
Model statements for module: Separate 64
633$ DUPLICATE, 100 - 50:
1,1372$,50:NEXT(1371$)
1371$ ASSIGN: Separate 64.NumberOut Orig=Separate 64.NumberOut Orig +
1:NEXT(652$)

```

```

1372$      ASSIGN:      Separate 64.NumberOut Dup=Separate 64.NumberOut Dup +
1:NEXT(634$)
      Model statements for module: Assign 560
652$      ASSIGN:      A NLocos=0:
      A RetornoH=0:NEXT(92$)
      Model statements for module: Assign 547
634$      ASSIGN:      Indice=1:NEXT(635$)
      Model statements for module: Route 31
635$      ROUTE:      0.0000000000000000,Retorno de helpers
      Model statements for module: Process 6
636$      ASSIGN:      2 Desanexa helpers em FJC.NumberIn=2 Desanexa helpers em
FJC.NumberIn + 1:
      2 Desanexa helpers em FJC.WIP=2 Desanexa helpers em
FJC.WIP+1
1376$      QUEUE,      2 Desanexa helpers em FJC.Queue
1375$      SEIZE,      2,VA:
      AnimaH2,1:NEXT(1374$)
1374$      DELAY:      E Helpers(2,1),,VA
1373$      RELEASE:    AnimaH2,1
1421$      ASSIGN:      2 Desanexa helpers em FJC.NumberOut=2 Desanexa helpers
em FJC.NumberOut + 1:
      2 Desanexa helpers em FJC.WIP=2 Desanexa helpers em
FJC.WIP-1:NEXT(637$)
      Model statements for module: Separate 65
637$      DUPLICATE,    100 - 50:
      1,1426$,50:NEXT(1425$)
1425$      ASSIGN:      Separate 65.NumberOut Orig=Separate 65.NumberOut Orig +
1:NEXT(652$)
1426$      ASSIGN:      Separate 65.NumberOut Dup=Separate 65.NumberOut Dup +
1:NEXT(638$)
      Model statements for module: Assign 553
638$      ASSIGN:      Indice=2:NEXT(635$)
      Model statements for module: Process 10
643$      ASSIGN:      3 Desanexa helpers em FLO.NumberIn=3 Desanexa helpers em
FLO.NumberIn + 1:
      3 Desanexa helpers em FLO.WIP=3 Desanexa helpers em
FLO.WIP+1
1430$      QUEUE,      3 Desanexa helpers em FLO.Queue
1429$      SEIZE,      2,VA:
      AnimaH3,1:NEXT(1428$)
1428$      DELAY:      E Helpers(2,1),,VA
1427$      RELEASE:    AnimaH3,1
1475$      ASSIGN:      3 Desanexa helpers em FLO.NumberOut=3 Desanexa helpers
em FLO.NumberOut + 1:
      3 Desanexa helpers em FLO.WIP=3 Desanexa helpers em
FLO.WIP-1:NEXT(644$)
      Model statements for module: Separate 67
644$      DUPLICATE,    100 - 50:
      1,1480$,50:NEXT(1479$)
1479$      ASSIGN:      Separate 67.NumberOut Orig=Separate 67.NumberOut Orig +
1:NEXT(652$)
1480$      ASSIGN:      Separate 67.NumberOut Dup=Separate 67.NumberOut Dup +
1:NEXT(645$)
      Model statements for module: Assign 555
645$      ASSIGN:      Indice=3:NEXT(635$)
      Model statements for module: Process 9
640$      ASSIGN:      5 Desanexa helpers em FOJ.NumberIn=5 Desanexa helpers em
FOJ.NumberIn + 1:
      5 Desanexa helpers em FOJ.WIP=5 Desanexa helpers em
FOJ.WIP+1
1484$      QUEUE,      5 Desanexa helpers em FOJ.Queue
1483$      SEIZE,      2,VA:
      AnimaH5,1:NEXT(1482$)
1482$      DELAY:      E Helpers(2,1),,VA
1481$      RELEASE:    AnimaH5,1

```

```

1529$      ASSIGN:          5 Desanexa helpers em FOJ.NumberOut=5 Desanexa helpers
em FOJ.NumberOut + 1:
                                5 Desanexa helpers em FOJ.WIP=5 Desanexa helpers em
FOJ.WIP-1:NEXT(641$)
    Model statements for module: Separate 66
641$      DUPLICATE,      100 - 50:
                                1,1534$,50:NEXT(1533$)
1533$      ASSIGN:          Separate 66.NumberOut Orig=Separate 66.NumberOut Orig +
1:NEXT(652$)
1534$      ASSIGN:          Separate 66.NumberOut Dup=Separate 66.NumberOut Dup +
1:NEXT(642$)
    Model statements for module: Assign 554
642$      ASSIGN:          Indice=5:NEXT(635$)
    Model statements for module: Process 12
646$      ASSIGN:          6 Desanexa helpers em FHA.NumberIn=6 Desanexa helpers em
FHA.NumberIn + 1:
                                6 Desanexa helpers em FHA.WIP=6 Desanexa helpers em
FHA.WIP+1
1538$      QUEUE,          6 Desanexa helpers em FHA.Queue
1537$      SEIZE,          2,VA:
                                AnimaH6,1:NEXT(1536$)
1536$      DELAY:          E Helpers(2,1),,VA
1535$      RELEASE:       AnimaH6,1
1583$      ASSIGN:          6 Desanexa helpers em FHA.NumberOut=6 Desanexa helpers
em FHA.NumberOut + 1:
                                6 Desanexa helpers em FHA.WIP=6 Desanexa helpers em
FHA.WIP-1:NEXT(647$)
    Model statements for module: Separate 68
647$      DUPLICATE,      100 - 50:
                                1,1588$,50:NEXT(1587$)
1587$      ASSIGN:          Separate 68.NumberOut Orig=Separate 68.NumberOut Orig +
1:NEXT(652$)
1588$      ASSIGN:          Separate 68.NumberOut Dup=Separate 68.NumberOut Dup +
1:NEXT(648$)
    Model statements for module: Assign 556
648$      ASSIGN:          Indice=6:NEXT(635$)
    Model statements for module: Process 14
649$      ASSIGN:          7 Desanexa helpers em ICG.NumberIn=7 Desanexa helpers em
ICG.NumberIn + 1:
                                7 Desanexa helpers em ICG.WIP=7 Desanexa helpers em
ICG.WIP+1
1592$      QUEUE,          7 Desanexa helpers em ICG.Queue
1591$      SEIZE,          2,VA:
                                AnimaH7,1:NEXT(1590$)
1590$      DELAY:          E Helpers(2,1),,VA
1589$      RELEASE:       AnimaH7,1
1637$      ASSIGN:          7 Desanexa helpers em ICG.NumberOut=7 Desanexa helpers
em ICG.NumberOut + 1:
                                7 Desanexa helpers em ICG.WIP=7 Desanexa helpers em
ICG.WIP-1:NEXT(650$)
    Model statements for module: Separate 69
650$      DUPLICATE,      100 - 50:
                                1,1642$,50:NEXT(1641$)
1641$      ASSIGN:          Separate 69.NumberOut Orig=Separate 69.NumberOut Orig +
1:NEXT(652$)
1642$      ASSIGN:          Separate 69.NumberOut Dup=Separate 69.NumberOut Dup +
1:NEXT(651$)
    Model statements for module: Assign 559
651$      ASSIGN:          Indice=7:NEXT(635$)
    Model statements for module: Decide 107
92$       BRANCH,          1:
                                If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),1643$,Yes:
                                Else,1644$,Yes

```

```

1643$      ASSIGN:      Ponto de checagem 4.NumberOut True=Ponto de checagem
4.NumberOut True + 1:NEXT(93$)
1644$      ASSIGN:      Ponto de checagem 4.NumberOut False=Ponto de checagem
4.NumberOut False + 1:NEXT(361$)
      Model statements for module:  Separate 9
93$        DUPLICATE,      100 - 50:
                        1,1647$,50:NEXT(1646$)
1646$      ASSIGN:      Separate 9.NumberOut Orig=Separate 9.NumberOut Orig +
1:NEXT(339$)
1647$      ASSIGN:      Separate 9.NumberOut Dup=Separate 9.NumberOut Dup +
1:NEXT(361$)
      Model statements for module:  Assign 707
339$      ASSIGN:      sh18=A PtAtual:
                        sh19=A TmpTrecho:
                        sh20=A Restricao:
                        sh21=indice:
                        sh22=indice2:NEXT(94$)
      Model statements for module:  Hold 16
94$        QUEUE,          Hold ENTRA PATIO.Queue:DETACH
      Model statements for module:  Assign 723
361$      ASSIGN:      Temp1=1+(A TmpSentido==0):NEXT(17$)
      Model statements for module:  Separate 4
17$        DUPLICATE,      100 - 50:
                        1,1650$,50:NEXT(1649$)
1649$      ASSIGN:      Cria entidade verificadora.NumberOut Orig=Cria entidade
verificadora.NumberOut Orig + 1:NEXT(112$)
1650$      ASSIGN:      Cria entidade verificadora.NumberOut Dup=Cria entidade
verificadora.NumberOut Dup + 1:NEXT(90$)
      Model statements for module:  Assign 141
112$      ASSIGN:      A Tmp=TNOW:NEXT(18$)
      Model statements for module:  Hold 10
18$        QUEUE,          E Filas Patios ( A PtAtual,1 + (A Sentido==0))
      WAIT:              A IDtrem,1:NEXT(268$)
      Model statements for module:  Assign 508
268$      ASSIGN:      StatusPP(A PtAtual,Indcoluna)=0:
                        NumPP(A PtAtual,1 + (A Sentido==0))=0:NEXT(160$)
      Model statements for module:  Decide 256
160$      BRANCH,          1:
                        If,A Ultrapassa==0 .AND. Temp3<>8300,1651$,Yes:
                        Else,1652$,Yes
1651$      ASSIGN:      Nao esta fazendo ultrapassagem2 ?.NumberOut True=
                        Nao esta fazendo ultrapassagem2 ?.NumberOut True +
1:NEXT(222$)
1652$      ASSIGN:      Nao esta fazendo ultrapassagem2 ?.NumberOut False=
                        Nao esta fazendo ultrapassagem2 ?.NumberOut False +
1:NEXT(264$)
      Model statements for module:  Decide 352
222$      BRANCH,          1:
                        If,NR(ESTA_R ( A PtAtual,1 + (A
Sentido==0)))==0,1653$,Yes:
                        Else,1654$,Yes
1653$      ASSIGN:      Decide 352.NumberOut True=Decide 352.NumberOut True +
1:NEXT(271$)
1654$      ASSIGN:      Decide 352.NumberOut False=Decide 352.NumberOut False +
1:NEXT(46$)
      Model statements for module:  Dispose 64
271$      ASSIGN:      Dispose 64.NumberOut=Dispose 64.NumberOut + 1
1655$      DISPOSE:      Yes
      Model statements for module:  Release 8
46$        RELEASE:      ESTA_R ( A PtAtual,1 + (A Sentido==0)),1:NEXT(264$)
      Model statements for module:  Decide 431
264$      BRANCH,          1:
                        If, TNOW > (V Warmup * 1440),1656$,Yes:
                        Else,1657$,Yes

```

```

1656$      ASSIGN:      Warmup do log de movimentacao.NumberOut True=Warmup do
log de movimentacao.NumberOut True + 1
                        :NEXT(350$)
1657$      ASSIGN:      Warmup do log de movimentacao.NumberOut False=Warmup do
log de movimentacao.NumberOut False + 1
                        :NEXT(21$)
      Model statements for module:  Decide 627
350$      BRANCH,      1:
                        If,V Gravar em texto == 1,1658$,Yes:
                        Else,1659$,Yes
1658$      ASSIGN:      Decide 627.NumberOut True=Decide 627.NumberOut True +
1:NEXT(356$)
1659$      ASSIGN:      Decide 627.NumberOut False=Decide 627.NumberOut False +
1:NEXT(148$)
      Especificação do arquivo de LOG:
      1 - Identificador único do trem (A IDTrem)
      2 - Pátio (A PtAtual)
      3 - Momento da entrada no pátio (A Tmp)
      4 - Momento da saída do trem (TNOW)
      5 - Número do trem (A NTrem)
      6 - Horário do trem (A Horario)
      7 - Tipo do trem (A Tipo)
      8 - Gráfico onde entra esta informação (E Patios (A PtAtual,9))
      Model statements for module:  ReadWrite 154
356$      WRITE,      Log de Movimentacao:
                        Dia,
                        A IDtrem,
                        N Trem,
                        A Horario,
                        A PtAtual,
                        A Tmp,
                        TNOW,
                        A Tipo,
                        A Trecho,
                        A Sentido,
                        E Patios (A PtAtual,9),
                        E Patios (A PtAtual,10),
                        A PtDestino,
                        A Produto,
                        A Cliente,
                        0:NEXT(21$)
      Model statements for module:  Dispose 15
21$      ASSIGN:      Dispose 15.NumberOut=Dispose 15.NumberOut + 1
1660$      DISPOSE:      Yes
      Especificação do arquivo de LOG:
      1 - Identificador único do trem (A IDTrem)
      2 - Pátio (A PtAtual)
      3 - Momento da entrada no pátio (A Tmp)
      4 - Momento da saída do trem (TNOW)
      5 - Número do trem (A NTrem)
      6 - Horário do trem (A Horario)
      7 - Tipo do trem (A Tipo)
      8 - Gráfico onde entra esta informação (E Patios (A PtAtual,9))
      Model statements for module:  ReadWrite 38
148$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeMovimentacao):
                        Dia,
                        A IDtrem,
                        N Trem,
                        A Horario,
                        A PtAtual,
                        A Tmp,
                        TNOW,
                        A Tipo,
                        A Trecho,
                        A Sentido,

```

```

E Patios (A PtAtual,9),
E Patios (A PtAtual,10),
A PtDestino,
A Produto,
A Cliente,
0:NEXT(21$)
Model statements for module: Decide 106
90$      BRANCH,      1:
          If,A Tmp==8123,1661$,Yes:
          Else,1662$,Yes
1661$      ASSIGN:      Trem entrou agora na malha ?.NumberOut True=Trem entrou
agora na malha ?.NumberOut True + 1
          :NEXT(91$)
1662$      ASSIGN:      Trem entrou agora na malha ?.NumberOut False=Trem entrou
agora na malha ?.NumberOut False + 1
          :NEXT(366$)
Model statements for module: Assign 131
91$      ASSIGN:      A Tmp=0:
          Indice2=1:
          Indice=0:NEXT(247$)
Model statements for module: Decide 407
247$      BRANCH,      1:
          If,E Patios (A PtAtual,10)==12,1663$,Yes:
          Else,1664$,Yes
1663$      ASSIGN:      Trem entrando no trecho de FVR ?.NumberOut True=Trem
entrando no trecho de FVR ?.NumberOut True + 1
          :NEXT(248$)
1664$      ASSIGN:      Trem entrando no trecho de FVR ?.NumberOut False=
Trem entrando no trecho de FVR ?.NumberOut False +
1:NEXT(56$)
Model statements for module: Assign 457
248$      ASSIGN:      V ContagemFVR=V ContagemFVR + 1:
          A FVR=1:NEXT(56$)
Model statements for module: Decide 79
56$      BRANCH,      1:
          If,Indice > Indice2 .AND. A Restricao == 1,1665$,Yes:
          Else,1666$,Yes
1665$      ASSIGN:      Este trem continuara em frente ?.NumberOut True=Este
trem continuara em frente ?.NumberOut True + 1
          :NEXT(78$)
1666$      ASSIGN:      Este trem continuara em frente ?.NumberOut False=
Este trem continuara em frente ?.NumberOut False +
1:NEXT(202$)
Model statements for module: Delay 16
78$      DELAY:      0.0000166666666667,,Other:NEXT(725$)
Model statements for module: Decide 539
725$      BRANCH,      1:
          If,A Node <> 0,1667$,Yes:
          Else,1668$,Yes
1667$      ASSIGN:      Saindo de entroncamento direto?.NumberOut True=Saindo de
entroncamento direto?.NumberOut True + 1
          :NEXT(727$)
1668$      ASSIGN:      Saindo de entroncamento direto?.NumberOut False=Saindo
de entroncamento direto?.NumberOut False + 1
          :NEXT(211$)
Model statements for module: Decide 540
727$      BRANCH,      1:
          If,A NodeS == 1,729$,Yes:
          If,A nodeS == 2,730$,Yes:
          Else,731$,Yes
Model statements for module: Dispose 80
731$      ASSIGN:      Dispose 80.NumberOut=Dispose 80.NumberOut + 1
1671$      DISPOSE:      Yes
Model statements for module: Record 23
729$      COUNT:      Record 23,A IDTrem == Rastreio:NEXT(726$)

```

```

Model statements for module: Assign 644
726$      ASSIGN:      V Node ( A Node, 1)=0:
                          A Node=0:
                          A NodeS=0:NEXT(211$)
Model statements for module: Record 24
730$      COUNT:      Record 24,A IDTrem == Rastreio:NEXT(728$)
Model statements for module: Assign 645
728$      ASSIGN:      V Node ( A Node, 2)=0:
                          A Node=0:
                          A NodeS=0:NEXT(211$)
Model statements for module: Decide 336
211$      BRANCH,      1:
                          If,V Destino ( A PtAtual,1 + (A Sentido==0)) ==
0,1672$,Yes:
                          Else,1673$,Yes
1672$      ASSIGN:      Nao esta ultrapassando neste patio?.NumberOut True=
                          Nao esta ultrapassando neste patio?.NumberOut True +
1:NEXT(63$)
1673$      ASSIGN:      Nao esta ultrapassando neste patio?.NumberOut False=
                          Nao esta ultrapassando neste patio?.NumberOut False +
1:NEXT(76$)
Model statements for module: Assign 107
63$      ASSIGN:      V ocpt ( A PtAtual, 1 + (A Sentido==0) )=0:NEXT(76$)
Model statements for module: Signal 12
76$      SIGNAL:      A IDtrem,1:NEXT(61$)
Model statements for module: Assign 104
61$      ASSIGN:      A Patio=A PtAtual:
                          Indice2=Indice2 + 1:
                          A Sentido=X Sentido (Indice2):
                          A Trecho=X Trecho (Indice2):
                          A PtAtual=X Patio (Indice2):
                          A Node=X Node (Indice2):
                          A NodeS=X NodeS (Indice2):
                          A TmpTrecho=A Trecho:
                          A TmpSentido=A Sentido:
                          Temp1=1*(A Tipo<4) + 2*(A Tipo==4) + 3*(A
Tipo==5):NEXT(100$)
Model statements for module: Decide 109
100$      BRANCH,      1:
                          With,(( V Trecho( A Trecho , 6 + Temp1)
)))/100,1674$,Yes:
                          Else,1675$,Yes
1674$      ASSIGN:      Este movimento tera contingencia de via ?.NumberOut
True=
                          Este movimento tera contingencia de via ?.NumberOut True
+ 1:NEXT(101$)
1675$      ASSIGN:      Este movimento tera contingencia de via ?.NumberOut
False=
                          Este movimento tera contingencia de via ?.NumberOut
False + 1:NEXT(99$)
Model statements for module: Assign 134
101$      ASSIGN:      A Tmp=((1 + (A Sentido==0)) * Temp1 ) + 6:
                          A TransitTime=TT Trechos ( A Trecho , A Tmp):NEXT(577$)
Model statements for module: Decide 368
577$      BRANCH,      1:
                          If,A TransitTime <= 180,1676$,Yes:
                          Else,1677$,Yes
1676$      ASSIGN:      Decide 368.NumberOut True=Decide 368.NumberOut True +
1:NEXT(272$)
1677$      ASSIGN:      Decide 368.NumberOut False=Decide 368.NumberOut False +
1:NEXT(578$)
Model statements for module: Decide 369
578$      BRANCH,      1:
                          If,A Tmp>6,1678$,Yes:
                          Else,1679$,Yes

```



```

1678$      ASSIGN:      Decide 369.NumberOut True=Decide 369.NumberOut True +
1:NEXT(579$)
1679$      ASSIGN:      Decide 369.NumberOut False=Decide 369.NumberOut False +
1:NEXT(580$)
      Model statements for module:  ReadWrite 70
579$      WRITE,      Log de Erros,"%1.8g. TT CONTINGENCIA de %1.8g. Trem tipo
%1.8g. Trecho %1.8g. Coluna %1.8g.\n":
      TNOW,
      A TransitTime,
      A Tipo,
      A Trecho,
      A Tmp - 6:NEXT(581$)
      Model statements for module:  Assign 440
581$      ASSIGN:      A TransitTime=180:NEXT(272$)
      Model statements for module:  ReadWrite 71
580$      WRITE,      Log de Erros,"%1.8g. TT NORMAL de %1.8g. Trem tipo
%1.8g. Trecho %1.8g. Coluna %1.8g.\n":
      TNOW,
      A TransitTime,
      A Tipo,
      A Trecho,
      A Tmp:NEXT(581$)
      Model statements for module:  Assign 567
272$      ASSIGN:      A TransitTime=A TransitTime * (1 * (A SouHelper==0) + E
Helpers(3,1) * (A SouHelper==1)):NEXT(732$)
      Model statements for module:  Decide 560
732$      BRANCH,      1:
      If,E Verifica entroncamento .AND. V Ativa Entroncamento
.AND. 1,1680$,Yes:
      Else,1681$,Yes
1680$      ASSIGN:      Atingiu um entroncamento no caminho?.NumberOut True=
Atingiu um entroncamento no caminho?.NumberOut True +
1:NEXT(736$)
1681$      ASSIGN:      Atingiu um entroncamento no caminho?.NumberOut False=
Atingiu um entroncamento no caminho?.NumberOut False +
1:NEXT(165$)
      Model statements for module:  Decide 562
736$      BRANCH,      1:
      If,V ROT (Indice2+1+A Indirota+A Sentido, A Rota) ==
0,165$,Yes:
      If,A TmpSentido == 1,734$,Yes:
      If,A TmpSentido==0,735$,Yes:
      Else,737$,Yes
      Model statements for module:  ReadWrite 105
737$      WRITE,      Log de Erros,
"%1.8g. TTT F1 Trem %1.8g Rot %1.8g. IndRt %1.8g PtAtual
%1.8g. APatio %1.8g Restr %1.8g.SB %1.8g. A Trecho %1.8g Trecho Indrt %1.8g Trecho
Indrt m1 %1.8g Indice2 %1.8g.\n":
      TNOW,
      A IDtrem,
      A Rota,
      A IndiRota,
      A PtAtual,
      A Patio,
      A Restricao,
      A PassouSB,
      A Trecho,
      V ROT (A Indirota, A Rota),
      V ROT (A Indirota+1, A Rota),
      Indice2:NEXT(738$)
      Model statements for module:  Hold 80
738$      QUEUE,      Hold 80.Queue:DETACH
      Model statements for module:  Record 29
734$      COUNT:      Record 29,A IDTrem == Rastreio:NEXT(733$)
      Model statements for module:  Assign 653

```

```

733$      ASSIGN:      A Node=V Trecho ( V ROT (Indice2 + A Indirota , A Rota)
,1 ):
                                A NodeS=
                                1*(V DROT (Indice2 + A Indirota , A Rota)) + 2 * ((V
DROT (Indice2 + A Indirota , A Rota))==0)
                                :NEXT(165$)
      Model statements for module: Record 30
735$      COUNT:      Record 30,A IDTrem == Rastreio:NEXT(733$)
      Model statements for module: Decide 258
165$      BRANCH,      1:
                                If,A Ultrapassa==0 .AND. V Destino ( A PtAtual,1 + (A
Sentido==0)) == 0,1684$,Yes:
                                Else,1685$,Yes
1684$      ASSIGN:      Nao esta ultrapassando ?.NumberOut True=Nao esta
ultrapassando ?.NumberOut True + 1:NEXT(130$)
1685$      ASSIGN:      Nao esta ultrapassando ?.NumberOut False=Nao esta
ultrapassando ?.NumberOut False + 1:NEXT(164$)
      Model statements for module: Decide 199
130$      BRANCH,      1:
                                If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),1686$,Yes:
                                Else,1687$,Yes
1686$      ASSIGN:      Ponto de checagem 8.NumberOut True=Ponto de checagem
8.NumberOut True + 1:NEXT(131$)
1687$      ASSIGN:      Ponto de checagem 8.NumberOut False=Ponto de checagem
8.NumberOut False + 1:NEXT(314$)
      Model statements for module: Separate 19
131$      DUPLICATE,      100 - 50:
                                1,1690$,50:NEXT(1689$)
1689$      ASSIGN:      Separate 19.NumberOut Orig=Separate 19.NumberOut Orig +
1:NEXT(132$)
1690$      ASSIGN:      Separate 19.NumberOut Dup=Separate 19.NumberOut Dup +
1:NEXT(314$)
      Model statements for module: Hold 31
132$      QUEUE,      Hold 31.Queue:DETACH
      Model statements for module: Decide 583
314$      BRANCH,      1:
                                If,A Restricao2 == 1 .AND. V Trecho ( A TmpTrecho ,
3)==1,1691$,Yes:
                                Else,1692$,Yes
1691$      ASSIGN:      Trem restritivo vai continuar por SB ?.NumberOut True=
Trem restritivo vai continuar por SB ?.NumberOut True +
1:NEXT(315$)
1692$      ASSIGN:      Trem restritivo vai continuar por SB ?.NumberOut False=
Trem restritivo vai continuar por SB ?.NumberOut False +
1:NEXT(349$)
      Model statements for module: Assign 692
315$      ASSIGN:      A EstAtual=M:NEXT(316$)
      Model statements for module: Route 33
316$      ROUTE:      0,Restricao por SB
      Model statements for module: Decide 617
349$      BRANCH,      1:
                                If,A Node == 0,1693$,Yes:
                                Else,1694$,Yes
1693$      ASSIGN:      Nao esta passando por entroncamento ?.NumberOut True=
Nao esta passando por entroncamento ?.NumberOut True +
1:NEXT(328$)
1694$      ASSIGN:      Nao esta passando por entroncamento ?.NumberOut False=
Nao esta passando por entroncamento ?.NumberOut False +
1:NEXT(348$)
      Model statements for module: Decide 596
328$      BRANCH,      1:
                                If,E Verifica entroncamento,1695$,Yes:
                                Else,1696$,Yes

```

```

1695$      ASSIGN:      Continua por entroncamento ?.NumberOut True=Continua por
entroncamento ?.NumberOut True + 1
                        :NEXT(62$)
1696$      ASSIGN:      Continua por entroncamento ?.NumberOut False=Continua
por entroncamento ?.NumberOut False + 1
                        :NEXT(323$)
      Model statements for module:  Route 7
162$      ROUTE:      A TransitTime,ESTA ( A PtAtual,1 + ( A Sentido == 0 ) )
      Model statements for module:  Assign 695
323$      ASSIGN:      V Reserva (A TmpTrecho, 1+(A TmpSentido==0) )=V Reserva
(A TmpTrecho, 1+(A TmpSentido==0) ) - 1
                        :NEXT(62$)
      Model statements for module:  Assign 709
348$      ASSIGN:      V Node ( A Node, A NodeS)=0:
                        A Node=0:
                        A NodeS=0:NEXT(328$)
      Model statements for module:  Assign 336
164$      ASSIGN:      Sorteio=Unif(21000,25000,3):
                        A Ultrapassa=1:NEXT(163$)
      Model statements for module:  Separate 30
163$      DUPLICATE,   100 - 50:
                        1,1699$,50:NEXT(1698$)
1698$      ASSIGN:      Separate 30.NumberOut Orig=Separate 30.NumberOut Orig +
1:NEXT(161$)
1699$      ASSIGN:      Separate 30.NumberOut Dup=Separate 30.NumberOut Dup +
1:NEXT(130$)
      Model statements for module:  Assign 335
161$      ASSIGN:      Picture=Picture.ultrapassa:
                        A PtAtual=A Patio:
                        A Sentido=A Sentido == 0:NEXT(162$)
      Model statements for module:  Route 21
162$      ROUTE:      0.000000000000000,Station Anima Evento
      Model statements for module:  Assign 133
99$      ASSIGN:      A Tmp=((1 + (A Sentido==0)) * Temp1):
                        A TransitTime=TT Trechos ( A Trecho , A Tmp ):NEXT(577$)
      Model statements for module:  Assign 369
202$      ASSIGN:      A Ultrapassa=0:
                        A IndiRota=A Indirota + Indice + A Nsb:
                        A Nsb=0:
                        A Entrou em Fila=0:
                        A Restricao2=0:NEXT(74$)
      Model statements for module:  Decide 91
74$      BRANCH,      1:
                        If,A PtAtual == A PtDestino,1700$,Yes:
                        Else,1701$,Yes
1700$      ASSIGN:      Trem chegou ao patio de destino ?.NumberOut True=
Trem chegou ao patio de destino ?.NumberOut True +
1:NEXT(262$)
1701$      ASSIGN:      Trem chegou ao patio de destino ?.NumberOut False=
Trem chegou ao patio de destino ?.NumberOut False +
1:NEXT(139$)
      Model statements for module:  Decide 428
262$      BRANCH,      1:
                        If,A Direto==1,1702$,Yes:
                        Else,1703$,Yes
1702$      ASSIGN:      Entrou pela linha de atividade direto ?.NumberOut True=
Entrou pela linha de atividade direto ?.NumberOut True +
1:NEXT(263$)
1703$      ASSIGN:      Entrou pela linha de atividade direto ?.NumberOut False=
Entrou pela linha de atividade direto ?.NumberOut False
+ 1:NEXT(75$)
      Model statements for module:  Assign 482
263$      ASSIGN:      A Direto=0:NEXT(77$)
      Model statements for module:  Signal 13
77$      SIGNAL:      A IDtrem,1:NEXT(122$)

```

```

Model statements for module: Route 11
122$      ROUTE:          0.0000000000000000,Logica de Patios
Model statements for module: Assign 112
75$      ASSIGN:         V ocpt ( A PtAtual, 1 + (A Sentido==0) )=0:NEXT(77$)
Model statements for module: Decide 209
139$      BRANCH,       1:
                          If,V Rot (A IndiRota +1 , A Rota)>0,1704$,Yes:
                          Else,1705$,Yes
1704$      ASSIGN:       Nao tem problema com o roteiro ?.NumberOut True=Nao tem
problema com o roteiro ?.NumberOut True + 1
                          :NEXT(129$)
1705$      ASSIGN:       Nao tem problema com o roteiro ?.NumberOut False=
                          Nao tem problema com o roteiro ?.NumberOut False +
1:NEXT(144$)
Model statements for module: Decide 190
129$      BRANCH,       1:
                          If,A ColPatio == 1 .AND. A SouHelper==0,1706$,Yes:
                          Else,1707$,Yes
1706$      ASSIGN:       Trem esta ocupando um patio de atividade ?.NumberOut
True=
                          Trem esta ocupando um patio de atividade ?.NumberOut
True + 1:NEXT(425$)
1707$      ASSIGN:       Trem esta ocupando um patio de atividade ?.NumberOut
False=
                          Trem esta ocupando um patio de atividade ?.NumberOut
False + 1:NEXT(123$)
Model statements for module: Decide 200
425$      BRANCH,       1:
                          If,E Patios (A PtAtual, A ColPatio) == 0 .OR. A
DesistiuLinha==1,1708$,Yes:
                          Else,1709$,Yes
1708$      ASSIGN:       Atividade realizada na propria linha de transito
?.NumberOut True=
                          Atividade realizada na propria linha de transito
?.NumberOut True + 1:NEXT(443$)
1709$      ASSIGN:       Atividade realizada na propria linha de transito
?.NumberOut False=
                          Atividade realizada na propria linha de transito
?.NumberOut False + 1:NEXT(424$)
Model statements for module: Assign 289
443$      ASSIGN:       Sorteio=Unif(8000,12000)+Unif(235,2350):
                          marca=tnow:NEXT(440$)
Model statements for module: Separate 23
440$      DUPLICATE,    100 - 50:
                          1,1712$,50:NEXT(1711$)
1711$      ASSIGN:       Separate 23.NumberOut Orig=Separate 23.NumberOut Orig +
1:NEXT(441$)
1712$      ASSIGN:       Separate 23.NumberOut Dup=Separate 23.NumberOut Dup +
1:NEXT(427$)
Model statements for module: Assign 288
441$      ASSIGN:       Picture=Picture.atividade:NEXT(444$)
Model statements for module: Route 15
444$      ROUTE:          0.0000000000000000,Station Anima Evento
Model statements for module: Decide 214
427$      BRANCH,       1:
                          If,chechar >= 0 .AND. chechar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),1713$,Yes:
                          Else,1714$,Yes
1713$      ASSIGN:       Ponto de checagem 10.NumberOut True=Ponto de checagem
10.NumberOut True + 1:NEXT(428$)
1714$      ASSIGN:       Ponto de checagem 10.NumberOut False=Ponto de checagem
10.NumberOut False + 1:NEXT(423$)
Model statements for module: Separate 21
428$      DUPLICATE,    100 - 50:
                          1,1717$,50:NEXT(1716$)

```



```

0,
0,
0,
Attribute 16,
A Produto,
A Cliente,
A PtOrigem,
A PtDestino,
A PtZero:NEXT(442$)
Model statements for module: Signal 22
442$ SIGNAL: sorteio,1:NEXT(431$)
Model statements for module: Assign 257
431$ ASSIGN: A ColPatio=0:
A DesistiuLinha=0:NEXT(123$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 45
439$ WRITE, Log de MovimentacaoMDB,RECORDSET(LogdeEvento):
Dia,
A IDtrem,
A Ntrem,
A Tipo,
marca,
A PtAtual,
tnow,
5,
0,
0,
0,
0,
0,
0,
0,
0,
0,
Attribute 16,
A Produto,
A Cliente,
A PtOrigem,
A PtDestino,
A PtZero:NEXT(442$)
Model statements for module: Separate 35
446$ DUPLICATE, 100 - 50:
1,1728$,50:NEXT(1727$)
1727$ ASSIGN: Separate 35.NumberOut Orig=Separate 35.NumberOut Orig +
1:NEXT(447$)
1728$ ASSIGN: Separate 35.NumberOut Dup=Separate 35.NumberOut Dup +
1:NEXT(417$)
Model statements for module: Hold 44
447$ QUEUE, Hold 44.Queue:DETACH
Model statements for module: Assign 220
418$ ASSIGN: marca=tnow:NEXT(421$)
Model statements for module: Delay 29
421$ DELAY: E TmpGrade (A Ntrem, A IndiAtiv),,Other:NEXT(422$)
Model statements for module: Assign 222
422$ ASSIGN: A IndiAtiv=A IndiAtiv + 1:NEXT(449$)
Model statements for module: Assign 221
419$ ASSIGN: marca=tnow:NEXT(420$)

```

```

Model statements for module: Delay 28
420$      DELAY:          E TmpExpresso (A Ntrem, A IndiAtiv),,Other:NEXT(422$)
Model statements for module: Assign 248
424$      ASSIGN:        V ocpt ( A PtAtual, 1 + (A Sentido==0) )=0:NEXT(426$)
Model statements for module: Signal 19
426$      SIGNAL:        A IDtrem,1:NEXT(443$)
Model statements for module: Decide 221
433$      BRANCH,        1:
                        If,A Tipo == 4,432$,Yes:
                        If,A Tipo == 5,434$,Yes:
                        Else,435$,Yes
Model statements for module: Assign 279
435$      ASSIGN:        V LocosNow(1)=V LocosNow(1) + E FrotasT (A Ntrem, 1):
                        V LocosNow(2)=V LocosNow(2) + E FrotasT (A Ntrem, 2):
                        V LocosNow(3)=V LocosNow(3) + E FrotasT (A Ntrem, 3):
                        V LocosNow(4)=V LocosNow(4) + E FrotasT (A Ntrem,4):
                        V LocosNow(5)=V LocosNow(5) + E FrotasT (A Ntrem,
5):NEXT(431$)
Model statements for module: Assign 277
432$      ASSIGN:        V LocosNow(1)=V LocosNow(1) + E FrotasG (A Ntrem, 1):
                        V LocosNow(2)=V LocosNow(2) + E FrotasG (A Ntrem, 2):
                        V LocosNow(3)=V LocosNow(3) + E FrotasG (A Ntrem, 3):
                        V LocosNow(4)=V LocosNow(4) + E FrotasG (A Ntrem,4):
                        V LocosNow(5)=V LocosNow(5) + E FrotasG (A Ntrem,
5):NEXT(431$)
Model statements for module: Assign 278
434$      ASSIGN:        V LocosNow(1)=V LocosNow(1) + E FrotasE (A Ntrem, 1):
                        V LocosNow(2)=V LocosNow(2) + E FrotasE (A Ntrem, 2):
                        V LocosNow(3)=V LocosNow(3) + E FrotasE (A Ntrem, 3):
                        V LocosNow(4)=V LocosNow(4) + E FrotasE (A Ntrem,4):
                        V LocosNow(5)=V LocosNow(5) + E FrotasE (A Ntrem,
5):NEXT(431$)
Model statements for module: Assign 280
436$      ASSIGN:        V LocosNow(1)=V LocosNow(1) - E FrotasG (A Ntrem, 1):
                        V LocosNow(2)=V LocosNow(2) - E FrotasG (A Ntrem, 2):
                        V LocosNow(3)=V LocosNow(3) - E FrotasG (A Ntrem, 3):
                        V LocosNow(4)=V LocosNow(4) - E FrotasG (A Ntrem,4):
                        V LocosNow(5)=V LocosNow(5) - E FrotasG (A Ntrem,
5):NEXT(418$)
Model statements for module: Assign 281
437$      ASSIGN:        V LocosNow(1)=V LocosNow(1) - E FrotasE (A Ntrem, 1):
                        V LocosNow(2)=V LocosNow(2) - E FrotasE (A Ntrem, 2):
                        V LocosNow(3)=V LocosNow(3) - E FrotasE (A Ntrem, 3):
                        V LocosNow(4)=V LocosNow(4) - E FrotasE (A Ntrem,4):
                        V LocosNow(5)=V LocosNow(5) - E FrotasE (A Ntrem,
5):NEXT(419$)
Model statements for module: Assign 282
438$      ASSIGN:        V LocosNow(1)=V LocosNow(1) - E FrotasT (A Ntrem, 1):
                        V LocosNow(2)=V LocosNow(2) - E FrotasT (A Ntrem, 2):
                        V LocosNow(3)=V LocosNow(3) - E FrotasT (A Ntrem, 3):
                        V LocosNow(4)=V LocosNow(4) - E FrotasT (A Ntrem,4):
                        V LocosNow(5)=V LocosNow(5) - E FrotasT (A Ntrem,
5):NEXT(416$)
Model statements for module: Decide 178
123$      BRANCH,        1:
                        If,V Trecho (A Trecho, 9) == 0 .OR. 0,1731$,Yes:
                        Else,1732$,Yes
1731$     ASSIGN:        Trecho fora da CPTM ?.NumberOut True=Trecho fora da CPTM
?.NumberOut True + 1:NEXT(471$)
1732$     ASSIGN:        Trecho fora da CPTM ?.NumberOut False=Trecho fora da
CPTM ?.NumberOut False + 1:NEXT(124$)
Model statements for module: Assign 569
471$      ASSIGN:        A Tmp=TNOW:NEXT(456$)
Model statements for module: Decide 480
456$      BRANCH,        1:

```

```

                                If,A Tipo < 4 .AND. A SouHelper == 0 .AND. A
Estado==1,452$,Yes:
                                If,A PtAtual==7 .AND. A SouHelper == 0,457$,Yes:
                                Else,84$,Yes
                                Model statements for module: Decide 222
452$      BRANCH,      1:
                                If,A PtAtual==131 .AND. A Sentido == 1,455$,Yes:
                                If,A PtAtual==143 .AND. A Sentido == 0,458$,Yes:
                                If,A PtAtual==144,458$,Yes:
                                If,A PtAtual==1 .AND. A Sentido == 0,462$,Yes:
                                If,A PtAtual==7,459$,Yes:
                                If,A PtAtual==30 .OR. A PtAtual==37,466$,Yes:
                                If,A PtAtual==75,467$,Yes:
                                Else,84$,Yes
                                Model statements for module: Process 2
455$      ASSIGN:      1 Anexa Helpers em FCC.NumberIn=1 Anexa Helpers em
FCC.NumberIn + 1:
                                1 Anexa Helpers em FCC.WIP=1 Anexa Helpers em FCC.WIP+1
1740$      QUEUE,      1 Anexa Helpers em FCC.Queue
1739$      SEIZE,      2,VA:
                                Helpers1,2:NEXT(1738$)
1738$      DELAY:      0,,VA
1785$      ASSIGN:      1 Anexa Helpers em FCC.NumberOut=1 Anexa Helpers em
FCC.NumberOut + 1:
                                1 Anexa Helpers em FCC.WIP=1 Anexa Helpers em FCC.WIP-
1:NEXT(472$)
                                Model statements for module: Assign 570
472$      ASSIGN:      V FilasH(1,1)=V FilasH(1,1) + 1:
                                V FilasH(1,2)=V FilasH(1,2) + (TNOW - A Tmp):NEXT(487$)
                                Model statements for module: Delay 47
487$      DELAY:      E Helpers(1,1),,Other:NEXT(454$)
                                Model statements for module: Assign 295
454$      ASSIGN:      V LocosNow(5)=V LocosNow(5) + 2:
                                A NLocos=2:NEXT(461$)
                                Model statements for module: Assign 551
461$      ASSIGN:      A RetornoH=A PtAtual:NEXT(84$)
                                Model statements for module: Search 8
458$      FINDJ,      A Indirota,A Indirota + 16:V Trecho (V ROT(J, A Rota) +
(183*(V ROT(J, A Rota)==0)) , 2) == 8
1788$      BRANCH,      1:
                                If,J<>0,1789$,Yes:
                                Else,1790$,Yes
1789$      DELAY:      0.0,,VA:NEXT(473$)
1790$      DELAY:      0.0,,VA:NEXT(84$)
                                Model statements for module: Decide 489
473$      BRANCH,      1:
                                If,A PtAtual==143,1791$,Yes:
                                Else,1792$,Yes
1791$      ASSIGN:      Esta em FAL ?.NumberOut True=Esta em FAL ?.NumberOut
True + 1:NEXT(483$)
1792$      ASSIGN:      Esta em FAL ?.NumberOut False=Esta em FAL ?.NumberOut
False + 1:NEXT(484$)
                                Model statements for module: Process 15
483$      ASSIGN:      2 Anexa Helpers em FAL.NumberIn=2 Anexa Helpers em
FAL.NumberIn + 1:
                                2 Anexa Helpers em FAL.WIP=2 Anexa Helpers em FAL.WIP+1
1796$      QUEUE,      2 Anexa Helpers em FAL.Queue
1795$      SEIZE,      2,VA:
                                Helpers2,2:NEXT(1794$)
1794$      DELAY:      0,,VA
1841$      ASSIGN:      2 Anexa Helpers em FAL.NumberOut=2 Anexa Helpers em
FAL.NumberOut + 1:
                                2 Anexa Helpers em FAL.WIP=2 Anexa Helpers em FAL.WIP-
1:NEXT(474$)
                                Model statements for module: Assign 571

```



```

474$          ASSIGN:          V FilasH(2,1)=V FilasH(2,1) + 1:
                                V FilasH(2,2)=V FilasH(2,2) + (TNOW - A Tmp):NEXT(487$)
      Model statements for module: Process 16
484$          ASSIGN:          2 Anexa Helpers em FLP.NumberIn=2 Anexa Helpers em
FLP.NumberIn + 1:
                                2 Anexa Helpers em FLP.WIP=2 Anexa Helpers em FLP.WIP+1
1847$         QUEUE,          2 Anexa Helpers em FLP.Queue
1846$         SEIZE,          2,VA:
                                Helpers2,2:NEXT(1845$)
1845$         DELAY:          0,,VA
1892$         ASSIGN:          2 Anexa Helpers em FLP.NumberOut=2 Anexa Helpers em
FLP.NumberOut + 1:
                                2 Anexa Helpers em FLP.WIP=2 Anexa Helpers em FLP.WIP-
1:NEXT(475$)
      Model statements for module: Assign 572
475$          ASSIGN:          V FilasH(3,1)=V FilasH(3,1) + 1:
                                V FilasH(3,2)=V FilasH(3,2) + (TNOW - A Tmp):NEXT(487$)
      Model statements for module: Search 11
462$         FINDJ,          A Indirota,A Indirota + 12:V Trecho (V ROT(J, A Rota) +
(183*(V ROT(J, A Rota)==0)) , 2) == 3
1895$         BRANCH,          1:
                                If,J<>0,1896$,Yes:
                                Else,1897$,Yes
1896$         DELAY:          0.0,,VA:NEXT(463$)
1897$         DELAY:          0.0,,VA:NEXT(84$)
      Model statements for module: Process 7
463$         ASSIGN:          3 Anexa Helpers em FOO.NumberIn=3 Anexa Helpers em
FOO.NumberIn + 1:
                                3 Anexa Helpers em FOO.WIP=3 Anexa Helpers em FOO.WIP+1
1901$         QUEUE,          3 Anexa Helpers em FOO.Queue
1900$         SEIZE,          2,VA:
                                Helpers3,3:NEXT(1899$)
1899$         DELAY:          0,,VA
1946$         ASSIGN:          3 Anexa Helpers em FOO.NumberOut=3 Anexa Helpers em
FOO.NumberOut + 1:
                                3 Anexa Helpers em FOO.WIP=3 Anexa Helpers em FOO.WIP-
1:NEXT(476$)
      Model statements for module: Assign 573
476$         ASSIGN:          V FilasH(4,1)=V FilasH(4,1) + 1:
                                V FilasH(4,2)=V FilasH(4,2) + (TNOW - A Tmp):NEXT(488$)
      Model statements for module: Delay 48
488$         DELAY:          E Helpers(1,1),,Other:NEXT(469$)
      Model statements for module: Assign 557
469$         ASSIGN:          V LocosNow(5)=V LocosNow(5) + 3:
                                A NLocos=3:NEXT(461$)
      Model statements for module: Search 10
459$         FINDJ,          A Indirota,A Indirota + 16:V Trecho (V ROT(J, A Rota) +
(183*(V ROT(J, A Rota)==0)) , 2) == 8
1949$         BRANCH,          1:
                                If,J<>0,1950$,Yes:
                                Else,1951$,Yes
1950$         DELAY:          0.0,,VA:NEXT(464$)
1951$         DELAY:          0.0,,VA:NEXT(84$)
      Model statements for module: Decide 482
464$         BRANCH,          1:
                                If,A Tipo < 4,1952$,Yes:
                                Else,1953$,Yes
1952$         ASSIGN:          Tabelas ocupam 2 helpers.NumberOut True=Tabelas ocupam 2
helpers.NumberOut True + 1:NEXT(460$)
1953$         ASSIGN:          Tabelas ocupam 2 helpers.NumberOut False=Tabelas ocupam
2 helpers.NumberOut False + 1:NEXT(465$)
      Model statements for module: Process 5
460$         ASSIGN:          5 Anexa 2 Helpers em FJC.NumberIn=5 Anexa 2 Helpers em
FJC.NumberIn + 1:

```

```

5 Anexa 2 Helpers em FJC.WIP=5 Anexa 2 Helpers em
FJC.WIP+1
1957$    QUEUE,    5 Anexa 2 Helpers em FJC.Queue
1956$    SEIZE,    1,VA:
                    Helpers5,2:NEXT(1955$)
1955$    DELAY:    0,,VA
2002$    ASSIGN:   5 Anexa 2 Helpers em FJC.NumberOut=5 Anexa 2 Helpers em
FJC.NumberOut + 1:
                    5 Anexa 2 Helpers em FJC.WIP=5 Anexa 2 Helpers em
FJC.WIP-1:NEXT(477$)
  Model statements for module: Assign 574
477$    ASSIGN:    V FilasH(5,1)=V FilasH(5,1) + 1:
                    V FilasH(5,2)=V FilasH(5,2) + (TNOW - A Tmp):NEXT(487$)
  Model statements for module: Process 8
465$    ASSIGN:    5 Anexa 1 Helper em FJC.NumberIn=5 Anexa 1 Helper em
FJC.NumberIn + 1:
                    5 Anexa 1 Helper em FJC.WIP=5 Anexa 1 Helper em
FJC.WIP+1
2008$    QUEUE,    5 Anexa 1 Helper em FJC.Queue
2007$    SEIZE,    2,VA:
                    Helpers5,1:NEXT(2006$)
2006$    DELAY:    0,,VA
2053$    ASSIGN:   5 Anexa 1 Helper em FJC.NumberOut=5 Anexa 1 Helper em
FJC.NumberOut + 1:
                    5 Anexa 1 Helper em FJC.WIP=5 Anexa 1 Helper em FJC.WIP-
1:NEXT(478$)
  Model statements for module: Assign 575
478$    ASSIGN:    V FilasH(5,1)=V FilasH(5,1) + 1:
                    V FilasH(5,2)=V FilasH(5,2) + (TNOW - A Tmp):NEXT(489$)
  Model statements for module: Delay 49
489$    DELAY:    E Helpers(1,1),,Other:NEXT(453$)
  Model statements for module: Assign 293
453$    ASSIGN:    V LocosNow(5)=V LocosNow(5) + 1:
                    A NLocos=1:NEXT(461$)
  Model statements for module: Search 12
466$    FINDJ,    A Indirota,A Indirota + 16:V Trecho (V ROT(J, A Rota) +
(183*(V ROT(J, A Rota)==0)) , 2) == 33
2056$    BRANCH,    1:
                    If,J<>0,2057$,Yes:
                    Else,2058$,Yes
2057$    DELAY:    0.0,,VA:NEXT(479$)
2058$    DELAY:    0.0,,VA:NEXT(84$)
  Model statements for module: Decide 490
479$    BRANCH,    1:
                    If,A PtAtual==30,2059$,Yes:
                    Else,2060$,Yes
2059$    ASSIGN:    Esta em FBP ?.NumberOut True=Esta em FBP ?.NumberOut
True + 1:NEXT(485$)
2060$    ASSIGN:    Esta em FBP ?.NumberOut False=Esta em FBP ?.NumberOut
False + 1:NEXT(486$)
  Model statements for module: Process 17
485$    ASSIGN:    6 Anexa Helpers em FBP.NumberIn=6 Anexa Helpers em
FBP.NumberIn + 1:
                    6 Anexa Helpers em FBP.WIP=6 Anexa Helpers em FBP.WIP+1
2064$    QUEUE,    6 Anexa Helpers em FBP.Queue
2063$    SEIZE,    1,VA:
                    Helpers6,4:NEXT(2062$)
2062$    DELAY:    0,,VA
2109$    ASSIGN:   6 Anexa Helpers em FBP.NumberOut=6 Anexa Helpers em
FBP.NumberOut + 1:
                    6 Anexa Helpers em FBP.WIP=6 Anexa Helpers em FBP.WIP-
1:NEXT(480$)
  Model statements for module: Assign 576
480$    ASSIGN:    V FilasH(6,1)=V FilasH(6,1) + 1:
                    V FilasH(6,2)=V FilasH(6,2) + (TNOW - A Tmp):NEXT(490$)

```

```

Model statements for module: Delay 50
490$      DELAY:          E Helpers(1,1),,Other:NEXT(470$)
Model statements for module: Assign 558
470$      ASSIGN:        V LocosNow(5)=V LocosNow(5) + 4:
                          A NLocos=4:NEXT(461$)
Model statements for module: Process 18
486$      ASSIGN:        6 Anexa Helpers em FQS.NumberIn=6 Anexa Helpers em
FQS.NumberIn + 1:
                          6 Anexa Helpers em FQS.WIP=6 Anexa Helpers em FQS.WIP+1
2115$     QUEUE,        6 Anexa Helpers em FQS.Queue
2114$     SEIZE,        2,VA:
                          Helpers6,4:NEXT(2113$)
2113$     DELAY:        0,,VA
2160$     ASSIGN:        6 Anexa Helpers em FQS.NumberOut=6 Anexa Helpers em
FQS.NumberOut + 1:
                          6 Anexa Helpers em FQS.WIP=6 Anexa Helpers em FQS.WIP-
1:NEXT(481$)
Model statements for module: Assign 577
481$      ASSIGN:        V FilasH(7,1)=V FilasH(7,1) + 1:
                          V FilasH(7,2)=V FilasH(7,2) + (TNOW - A Tmp):NEXT(490$)
Model statements for module: Search 13
467$      FINDJ,        A Indirota,A Indirota + 16:V Trecho (V ROT(J, A Rota) +
(183*(V ROT(J, A Rota)==0)) , 2) == 179
2163$     BRANCH,        1:
                          If,J<>0,2164$,Yes:
                          Else,2165$,Yes
2164$     DELAY:        0.0,,VA:NEXT(468$)
2165$     DELAY:        0.0,,VA:NEXT(84$)
Model statements for module: Process 13
468$      ASSIGN:        7 Anexa Helpers em FSJ.NumberIn=7 Anexa Helpers em
FSJ.NumberIn + 1:
                          7 Anexa Helpers em FSJ.WIP=7 Anexa Helpers em FSJ.WIP+1
2169$     QUEUE,        7 Anexa Helpers em FSJ.Queue
2168$     SEIZE,        2,VA:
                          Helpers7,1:NEXT(2167$)
2167$     DELAY:        0,,VA
2214$     ASSIGN:        7 Anexa Helpers em FSJ.NumberOut=7 Anexa Helpers em
FSJ.NumberOut + 1:
                          7 Anexa Helpers em FSJ.WIP=7 Anexa Helpers em FSJ.WIP-
1:NEXT(482$)
Model statements for module: Assign 578
482$      ASSIGN:        V FilasH(8,1)=V FilasH(8,1) + 1:
                          V FilasH(8,2)=V FilasH(8,2) + (TNOW - A Tmp):NEXT(489$)
Model statements for module: Search 7
457$      FINDJ,        A Indirota,A Indirota + 26:V Trecho (V ROT(J, A Rota) +
(183*(V ROT(J, A Rota)==0)) , 2) == 21
2217$     BRANCH,        1:
                          If,J<>0,2218$,Yes:
                          Else,2219$,Yes
2218$     DELAY:        0.0,,VA:NEXT(459$)
2219$     DELAY:        0.0,,VA:NEXT(84$)
Model statements for module: Decide 93
84$       BRANCH,        1:
                          If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2220$,Yes:
                          Else,2221$,Yes
2220$     ASSIGN:        Ponto de checagem 3.NumberOut True=Ponto de checagem
3.NumberOut True + 1:NEXT(85$)
2221$     ASSIGN:        Ponto de checagem 3.NumberOut False=Ponto de checagem
3.NumberOut False + 1:NEXT(134$)
Model statements for module: Separate 8
85$       DUPLICATE,    100 - 50:
                          1,2224$,50:NEXT(2223$)
2223$     ASSIGN:        Separate 8.NumberOut Orig=Separate 8.NumberOut Orig +
1:NEXT(86$)

```

```

2224$      ASSIGN:      Separate 8.NumberOut Dup=Separate 8.NumberOut Dup +
1:NEXT(134$)
      Model statements for module:  Hold 15
86$        QUEUE,      Hold 15.Queue:DETACH
      Model statements for module:  Decide 204
134$        BRANCH,    1:
                        If,V ROT(A IndiRota+1, A Rota)==0,2225$,Yes:
                        Else,2226$,Yes
2225$      ASSIGN:      Proximo trecho zerado ?.NumberOut True=Proximo trecho
zerado ?.NumberOut True + 1:NEXT(135$)
2226$      ASSIGN:      Proximo trecho zerado ?.NumberOut False=Proximo trecho
zerado ?.NumberOut False + 1:NEXT(753$)
      Model statements for module:  ReadWrite 33
135$        WRITE,     Log de Erros,
                        "%1.8g. ERR3 Erro no trem %1.8g. Tipo %1.8g. Roteiro
%1.8g. Com destino a %1.8g. Roteiro terminado em %1.8g sem encontrar patio de
destino. Trem abortado.\n":
                        TNOW,
                        A IDtrem,
                        A Tipo,
                        A Rota,
                        A PtDestino,
                        A PtAtual:NEXT(138$)
      Model statements for module:  Signal 17
138$        SIGNAL:    A IDtrem,1:NEXT(137$)
      Model statements for module:  Assign 249
137$        ASSIGN:    A Trecho=V ROT(A IndiRota - ((V DROT(A IndiRota, A
Rota)) == 0), A Rota):
                        V Ocpt ( A PtAtual , 1+ (A Sentido==0) )=0:NEXT(136$)
      Model statements for module:  Dispose 40
136$        ASSIGN:    Dispose 40.NumberOut=Dispose 40.NumberOut + 1
2227$        DISPOSE:   Yes
      Model statements for module:  Decide 647
753$        BRANCH,    1:
                        If,A PtAtual==147 .AND. A Sentido==1,2228$,Yes:
                        Else,2229$,Yes
2228$      ASSIGN:    Esta em FQT indo para FDM ?.NumberOut True=Esta em FQT
indo para FDM ?.NumberOut True + 1
                        :NEXT(754$)
2229$      ASSIGN:    Esta em FQT indo para FDM ?.NumberOut False=Esta em FQT
indo para FDM ?.NumberOut False + 1
                        :NEXT(759$)
      Model statements for module:  Decide 648
754$        BRANCH,    1:
                        If,A PtDestino == 351 .AND. A Tipo==1,2230$,Yes:
                        Else,2231$,Yes
2230$      ASSIGN:    Caso de excecao Minerio indo para FOB?.NumberOut True=
Caso de excecao Minerio indo para FOB?.NumberOut True +
1:NEXT(755$)
2231$      ASSIGN:    Caso de excecao Minerio indo para FOB?.NumberOut False=
Caso de excecao Minerio indo para FOB?.NumberOut False +
1:NEXT(209$)
      Model statements for module:  Decide 649
755$        BRANCH,    1:
                        If,(V Node (36,1)==1) .OR. (V StatusPt (351,
3)==0),2232$,Yes:
                        Else,2233$,Yes
2232$      ASSIGN:    Caso de excecao ?.NumberOut True=Caso de excecao
?.NumberOut True + 1:NEXT(756$)
2233$      ASSIGN:    Caso de excecao ?.NumberOut False=Caso de excecao
?.NumberOut False + 1:NEXT(209$)
      Model statements for module:  Assign 714
756$        ASSIGN:    Temp2=A Produto * (A Produto <> 2) + 1 * (A Produto ==
2):
                        V Destino ( A PtAtual,1 + (A Sentido==0))=A PtDestino:

```

```

A Entrou em Fila=1:
marca=tnow:
V DestinoP ( A PtAtual,1 + (A Sentido==0))=Temp2:
A PassouFQD=1:NEXT(757$)
Model statements for module: Hold 89
757$      QUEUE,      Testa se liberou terminal.Queue
          SCAN:      (V Node (36,1)==0) .AND. (V StatusPt (351,
3)>0):NEXT(209$)
Model statements for module: Decide 660
759$      BRANCH,      1:
          If,A PtAtual==149 .AND. A Sentido==1,2234$,Yes:
          Else,2235$,Yes
2234$      ASSIGN:      Esta em FMO indo para FDM ?.NumberOut True=Esta em FMO
indo para FDM ?.NumberOut True + 1
          :NEXT(760$)
2235$      ASSIGN:      Esta em FMO indo para FDM ?.NumberOut False=Esta em FMO
indo para FDM ?.NumberOut False + 1
          :NEXT(209$)
Model statements for module: Decide 661
760$      BRANCH,      1:
          If,NQ(FQN_X_queue) > 0,2236$,Yes:
          Else,2237$,Yes
2236$      ASSIGN:      Trem esperando em FQN sentido FDM ?.NumberOut True=
Trem esperando em FQN sentido FDM ?.NumberOut True +
1:NEXT(763$)
2237$      ASSIGN:      Trem esperando em FQN sentido FDM ?.NumberOut False=
Trem esperando em FQN sentido FDM ?.NumberOut False +
1:NEXT(209$)
Model statements for module: Decide 662
763$      BRANCH,      1:
          If,AQUE(FQN_X_queue,1,NSYM(A PTDestino)) ==
351,2238$,Yes:
          Else,2239$,Yes
2238$      ASSIGN:      Trem destinado a FOB ?.NumberOut True=Trem destinado a
FOB ?.NumberOut True + 1:NEXT(761$)
2239$      ASSIGN:      Trem destinado a FOB ?.NumberOut False=Trem destinado a
FOB ?.NumberOut False + 1:NEXT(209$)
Model statements for module: Hold 90
761$      QUEUE,      Aguarda saida do trem de execucao.Queue
          SCAN:      NQ(FQN_X_queue)==0:NEXT(209$)
Model statements for module: Decide 333
209$      BRANCH,      1:
          If,A PtAtual == V Trecho (V ROT(A Indirota + 1, A Rota)
, 2 ),2240$,Yes:
          Else,2241$,Yes
2240$      ASSIGN:      Atingiu conexao de trechos ?.NumberOut True=Atingiu
conexao de trechos ?.NumberOut True + 1
          :NEXT(210$)
2241$      ASSIGN:      Atingiu conexao de trechos ?.NumberOut False=Atingiu
conexao de trechos ?.NumberOut False + 1
          :NEXT(253$)
Model statements for module: Assign 380
210$      ASSIGN:      A IndiRota=A Indirota + 1:NEXT(253$)
Model statements for module: Decide 422
253$      BRANCH,      1:
          If,V ROT(A Indirota+3, A Rota)<>0,2242$,Yes:
          Else,2243$,Yes
2242$      ASSIGN:      Este roteiro nao termina nos proximos 3 trechos
?.NumberOut True=
          Este roteiro nao termina nos proximos 3 trechos
?.NumberOut True + 1:NEXT(254$)
2243$      ASSIGN:      Este roteiro nao termina nos proximos 3 trechos
?.NumberOut False=
          Este roteiro nao termina nos proximos 3 trechos
?.NumberOut False + 1:NEXT(255$)

```

```

Model statements for module: Assign 472
254$      ASSIGN:      A SgDestino=E Patios ( V Trecho (V ROT(A Indirota+3, A
Rota) , 2) ,10):NEXT(7$)
  ATRIBUTOS
  Indice: conta quantos trechos foram
  verificados. Marca a posição de A Stop
  sendo usada
  Indicel: Posição do roteiro em que está
  sendo feita a verificação. Começa com
  valor de A IndiRota e vai sendo incrementado.
Model statements for module: Assign 69
7$        ASSIGN:      Indice=1:
                          Indicel=A IndiRota+1:
                          A Restricao=0:
                          A PassouSB=0:
                          A Ultrapassa=0:
                          A Nsb=0:
                          A Aviso=0:
                          A RestFila=0:NEXT(9$)
Model statements for module: Assign 70
9$        ASSIGN:      A Sentido=V DROT(Indicel-1, A Rota):
                          A Trecho=V ROT(Indicel- 1, A Rota) - (A Sentido==0):
                          A Patio=V Trecho (V ROT(Indicel, A Rota) , 2 ):
                          A FilaNode=E Verifica entroncamento:NEXT(12$)
Model statements for module: Decide 58
12$       BRANCH,      1:
                          If,E Patios (A Patio , 6 ) == 1,2244$,Yes:
                          Else,2245$,Yes
2244$     ASSIGN:      Patio esta ativado ?.NumberOut True=Patio esta ativado
?.NumberOut True + 1:NEXT(203$)
2245$     ASSIGN:      Patio esta ativado ?.NumberOut False=Patio esta ativado
?.NumberOut False + 1:NEXT(34$)
Model statements for module: Decide 329
203$     BRANCH,      1:
                          If,V Trecho(A Trecho,5) > 1 .OR. A Patio == A
PtDestino,2246$,Yes:
                          Else,2247$,Yes
2246$     ASSIGN:      Encontrou linha dupla ou patio de destino ?.NumberOut
True=
                          Encontrou linha dupla ou patio de destino ?.NumberOut
True + 1:NEXT(232$)
2247$     ASSIGN:      Encontrou linha dupla ou patio de destino ?.NumberOut
False=
                          Encontrou linha dupla ou patio de destino ?.NumberOut
False + 1:NEXT(8$)
Model statements for module: Decide 362
332$     BRANCH,      1:
                          If,A Patio == A PtDestino,2248$,Yes:
                          Else,2249$,Yes
2248$     ASSIGN:      Patio de destino ?.NumberOut True=Patio de destino
?.NumberOut True + 1:NEXT(205$)
2249$     ASSIGN:      Patio de destino ?.NumberOut False=Patio de destino
?.NumberOut False + 1:NEXT(273$)
Model statements for module: Assign 378
205$     ASSIGN:      X Sentido(Indice)=A Sentido:
                          X Trecho(Indice)=A Trecho:
                          X Patio(Indice)=A Patio:NEXT(292$)
Model statements for module: Search 18
292$     FINDJ,        1,50:V IndTrecho(J,1) == A Trecho + (A Sentido==0)
2250$     BRANCH,      1:
                          If,J<>0,2251$,Yes:
                          Else,2252$,Yes
2251$     DELAY:      0.0,,VA:NEXT(296$)
2252$     DELAY:      0.0,,VA:NEXT(295$)
Model statements for module: Assign 641

```

```

296$      ASSIGN:      X NodeS(Indice)=1:NEXT(293$)
      Model statements for module: Assign 639
293$      ASSIGN:      X Node(Indice)=J:NEXT(16$)
      Model statements for module: Assign 74
16$       ASSIGN:      A Sentido=V DROT( A IndiRota, A Rota ):
      A Trecho=V ROT(A Indirota , A Rota):
      Indicel=A Indirota:
      A ColPatio=0:
      A Excecao=0:NEXT(616$)
      Model statements for module: Decide 402
616$      BRANCH,      1:
      If,
      (A PtAtual==29) .OR. (A PtAtual==55) .OR. (A
PtAtual==82) .OR. (A PtAtual==31) .AND. A SgDestino==12 .AND. A Repescagem==0,
      629$,Yes:
      If,
      (A PtAtual==28) .OR. (A PtAtual==56) .OR. (A
PtAtual==83) .OR. (A PtAtual==32) .AND. A Restricao==1,
      215$,Yes:
      Else,215$,Yes
      Model statements for module: Decide 421
629$      BRANCH,      1:
      If,V ContagemFVR > 15,2255$,Yes:
      Else,2256$,Yes
2255$     ASSIGN:      Precisa esperar em fila de FVR ?.NumberOut True=Precisa
esperar em fila de FVR ?.NumberOut True + 1
      :NEXT(627$)
2256$     ASSIGN:      Precisa esperar em fila de FVR ?.NumberOut False=
Precisa esperar em fila de FVR ?.NumberOut False +
1:NEXT(630$)
      Model statements for module: Assign 467
627$     ASSIGN:      V Destino ( A PtAtual,1 + (A Sentido==0))=6000:
      marca=tnow:
      V DestinoP ( A PtAtual,1 + (A Sentido==0))=0:NEXT(623$)
      Model statements for module: Decide 411
623$     BRANCH,      1:
      If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2257$,Yes:
      Else,2258$,Yes
2257$     ASSIGN:      Ponto de checagem 30.NumberOut True=Ponto de checagem
30.NumberOut True + 1:NEXT(624$)
2258$     ASSIGN:      Ponto de checagem 30.NumberOut False=Ponto de checagem
30.NumberOut False + 1:NEXT(620$)
      Model statements for module: Separate 60
624$     DUPLICATE,    100 - 50:
      1,2261$,50:NEXT(2260$)
2260$     ASSIGN:      Separate 60.NumberOut Orig=Separate 60.NumberOut Orig +
1:NEXT(625$)
2261$     ASSIGN:      Separate 60.NumberOut Dup=Separate 60.NumberOut Dup +
1:NEXT(620$)
      Model statements for module: Hold 73
625$     QUEUE,        Hold 73.Queue:DETACH
      Model statements for module: Seize 8
620$     QUEUE,        Ocupa controle de Fvr.Queue
      SEIZE,           2,Other:
      Controle de FVR,1:NEXT(2263$)
2263$     DELAY:       0.0,,VA:NEXT(618$)
      Model statements for module: Hold 72
618$     QUEUE,        Aguarda espaco no trecho de FVR.Queue
      SCAN:           V ContagemFVR < 16:NEXT(617$)
      Model statements for module: Assign 455
617$     ASSIGN:      V ContagemFVR=V ContagemFVR + 1:
      A FVR=1:NEXT(628$)
      Model statements for module: Assign 468
628$     ASSIGN:      V Destino ( A PtAtual,1 + (A Sentido==0))=0:

```

```

          V DestinoP ( A PtAtual,1 + (A Sentido==0))=0:NEXT(622$)
Model statements for module: Release 16
622$      RELEASE:      Controle de FVR,1:NEXT(215$)
Model statements for module: Assign 471
630$      ASSIGN:      V ContagemFVR=V ContagemFVR + 1:
          A FVR=1:NEXT(215$)
Model statements for module: Decide 345
215$      BRANCH,      1:
          If,A Entrou em Fila == 1 .AND. A Tipo < 4,2264$,Yes:
          Else,2265$,Yes
2264$      ASSIGN:      Trem tabela entrou em fila ?.NumberOut True=Trem tabela
entrou em fila ?.NumberOut True + 1
          :NEXT(267$)
2265$      ASSIGN:      Trem tabela entrou em fila ?.NumberOut False=Trem tabela
entrou em fila ?.NumberOut False + 1
          :NEXT(335$)
Model statements for module: Decide 434
267$      BRANCH,      1:
          If,TNOW > (V Warmup * 1440),2266$,Yes:
          Else,2267$,Yes
2266$      ASSIGN:      Warmup do log de eventos 2.NumberOut True=Warmup do log
de eventos 2.NumberOut True + 1:NEXT(351$)
2267$      ASSIGN:      Warmup do log de eventos 2.NumberOut False=Warmup do log
de eventos 2.NumberOut False + 1
          :NEXT(217$)
Model statements for module: Decide 631
351$      BRANCH,      1:
          If,V Gravar em texto == 1,2268$,Yes:
          Else,2269$,Yes
2268$      ASSIGN:      Decide 631.NumberOut True=Decide 631.NumberOut True +
1:NEXT(357$)
2269$      ASSIGN:      Decide 631.NumberOut False=Decide 631.NumberOut False +
1:NEXT(216$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 157
357$      WRITE,      Log de Eventos:
          Dia,
          A IDtrem,
          A Ntrem,
          A Tipo,
          marca,
          A PtAtual,
          tnow,
          9 * (A Estado ==1) + 8 * (A Estado == 0),
          A Peso,
          A Dist,
          E FrotasT (A Ntrem, 1),
          E FrotasT (A Ntrem, 2),
          E FrotasT (A Ntrem, 3),
          E FrotasT (A Ntrem, 4),
          E FrotasT (A Ntrem, 5),
          A Estado,
          A Produto,
          A Cliente,
          A PtOrigem,

```



```

                A PtDestino,
                A PtZero:NEXT(217$)
Model statements for module: Assign 418
217$      ASSIGN:      A Entrou em Fila=0:
                marca=tnow:NEXT(354$)
Model statements for module: Assign 717
354$      ASSIGN:      V Destino ( A PtAtual,1 + (A Sentido==0))=0:
                V DestinoP ( A PtAtual,1 + (A Sentido==0))=0:
                A FilaNode=0:
                A PassouFQD=0:NEXT(335$)
Model statements for module: Decide 608
335$      BRANCH,      1:
                If, checar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2270$,Yes:
                Else,2271$,Yes
2270$      ASSIGN:      Ponto de checagem 13.NumberOut True=Ponto de checagem
13.NumberOut True + 1:NEXT(336$)
2271$      ASSIGN:      Ponto de checagem 13.NumberOut False=Ponto de checagem
13.NumberOut False + 1:NEXT(521$)
Model statements for module: Separate 74
336$      DUPLICATE,    100 - 50:
                1,2274$,50:NEXT(2273$)
2273$      ASSIGN:      Separate 74.NumberOut Orig=Separate 74.NumberOut Orig +
1:NEXT(337$)
2274$      ASSIGN:      Separate 74.NumberOut Dup=Separate 74.NumberOut Dup +
1:NEXT(521$)
Model statements for module: Hold 87
337$      QUEUE,        Hold 87.Queue:DETACH
Model statements for module: Decide 183
521$      BRANCH,      1:
                If,A Patio == A PtDestino .AND. A Tipo <4 .AND. A
SouHelper == 0,2275$,Yes:
                Else,2276$,Yes
2275$      ASSIGN:      Destino do trem unitario no proximo patio ?.NumberOut
True=
                Destino do trem unitario no proximo patio ?.NumberOut
True + 1:NEXT(543$)
2276$      ASSIGN:      Destino do trem unitario no proximo patio ?.NumberOut
False=
                Destino do trem unitario no proximo patio ?.NumberOut
False + 1:NEXT(517$)
Model statements for module: Decide 301
543$      BRANCH,      1:
                If,A PtDestino <> 151 .AND. A PtDestino <> 7 .AND. A
PtDestino <> 148 .OR. A Etapa <> 3,2277$,Yes:
                Else,2278$,Yes
2277$      ASSIGN:      Proximo patio nao eh ponto zero ?.NumberOut True=
                Proximo patio nao eh ponto zero ?.NumberOut True +
1:NEXT(522$)
2278$      ASSIGN:      Proximo patio nao eh ponto zero ?.NumberOut False=
                Proximo patio nao eh ponto zero ?.NumberOut False +
1:NEXT(528$)
Model statements for module: Decide 184
522$      BRANCH,      1:
                If,A Tipo == 2 .OR. (A Tipo == 3 .AND. A Estado ==
0),2279$,Yes:
                Else,2280$,Yes
2279$      ASSIGN:      Vai carregar ?.NumberOut True=Vai carregar ?.NumberOut
True + 1:NEXT(523$)
2280$      ASSIGN:      Vai carregar ?.NumberOut False=Vai carregar ?.NumberOut
False + 1:NEXT(524$)
Model statements for module: Assign 232
523$      ASSIGN:      A ColPatio=2:NEXT(530$)
Model statements for module: Decide 188
530$      BRANCH,      1:

```

```

                If,E Patios (A Patio, A ColPatio) > 0,2281$,Yes:
                Else,2282$,Yes
2281$      ASSIGN:      Este patio tem este tipo de linha originalmente
?.NumberOut True=
                Este patio tem este tipo de linha originalmente
?.NumberOut True + 1:NEXT(545$)
2282$      ASSIGN:      Este patio tem este tipo de linha originalmente
?.NumberOut False=
                Este patio tem este tipo de linha originalmente
?.NumberOut False + 1:NEXT(531$)
                Model statements for module: Assign 419
545$      ASSIGN:      marca=tnow:NEXT(557$)
                Model statements for module: Assign 532
557$      ASSIGN:      Temp2=A Produto * (A Produto <> 2) + 1 * (A Produto ==
2):NEXT(560$)
                Model statements for module: Decide 475
560$      BRANCH,      1:
                If,V CabFila(A Patio , Temp2) == 0,2283$,Yes:
                Else,2284$,Yes
2283$      ASSIGN:      Verifica se ja existe trem encabecendo a fila.NumberOut
True=
                Verifica se ja existe trem encabecendo a fila.NumberOut
True + 1:NEXT(561$)
2284$      ASSIGN:      Verifica se ja existe trem encabecendo a fila.NumberOut
False=
                Verifica se ja existe trem encabecendo a fila.NumberOut
False + 1:NEXT(561$)
                Model statements for module: Assign 537
561$      ASSIGN:      V CabFila (A Patio, Temp2)=A IDTrem:
                A CabFila=1:NEXT(525$)
                Model statements for module: Decide 186
525$      BRANCH,      1:
                If,
                V StatusPt (A Patio, A ColPatio) > 0 .AND. V
StatusPtProd (A Patio, Temp2 + (5*(A ColPatio==3))) < E PtProd(A Patio, Temp2 + (5*(A
ColPatio==3))),
                2285$,Yes:
                Else,2286$,Yes
2285$      ASSIGN:      Linha de carga ou descarga esta livre ?.NumberOut True=
                Linha de carga ou descarga esta livre ?.NumberOut True +
1:NEXT(546$)
2286$      ASSIGN:      Linha de carga ou descarga esta livre ?.NumberOut False=
                Linha de carga ou descarga esta livre ?.NumberOut False
+ 1:NEXT(564$)
                Model statements for module: Decide 346
546$      BRANCH,      1:
                If,A Entrou em Fila == 0,2287$,Yes:
                Else,2288$,Yes
2287$      ASSIGN:      Trem nao entrou em fila ?.NumberOut True=Trem nao entrou
em fila ?.NumberOut True + 1:NEXT(538$)
2288$      ASSIGN:      Trem nao entrou em fila ?.NumberOut False=Trem nao
entrou em fila ?.NumberOut False + 1:NEXT(556$)
                Model statements for module: Decide 266
538$      BRANCH,      1:
                If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2289$,Yes:
                Else,2290$,Yes
2289$      ASSIGN:      Ponto de checagem 14.NumberOut True=Ponto de checagem
14.NumberOut True + 1:NEXT(539$)
2290$      ASSIGN:      Ponto de checagem 14.NumberOut False=Ponto de checagem
14.NumberOut False + 1:NEXT(548$)
                Model statements for module: Separate 34
539$      DUPLICATE,      100 - 50:
                1,2293$,50:NEXT(2292$)

```

```

2292$      ASSIGN:      Separate 34.NumberOut Orig=Separate 34.NumberOut Orig +
1:NEXT(540$)
2293$      ASSIGN:      Separate 34.NumberOut Dup=Separate 34.NumberOut Dup +
1:NEXT(548$)
      Model statements for module:  Hold 43
540$      QUEUE,      Hold 43.Queue:DETACH
      Model statements for module:  Decide 356
548$      BRANCH,      1:
                          If,A LinhaOcupada <> 0,2294$,Yes:
                          Else,2295$,Yes
2294$      ASSIGN:      Trem ja estava ocupando uma linha de estacionamento
?.NumberOut True=
                          Trem ja estava ocupando uma linha de estacionamento
?.NumberOut True + 1:NEXT(549$)
2295$      ASSIGN:      Trem ja estava ocupando uma linha de estacionamento
?.NumberOut False=
                          Trem ja estava ocupando uma linha de estacionamento
?.NumberOut False + 1:NEXT(551$)
      Model statements for module:  Assign 427
549$      ASSIGN:      V StatusPt (A PtAtual, 1)=V StatusPt (A PtAtual, 1) + 1:
                          A LinhaOcupada=0:NEXT(551$)
      Model statements for module:  Decide 360
551$      BRANCH,      1:
                          If,A ColPatio == 1,2296$,Yes:
                          Else,2297$,Yes
2296$      ASSIGN:      Esta direcionado para linha de estacionamento
?.NumberOut True=
                          Esta direcionado para linha de estacionamento
?.NumberOut True + 1:NEXT(550$)
2297$      ASSIGN:      Esta direcionado para linha de estacionamento
?.NumberOut False=
                          Esta direcionado para linha de estacionamento
?.NumberOut False + 1:NEXT(558$)
      Model statements for module:  Assign 430
550$      ASSIGN:      A LinhaOcupada=A Patio:NEXT(558$)
      Model statements for module:  Decide 471
558$      BRANCH,      1:
                          If,A Tipo < 4,2298$,Yes:
                          Else,2299$,Yes
2298$      ASSIGN:      Trem tabela para ocupar terminal ?.NumberOut True=
                          Trem tabela para ocupar terminal ?.NumberOut True +
1:NEXT(526$)
2299$      ASSIGN:      Trem tabela para ocupar terminal ?.NumberOut False=
                          Trem tabela para ocupar terminal ?.NumberOut False +
1:NEXT(559$)
      Model statements for module:  Assign 234
526$      ASSIGN:      V StatusPt (A Patio, A ColPatio)=V StatusPt (A Patio, A
ColPatio) - 1:
                          Temp2=A Produto * (A Produto <> 2) + 1 * (A Produto ==
2):
                          V StatusPtProd (A Patio, Temp2 + (5*(A ColPatio==3)))=
                          V StatusPtProd (A Patio, Temp2 + (5*(A ColPatio==3))) +
1:
                          A Aviso=0:NEXT(562$)
      Model statements for module:  Assign 538
562$      ASSIGN:      V CabFila (A Patio, Temp2)=0:NEXT(528$)
      Model statements for module:  Assign 236
528$      ASSIGN:      A PtAtual=A PtAtual:
                          marca=tnow:
                          A FilaNode=0:NEXT(299$)
      Model statements for module:  Assign 536
559$      ASSIGN:      V StatusPt (A Patio, A ColPatio)=V StatusPt (A Patio, A
ColPatio) - 1:
                          A Aviso=0:NEXT(528$)
      Model statements for module:  Decide 435

```

```

556$          BRANCH,          1:
                                If, TNOW > (V Warmup * 1440), 2300$, Yes:
                                Else, 2301$, Yes
2300$          ASSIGN:          Warmup do log de eventos 3.NumberOut True=Warmup do log
de eventos 3.NumberOut True + 1:NEXT(566$)
2301$          ASSIGN:          Warmup do log de eventos 3.NumberOut False=Warmup do log
de eventos 3.NumberOut False + 1
                                :NEXT(516$)
                                Model statements for module: Decide 632
566$          BRANCH,          1:
                                If, V Gravar em texto == 1, 2302$, Yes:
                                Else, 2303$, Yes
2302$          ASSIGN:          Decide 632.NumberOut True=Decide 632.NumberOut True +
1:NEXT(567$)
2303$          ASSIGN:          Decide 632.NumberOut False=Decide 632.NumberOut False +
1:NEXT(515$)
                                Especificação do arquivo de LOG de eventos:
                                1 - Dia do evento
                                2- Identificador único do trem (A IDTrem)
                                3 - Número do trem (A NTrem)
                                4 - Tipo do trem (A Tipo)
                                5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
                                6 - Pátio (A PtAtual)
                                7 - Momento da saída do trem (TNOW)
                                8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
                                Model statements for module: ReadWrite 158
567$          WRITE,           Log de Eventos:
                                Dia,
                                A IDtrem,
                                A Ntrem,
                                A Tipo,
                                marca,
                                A PtAtual,
                                tnow,
                                9 * (A Estado ==1) + 8 * (A Estado == 0),
                                tnow - marca,
                                A Dist,
                                E FrotasT (A Ntrem, 1),
                                E FrotasT (A Ntrem, 2),
                                E FrotasT (A Ntrem, 3),
                                E FrotasT (A Ntrem, 4),
                                E FrotasT (A Ntrem, 5),
                                A Estado,
                                A Produto,
                                A Cliente,
                                A PtOrigem,
                                A PtDestino,
                                A PtZero:NEXT(516$)
                                Model statements for module: Assign 421
516$          ASSIGN:          A Entrou em Fila=0:NEXT(538$)
                                Especificação do arquivo de LOG de eventos:
                                1 - Dia do evento
                                2- Identificador único do trem (A IDTrem)
                                3 - Número do trem (A NTrem)
                                4 - Tipo do trem (A Tipo)
                                5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
                                6 - Pátio (A PtAtual)
                                7 - Momento da saída do trem (TNOW)
                                8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
                                Model statements for module: ReadWrite 65
515$          WRITE,           Log de MovimentacaoMDB, RECORDSET (LogdeEvento):

```

```

Dia,
A IDtrem,
A Ntrem,
A Tipo,
marca,
A PtAtual,
tnow,
9 * (A Estado ==1) + 8 * (A Estado == 0),
tnow - marca,
A Dist,
E FrotasT (A Ntrem, 1),
E FrotasT (A Ntrem, 2),
E FrotasT (A Ntrem, 3),
E FrotasT (A Ntrem, 4),
E FrotasT (A Ntrem, 5),
A Estado,
A Produto,
A Cliente,
A PtOrigem,
A PtDestino,
A PtZero:NEXT(516$)
Model statements for module: Decide 504
564$      BRANCH,      1:
                If,A FilaNode,2304$,Yes:
                Else,2305$,Yes
2304$      ASSIGN:      Verifica se esta no entroncamento sinalizando
                Verifica se esta no entroncamento sinalizando
                fila.NumberOut True + 1:NEXT(565$)
2305$      ASSIGN:      Verifica se esta no entroncamento sinalizando
                Verifica se esta no entroncamento sinalizando
                fila.NumberOut False + 1:NEXT(537$)
Model statements for module: Assign 603
565$      ASSIGN:      V NodeD ( V Trecho (V ROT(A Indirota + (A Sentido ==0),
A Rota),1) , 1 + (A Sentido==0))=
                A PtDestino:
                A ParEsp=A PtAtual:
                Sh5=A IDTrem:
                A Entrou em Fila=1:
                V NodeDP ( V Trecho (V ROT(A Indirota + (A Sentido ==0),
A Rota),1) , 1 + (A Sentido==0))=Temp2
                :NEXT(555$)
Model statements for module: Decide 391
555$      BRANCH,      1:
                If,A Aviso == 0,2306$,Yes:
                Else,2307$,Yes
2306$      ASSIGN:      Nao foi avisado ?.NumberOut True=Nao foi avisado
                ?.NumberOut True + 1:NEXT(552$)
2307$      ASSIGN:      Nao foi avisado ?.NumberOut False=Nao foi avisado
                ?.NumberOut False + 1:NEXT(554$)
Model statements for module: Decide 372
552$      BRANCH,      1:
                If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2308$,Yes:
                Else,2309$,Yes
2308$      ASSIGN:      Ponto de checagem 24.NumberOut True=Ponto de checagem
                24.NumberOut True + 1:NEXT(536$)
2309$      ASSIGN:      Ponto de checagem 24.NumberOut False=Ponto de checagem
                24.NumberOut False + 1:NEXT(536$)
Model statements for module: Delay 32
536$      DELAY:      norm(1,0.4),,Other:NEXT(525$)
Model statements for module: Decide 390
554$      BRANCH,      1:

```

```

                                If,V Destino ( A Aviso,1 + (A Sentido==0) ) <> A
PtDestino,2310$,Yes:
                                Else,2311$,Yes
2310$          ASSIGN:          Fila andou para este trem?.NumberOut True=Fila andou
para este trem?.NumberOut True + 1:NEXT(563$)
2311$          ASSIGN:          Fila andou para este trem?.NumberOut False=Fila andou
para este trem?.NumberOut False + 1
                                :NEXT(552$)
                                Model statements for module: Assign 539
563$          ASSIGN:          V CabFila (A Patio, Temp2)=0:
                                A CabFila=0:NEXT(283$)
                                Model statements for module: Assign 318
537$          ASSIGN:          V Destino ( A PtAtual,1 + (A Sentido==0))=A PtDestino:
                                A ParEsp=A PtAtual:
                                Sh5=A IDTrem:
                                A Entrou em Fila=1:
                                V DestinoP ( A PtAtual,1 + (A
Sentido==0))=Temp2:NEXT(555$)
                                Model statements for module: Assign 237
531$          ASSIGN:          V ocpt ( A PtAtual, 1 + (A Sentido==0) )=0:NEXT(532$)
                                Model statements for module: Signal 16
532$          SIGNAL:          A IDtrem,1:NEXT(533$)
                                Model statements for module: ReadWrite 32
533$          WRITE,          Log de Erros,
                                "%1.8g. ERR8 Erro no trem %1.8g. Roteiro %1.8g. Tipo
%1.8g. Patio %1.8g nao tem a linha requisitada para a atividade %1.8g. Trem abortado
no patio %1.8g.\n":
                                TNOW,
                                A IDtrem,
                                A Rota,
                                A Tipo,
                                A Patio,
                                A ColPatio,
                                A PtAtual:NEXT(542$)
                                Model statements for module: Dispose 52
542$          ASSIGN:          Dispose 52.NumberOut=Dispose 52.NumberOut + 1
2312$         DISPOSE:          Yes
                                Model statements for module: Assign 233
524$          ASSIGN:          A ColPatio=3:NEXT(530$)
                                Model statements for module: Decide 175
517$          BRANCH,          1:
                                If,A SouHelper == 1,528$,Yes:
                                If,A Tipo == 4,519$,Yes:
                                If,A Tipo == 5,520$,Yes:
                                Else,518$,Yes
                                Model statements for module: Search 2
518$          FINDJ,          1,50:E Atividades ( J , 1) == A Patio
2315$         BRANCH,          1:
                                If,J<>0,2316$,Yes:
                                Else,2317$,Yes
2316$         DELAY:          0.0,,VA:NEXT(535$)
2317$         DELAY:          0.0,,VA:NEXT(528$)
                                Model statements for module: Assign 239
535$          ASSIGN:          A Escolha=J:NEXT(544$)
                                Model statements for module: Decide 331
544$          BRANCH,          1:
                                If,V EstAtiv (A Escolha, (A Estado==0)+1) ==
1,2318$,Yes:
                                Else,2319$,Yes
2318$         ASSIGN:          Trem neste estado de carga realiza atividade neste patio
?.NumberOut True=
                                Trem neste estado de carga realiza atividade neste patio
?.NumberOut True + 1:NEXT(527$)
2319$         ASSIGN:          Trem neste estado de carga realiza atividade neste patio
?.NumberOut False=

```

```

Trem neste estado de carga realiza atividade neste patio
?.NumberOut False + 1:NEXT(528$)
  Model statements for module: Assign 235
527$    ASSIGN:    A ColPatio=1:NEXT(534$)
  Model statements for module: Decide 189
534$    BRANCH,    1:
                    If,E Patios (A Patio, A ColPatio) > 0,2320$,Yes:
                    Else,2321$,Yes
2320$    ASSIGN:    Este patio tem linha de atividade originalmente
?.NumberOut True=
                    Este patio tem linha de atividade originalmente
?.NumberOut True + 1:NEXT(529$)
2321$    ASSIGN:    Este patio tem linha de atividade originalmente
?.NumberOut False=
                    Este patio tem linha de atividade originalmente
?.NumberOut False + 1:NEXT(528$)
  Model statements for module: Decide 187
529$    BRANCH,    1:
                    If,V StatusPt (A Patio, A ColPatio) > 0,2322$,Yes:
                    Else,2323$,Yes
2322$    ASSIGN:    Linha de atividade ou espera esta livre ?.NumberOut
True=
                    Linha de atividade ou espera esta livre ?.NumberOut True
+ 1:NEXT(547$)
2323$    ASSIGN:    Linha de atividade ou espera esta livre ?.NumberOut
False=
                    Linha de atividade ou espera esta livre ?.NumberOut
False + 1:NEXT(553$)
  Model statements for module: Decide 350
547$    BRANCH,    1:
                    If,A Tipo < 4,2324$,Yes:
                    Else,2325$,Yes
2324$    ASSIGN:    Trens tabela devem gravar no log.NumberOut True=Trens
tabela devem gravar no log.NumberOut True + 1
                    :NEXT(546$)
2325$    ASSIGN:    Trens tabela devem gravar no log.NumberOut False=
                    Trens tabela devem gravar no log.NumberOut False +
1:NEXT(538$)
  Model statements for module: Assign 443
553$    ASSIGN:    A DesistiuLinha=1:NEXT(528$)
  Model statements for module: Decide 176
519$    BRANCH,    1:
                    If,E PtGrade (A Ntrem, A IndiAtiv) == A Patio,2326$,Yes:
                    Else,2327$,Yes
2326$    ASSIGN:    Trem grade tem atividade neste patio ?.NumberOut True=
                    Trem grade tem atividade neste patio ?.NumberOut True +
1:NEXT(527$)
2327$    ASSIGN:    Trem grade tem atividade neste patio ?.NumberOut False=
                    Trem grade tem atividade neste patio ?.NumberOut False +
1:NEXT(528$)
  Model statements for module: Decide 177
520$    BRANCH,    1:
                    If,E PtExpresso (A Ntrem, A IndiAtiv) == A
Patio,2328$,Yes:
                    Else,2329$,Yes
2328$    ASSIGN:    Trem expresso tem atividade neste patio ?.NumberOut
True=
                    Trem expresso tem atividade neste patio ?.NumberOut True
+ 1:NEXT(527$)
2329$    ASSIGN:    Trem expresso tem atividade neste patio ?.NumberOut
False=
                    Trem expresso tem atividade neste patio ?.NumberOut
False + 1:NEXT(528$)
  Model statements for module: Assign 648
299$    ASSIGN:    A aTemp3=A aTemp3:NEXT(20$)

```

```

Indice2: Atributo que irá contar quantas linhas de A Stop já
foram lidas, sendo que o número máximo de linhas a serem
verificadas está registrado no atributo Indice.
Model statements for module: Assign 75
20$      ASSIGN:      Indice2=1:NEXT(195$)
Model statements for module: Seize 5
195$     QUEUE,      SEIZE Passo 3.Queue
          SEIZE,      2,Other:
                          Passo3,1:NEXT(2331$)
2331$     DELAY:      0.0,,VA:NEXT(80$)
Model statements for module: Decide 92
80$      BRANCH,      1:
          If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2332$,Yes:
          Else,2333$,Yes
2332$     ASSIGN:      Ponto de checagem 2.NumberOut True=Ponto de checagem
2.NumberOut True + 1:NEXT(81$)
2333$     ASSIGN:      Ponto de checagem 2.NumberOut False=Ponto de checagem
2.NumberOut False + 1:NEXT(79$)
Model statements for module: Separate 7
81$      DUPLICATE,   100 - 50:
          1,2336$,50:NEXT(2335$)
2335$     ASSIGN:      Separate 7.NumberOut Orig=Separate 7.NumberOut Orig +
1:NEXT(82$)
2336$     ASSIGN:      Separate 7.NumberOut Dup=Separate 7.NumberOut Dup +
1:NEXT(79$)
Model statements for module: Hold 14
82$      QUEUE,      Hold VERIFICACAO.Queue:DETACH
          1 = CRESCENTE
          Próximo Pátio: Indirota+1
          Trecho usado: Indirota
          0 = DECRESCENTE
          Próximo Pátio: Indirota+1
          Trecho usado: Indirota+1
Model statements for module: Assign 114
79$      ASSIGN:      A TmpSentido=X Sentido(Indice2):
          A TmpTrecho=X trecho(Indice2):
          A Patio=X Patio(Indice2):
          X Node(Indice2)=0:
          X NodeS(Indice2)=0:NEXT(22$)
Model statements for module: Decide 59
22$      BRANCH,      1:
          If,
          V Sentido (A TmpTrecho) == 2 .OR. V Sentido (A
TmpTrecho) == A TmpSentido .OR. V Trecho (A TmpTrecho,5)>1,
          2337$,Yes:
          Else,2338$,Yes
2337$     ASSIGN:      Sentido deste trecho liberado ou a favor ?.NumberOut
True=
          Sentido deste trecho liberado ou a favor ?.NumberOut
True + 1:NEXT(24$)
2338$     ASSIGN:      Sentido deste trecho liberado ou a favor ?.NumberOut
False=
          Sentido deste trecho liberado ou a favor ?.NumberOut
False + 1:NEXT(185$)
Model statements for module: Decide 62
24$      BRANCH,      1:
          If,V oclb ( A TmpTrecho , 1 + ( 4 * (V Trecho(A
TmpTrecho,5) == 1) * ( A TmpSentido == 0) ) ) == 0,
          2339$,Yes:
          Else,2340$,Yes
2339$     ASSIGN:      Linha livre ?.NumberOut True=Linha livre ?.NumberOut
True + 1:NEXT(23$)
2340$     ASSIGN:      Linha livre ?.NumberOut False=Linha livre ?.NumberOut
False + 1:NEXT(186$)

```



```

Model statements for module: Decide 60
23$      BRANCH,      1:
                If,V Trecho (A TmpTrecho , 3) == 1,2341$,Yes:
                Else,2342$,Yes
2341$      ASSIGN:      Linha sinalizada ?.NumberOut True=Linha sinalizada
?.NumberOut True + 1:NEXT(35$)
2342$      ASSIGN:      Linha sinalizada ?.NumberOut False=Linha sinalizada
?.NumberOut False + 1:NEXT(88$)
Model statements for module: Decide 72
35$      BRANCH,      1:
                If,A Restricao == 1 .AND. V Trecho(A TmpTrecho,5) ==
1,2343$,Yes:
                Else,2344$,Yes
2343$      ASSIGN:      Trem com restricao de parada ?.NumberOut True=Trem com
restricao de parada ?.NumberOut True + 1
                :NEXT(302$)
2344$      ASSIGN:      Trem com restricao de parada ?.NumberOut False=Trem com
restricao de parada ?.NumberOut False + 1
                :NEXT(190$)
Model statements for module: Decide 577
302$      BRANCH,      1:
                If,V Sentido (A TmpTrecho) == A TmpSentido,2345$,Yes:
                Else,2346$,Yes
2345$      ASSIGN:      Linha restritiva ja esta reservada para o mesmo sentido
?.NumberOut True=
                Linha restritiva ja esta reservada para o mesmo sentido
?.NumberOut True + 1:NEXT(88$)
2346$      ASSIGN:      Linha restritiva ja esta reservada para o mesmo sentido
?.NumberOut False=
                Linha restritiva ja esta reservada para o mesmo sentido
?.NumberOut False + 1:NEXT(36$)
Model statements for module: Decide 104
88$      BRANCH,      1:
                If,E Patios (A Patio , 6 ) == 1 .AND. A
Restricao==0,2347$,Yes:
                Else,2348$,Yes
2347$      ASSIGN:      Este patio esta ATIVADO e trem sem restricao ?.NumberOut
True=
                Este patio esta ATIVADO e trem sem restricao ?.NumberOut
True + 1:NEXT(655$)
2348$      ASSIGN:      Este patio esta ATIVADO e trem sem restricao ?.NumberOut
False=
                Este patio esta ATIVADO e trem sem restricao ?.NumberOut
False + 1:NEXT(25$)
Model statements for module: Decide 508
655$      BRANCH,      1:
                If,A Patio == A PtDestino,2349$,Yes:
                Else,2350$,Yes
2349$      ASSIGN:      Este patio sera o destino final ?.NumberOut True=
                Este patio sera o destino final ?.NumberOut True +
1:NEXT(664$)
2350$      ASSIGN:      Este patio sera o destino final ?.NumberOut False=
                Este patio sera o destino final ?.NumberOut False +
1:NEXT(660$)
Model statements for module: Decide 611
664$      BRANCH,      1:
                If,A Tipo <> 2 .OR. (A Patio <> 151 .AND. A Patio <> 7
.AND. A Patio <> 148),2351$,Yes:
                Else,2352$,Yes
2351$      ASSIGN:      Nao esta destinado a ponto zero ?.NumberOut True=
                Nao esta destinado a ponto zero ?.NumberOut True +
1:NEXT(656$)
2352$      ASSIGN:      Nao esta destinado a ponto zero ?.NumberOut False=
                Nao esta destinado a ponto zero ?.NumberOut False +
1:NEXT(660$)

```

```

Model statements for module: Assign 604
656$      ASSIGN:      A Direto=1:NEXT(25$)
Model statements for module: Decide 546
660$      BRANCH,      1:
                If,E Verifica entroncamento A Patio .AND. V Ativa
Entroncamento .AND. 1,2353$,Yes:
                Else,2354$,Yes
2353$      ASSIGN:      Identifica se esta no patio de entroncamento.NumberOut
True=
                Identifica se esta no patio de entroncamento.NumberOut
True + 1:NEXT(663$)
2354$      ASSIGN:      Identifica se esta no patio de entroncamento.NumberOut
False=
                Identifica se esta no patio de entroncamento.NumberOut
False + 1:NEXT(654$)
Model statements for module: Decide 600
663$      BRANCH,      1:
                If,V DROT ((A Indirota + Indice2) , A Rota) ==
1,2355$,Yes:
                Else,2356$,Yes
2355$      ASSIGN:      Identifica o sentido do trecho.NumberOut True=Identifica
o sentido do trecho.NumberOut True + 1
                :NEXT(653$)
2356$      ASSIGN:      Identifica o sentido do trecho.NumberOut
False=Identifica o sentido do trecho.NumberOut False + 1
                :NEXT(662$)
Model statements for module: Decide 505
653$      BRANCH,      1:
                If,(V Node ( V Trecho(V ROT ((A Indirota + Indice2) , A
Rota) ,1), 1) == 1).AND.0,2357$,Yes:
                Else,2358$,Yes
2357$      ASSIGN:      Caminho crescente do entroncamento esta ocupado
?.NumberOut True=
                Caminho crescente do entroncamento esta ocupado
?.NumberOut True + 1:NEXT(659$)
2358$      ASSIGN:      Caminho crescente do entroncamento esta ocupado
?.NumberOut False=
                Caminho crescente do entroncamento esta ocupado
?.NumberOut False + 1:NEXT(661$)
Model statements for module: Decide 531
659$      BRANCH,      1:
                If,A IDTrem <> Rastreio,2359$,Yes:
                Else,2360$,Yes
2359$      ASSIGN:      Nao atende patio de entroncamento crescente.NumberOut
True=
                Nao atende patio de entroncamento crescente.NumberOut
True + 1:NEXT(189$)
2360$      ASSIGN:      Nao atende patio de entroncamento crescente.NumberOut
False=
                Nao atende patio de entroncamento crescente.NumberOut
False + 1:NEXT(189$)
Model statements for module: Assign 661
661$      ASSIGN:      X Node(Indice2)=V Trecho(V ROT ((A Indirota + Indice2) ,
A Rota) ,1):
                X NodeS(Indice2)=V DROT ((A Indirota + Indice2) , A
Rota):NEXT(25$)
Model statements for module: Decide 599
662$      BRANCH,      1:
                If,V Node ( V Trecho(V ROT ((A Indirota + Indice2) , A
Rota) ,1), 2) == 1,2361$,Yes:
                Else,2362$,Yes
2361$      ASSIGN:      Caminho decrescente do entroncamento esta ocupado
?.NumberOut True=
                Caminho decrescente do entroncamento esta ocupado
?.NumberOut True + 1:NEXT(665$)

```

```

2362$      ASSIGN:      Caminho decrescente do entroncamento esta ocupado
?.NumberOut False=
Caminho decrescente do entroncamento esta ocupado
?.NumberOut False + 1:NEXT(661$)
  Model statements for module:  Decide 681
665$      BRANCH,      1:
                          If,A IDTrem <> Rastreo,2363$,Yes:
                          Else,2364$,Yes
2363$      ASSIGN:      Nao atende patio de entroncamento decrescente.NumberOut
True=
Nao atende patio de entroncamento decrescente.NumberOut
True + 1:NEXT(189$)
2364$      ASSIGN:      Nao atende patio de entroncamento decrescente.NumberOut
False=
Nao atende patio de entroncamento decrescente.NumberOut
False + 1:NEXT(189$)
  Model statements for module:  Decide 507
654$      BRANCH,      1:
                          If,V ocpt ( A Patio , 1+ ( A TmpSentido == 0 ) ) ==
0,2365$,Yes:
                          Else,2366$,Yes
2365$      ASSIGN:      Patio esta livre ?.NumberOut True=Patio esta livre
?.NumberOut True + 1:NEXT(666$)
2366$      ASSIGN:      Patio esta livre ?.NumberOut False=Patio esta livre
?.NumberOut False + 1:NEXT(657$)
  Model statements for module:  Decide 684
666$      BRANCH,      1:
                          If,
                          V PrioPatio( A Patio , 2 - ( A TmpSentido == 0 ) ) == 0
.OR. V PrioPatio( A Patio , 2 - ( A TmpSentido == 0 ) ) > A Prioridade .OR. A
Prioridade2==1,
                          2367$,Yes:
                          Else,2368$,Yes
2367$      ASSIGN:      Patio no sentido inverso tem prioridade menor que deste
trem ou esta vazio?.NumberOut True=
Patio no sentido inverso tem prioridade menor que deste
trem ou esta vazio?.NumberOut True + 1
:NEXT(668$)
2368$      ASSIGN:      Patio no sentido inverso tem prioridade menor que deste
trem ou esta vazio?.NumberOut False=
Patio no sentido inverso tem prioridade menor que deste
trem ou esta vazio?.NumberOut False + 1
:NEXT(667$)
  Model statements for module:  Assign 743
668$      ASSIGN:      A Prioridade2=0:NEXT(25$)
  Model statements for module:  Assign 742
667$      ASSIGN:      A Prioridade2=1:NEXT(189$)
  Model statements for module:  Decide 509
657$      BRANCH,      1:
                          If,V Destino ( A Patio,1 + ( A TmpSentido==0) ) ==
0,2369$,Yes:
                          Else,2370$,Yes
2369$      ASSIGN:      Patio em teste nao esta ocupado por trem aguardando
terminal ?.NumberOut True=
Patio em teste nao esta ocupado por trem aguardando
terminal ?.NumberOut True + 1:NEXT(189$)
2370$      ASSIGN:      Patio em teste nao esta ocupado por trem aguardando
terminal ?.NumberOut False=
Patio em teste nao esta ocupado por trem aguardando
terminal ?.NumberOut False + 1:NEXT(658$)
  Model statements for module:  Decide 510
658$      BRANCH,      1:
                          If,V ocpt ( A Patio , 2 - ( A TmpSentido == 0 ) ) ==
0,2371$,Yes:
                          Else,2372$,Yes

```

```

2371$      ASSIGN:      Patio no sentido inverso esta livre ?.NumberOut True=
                                Patio no sentido inverso esta livre ?.NumberOut True +
1:NEXT(25$)
2372$      ASSIGN:      Patio no sentido inverso esta livre ?.NumberOut False=
                                Patio no sentido inverso esta livre ?.NumberOut False +
1:NEXT(189$)
    Model statements for module:  Decide 63
25$        BRANCH,      1:
                                If,Indice2 >= Indice,2373$,Yes:
                                Else,2374$,Yes
2373$      ASSIGN:      Verificou todo o percurso ?.NumberOut True=Verificou
                                todo o percurso ?.NumberOut True + 1:NEXT(28$)
2374$      ASSIGN:      Verificou todo o percurso ?.NumberOut False=Verificou
                                todo o percurso ?.NumberOut False + 1
                                :NEXT(26$)
    Indice2: Atributo que irá contar quantas linhas de A Stop já
    foram lidas, sendo que o número máximo de linhas a serem
    verificadas está registrado no atributo Indice.
    Model statements for module:  Assign 78
28$        ASSIGN:      Indice2=1:
                                Indicel=A Indirota:
                                A Excecao=0:
                                A RestFila=0:
                                A Repescagem=0:NEXT(681$)
    Model statements for module:  Decide 512
681$        BRANCH,      1:
                                If,A Node <> 0,2375$,Yes:
                                Else,2376$,Yes
2375$      ASSIGN:      Saindo de entroncamento ?.NumberOut True=Saindo de
                                entroncamento ?.NumberOut True + 1:NEXT(688$)
2376$      ASSIGN:      Saindo de entroncamento ?.NumberOut False=Saindo de
                                entroncamento ?.NumberOut False + 1:NEXT(113$)
    Model statements for module:  Decide 606
688$        BRANCH,      1:
                                If,chechar >= 0 .AND. chechar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2377$,Yes:
                                Else,2378$,Yes
2377$      ASSIGN:      Ponto de checagem 36.NumberOut True=Ponto de checagem
                                36.NumberOut True + 1:NEXT(689$)
2378$      ASSIGN:      Ponto de checagem 36.NumberOut False=Ponto de checagem
                                36.NumberOut False + 1:NEXT(683$)
    Model statements for module:  Separate 73
689$        DUPLICATE,    100 - 50:
                                1,2381$,50:NEXT(2380$)
2380$      ASSIGN:      Separate 73.NumberOut Orig=Separate 73.NumberOut Orig +
1:NEXT(690$)
2381$      ASSIGN:      Separate 73.NumberOut Dup=Separate 73.NumberOut Dup +
1:NEXT(683$)
    Model statements for module:  Hold 86
690$        QUEUE,        Hold 86.Queue:DETACH
    Model statements for module:  Decide 527
683$        BRANCH,      1:
                                If,A NodeS == 1,685$,Yes:
                                If,A NodeS == 2,686$,Yes:
                                Else,687$,Yes
    Model statements for module:  Dispose 75
687$      ASSIGN:      Dispose 75.NumberOut=Dispose 75.NumberOut + 1
2384$      DISPOSE:      Yes
    Model statements for module:  Record 7
685$      COUNT:        Record 7,A IDTrem == Rastreio:NEXT(682$)
    Model statements for module:  Assign 617
682$      ASSIGN:      V Node ( A Node , 1)=0:
                                A Node=0:
                                A NodeS=0:NEXT(113$)
    Model statements for module:  Record 8

```

```

686$          COUNT:          Record 8,A IDTrem == Rastreio:NEXT(684$)
Model statements for module: Assign 618
684$          ASSIGN:         V Node ( A Node , 2)=0:
                                A Node=0:
                                A NodeS=0:NEXT(113$)
Indice2: Atributo que irá contar quantas linhas de A Stop já
foram lidas, sendo que o número máximo de linhas a serem
verificadas está registrado no atributo Indice.
Model statements for module: Assign 147
113$          ASSIGN:         A TmpSentido=X Sentido(Indice2):
                                A TmpTrecho=X Trecho(Indice2):
                                A Patio=X Patio(Indice2):
                                Temp1=1+(A TmpSentido==0):NEXT(64$)
Model statements for module: Decide 86
64$           BRANCH,        1:
                                If, checar >= 0 .AND. checar < TNOW.OR. (A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2385$,Yes:
                                Else,2386$,Yes
2385$          ASSIGN:         Ponto de checagem 1.NumberOut True=Ponto de checagem
1.NumberOut True + 1:NEXT(65$)
2386$          ASSIGN:         Ponto de checagem 1.NumberOut False=Ponto de checagem
1.NumberOut False + 1:NEXT(151$)
Model statements for module: Separate 6
65$           DUPLICATE,     100 - 50:
                                1,2389$,50:NEXT(2388$)
2388$          ASSIGN:         Separate 6.NumberOut Orig=Separate 6.NumberOut Orig +
1:NEXT(66$)
2389$          ASSIGN:         Separate 6.NumberOut Dup=Separate 6.NumberOut Dup +
1:NEXT(151$)
Model statements for module: Hold 12
66$           QUEUE,         Hold 12.Queue:DETACH
Model statements for module: Decide 229
151$          BRANCH,        1:
                                If,V Trecho(A TmpTrecho,5)>1 .AND. A
Restricao==0,2390$,Yes:
                                Else,2391$,Yes
2390$          ASSIGN:         Linha dupla e trem nao restritivo ?.NumberOut True=
Linha dupla e trem nao restritivo ?.NumberOut True +
1:NEXT(360$)
2391$          ASSIGN:         Linha dupla e trem nao restritivo ?.NumberOut False=
Linha dupla e trem nao restritivo ?.NumberOut False +
1:NEXT(326$)
Model statements for module: Assign 345
360$          ASSIGN:         V ocbl ( A TmpTrecho , 1+ ( 4 * ( A TmpSentido == 0))
)=1:
                                V CronoPPT(A TmpTrecho,Temp1)=tnow:NEXT(29$)
Model statements for module: Decide 67
29$           BRANCH,        1:
                                If,V Trecho ( A TmpTrecho , 3) == 1,2392$,Yes:
                                Else,2393$,Yes
2392$          ASSIGN:         2 Linha sinalizada ?.NumberOut True=2 Linha sinalizada
?.NumberOut True + 1:NEXT(27$)
2393$          ASSIGN:         2 Linha sinalizada ?.NumberOut False=2 Linha sinalizada
?.NumberOut False + 1:NEXT(89$)
Model statements for module: Decide 65
27$           BRANCH,        1:
                                If,A Restricao == 0,2394$,Yes:
                                Else,2395$,Yes
2394$          ASSIGN:         2 Trem nao tem nenhuma restricao de parada ?.NumberOut
True=
                                2 Trem nao tem nenhuma restricao de parada ?.NumberOut
True + 1:NEXT(108$)
2395$          ASSIGN:         2 Trem nao tem nenhuma restricao de parada ?.NumberOut
False=

```

```

                2 Trem nao tem nenhuma restricao de parada ?.NumberOut
False + 1:NEXT(303$)
    Model statements for module:  Release 9
108$      RELEASE:      Passo3,1:NEXT(270$)
    Model statements for module:  Assign 544
270$      ASSIGN:      Temp2=A Produto * (A Produto <> 2) + 1 * (A Produto ==
2):NEXT(675$)
    Model statements for module:  Decide 248
675$      BRANCH,      1:
                        If,A ParEsp == 0,2396$,Yes:
                        Else,2397$,Yes
2396$      ASSIGN:      V Destino nao esta sendo usada por este trem ?.NumberOut
True=
                        V Destino nao esta sendo usada por este trem ?.NumberOut
True + 1:NEXT(120$)
2397$      ASSIGN:      V Destino nao esta sendo usada por este trem ?.NumberOut
False=
                        V Destino nao esta sendo usada por este trem ?.NumberOut
False + 1:NEXT(678$)
    Model statements for module:  Decide 477
678$      BRANCH,      1:
                        If,A CabFila == 1,2398$,Yes:
                        Else,2399$,Yes
2398$      ASSIGN:      Trem encabeca fila ?.NumberOut True=Trem encabeca fila
?.NumberOut True + 1:NEXT(676$)
2399$      ASSIGN:      Trem encabeca fila ?.NumberOut False=Trem encabeca fila
?.NumberOut False + 1:NEXT(677$)
    Model statements for module:  Assign 328
676$      ASSIGN:      V Destino ( A PtAtual,1 + (A Sentido==0))=0:
                        V DestinoP ( A PtAtual,1 + (A Sentido==0))=0:
                        A ParEsp=0:
                        A CabFila=0:NEXT(680$)
    Model statements for module:  Decide 518
680$      BRANCH,      1:
                        If,E Verifica entroncamento,2400$,Yes:
                        Else,2401$,Yes
2400$      ASSIGN:      Saindo de fila de entroncamento para SB?.NumberOut True=
Saindo de fila de entroncamento para SB?.NumberOut True
+ 1:NEXT(679$)
2401$      ASSIGN:      Saindo de fila de entroncamento para SB?.NumberOut
False=
                        Saindo de fila de entroncamento para SB?.NumberOut False
+ 1:NEXT(120$)
    Model statements for module:  Assign 610
679$      ASSIGN:      V NodeD ( V Trecho (V ROT(A Indirota + (A Sentido ==0),
A Rota),1) , 1 + (A Sentido==0))=0:
                        V NodeDP ( V Trecho (V ROT(A Indirota + (A Sentido ==0),
A Rota),1) , 1 + (A Sentido==0))=0
                        :NEXT(120$)
    Model statements for module:  Assign 540
677$      ASSIGN:      V Destino ( A PtAtual,1 + (A Sentido==0))=A PtDestino:
                        V DestinoP ( A PtAtual,1 + (A
Sentido==0))=Temp2:NEXT(676$)
    Model statements for module:  Assign 162
120$      ASSIGN:      A OldSentido=1 + (A TmpSentido==0):NEXT(224$)
    Model statements for module:  Decide 358
224$      BRANCH,      1:
                        If,A LinhaOcupada == A PtAtual,2402$,Yes:
                        Else,2403$,Yes
2402$      ASSIGN:      Trem esta ocupando uma linha de estacionamento
?.NumberOut True=
                        Trem esta ocupando uma linha de estacionamento
?.NumberOut True + 1:NEXT(225$)
2403$      ASSIGN:      Trem esta ocupando uma linha de estacionamento
?.NumberOut False=

```

```

Trem esta ocupando uma linha de estacionamento
?.NumberOut False + 1:NEXT(265$)
  Model statements for module: Assign 428
225$      ASSIGN:      V StatusPt (A PtAtual, 1)=V StatusPt (A PtAtual, 1) + 1:
              A LinhaOcupada=0:NEXT(265$)
  Model statements for module: Decide 432
265$      BRANCH,      1:
              If, TNOW > (V Warmup * 1440), 2404$, Yes:
              Else, 2405$, Yes
2404$      ASSIGN:      Warmup do log de movimentacao 2.NumberOut True=Warmup do
log de movimentacao 2.NumberOut True + 1
              :NEXT(221$)
2405$      ASSIGN:      Warmup do log de movimentacao 2.NumberOut False=Warmup
do log de movimentacao 2.NumberOut False + 1
              :NEXT(334$)
  Model statements for module: Decide 348
221$      BRANCH,      1:
              If, (tnow - marca) < 0.001, 2406$, Yes:
              Else, 2407$, Yes
2406$      ASSIGN:      Nao houve espera neste trem por problemas de transito
?.NumberOut True=
              Nao houve espera neste trem por problemas de transito
?.NumberOut True + 1:NEXT(334$)
2407$      ASSIGN:      Nao houve espera neste trem por problemas de transito
?.NumberOut False=
              Nao houve espera neste trem por problemas de transito
?.NumberOut False + 1:NEXT(353$)
  Model statements for module: Decide 607
334$      BRANCH,      1:
              If, A Restricao == 0, 2408$, Yes:
              Else, 2409$, Yes
2408$      ASSIGN:      Trem sem restricao ?.NumberOut True=Trem sem restricao
?.NumberOut True + 1:NEXT(333$)
2409$      ASSIGN:      Trem sem restricao ?.NumberOut False=Trem sem restricao
?.NumberOut False + 1:NEXT(332$)
  Model statements for module: Assign 706
333$      ASSIGN:      Temp1=1*(A Tipo<4) + 2*(A Tipo==4) + 3*(A
Tipo==5):NEXT(37$)
  Model statements for module: Signal 7
37$      SIGNAL:      A IDtrem, 1:NEXT(103$)
  Model statements for module: Decide 110
103$      BRANCH,      1:
              With, (( V Trecho( A TmpTrecho , 6 + Temp1)
)) / 100, 2410$, Yes:
              Else, 2411$, Yes
2410$      ASSIGN:      Este movimento em SBs tera contingencia de via
?.NumberOut True=
              Este movimento em SBs tera contingencia de via
?.NumberOut True + 1:NEXT(104$)
2411$      ASSIGN:      Este movimento em SBs tera contingencia de via
?.NumberOut False=
              Este movimento em SBs tera contingencia de via
?.NumberOut False + 1:NEXT(102$)
  Model statements for module: Assign 136
104$      ASSIGN:      A Tmp=((1 + (A TmpSentido==0)) * Temp1 ) + 6:
              A TransitTime=TT Trechos ( A TmpTrecho , A
Tmp):NEXT(582$)
  Model statements for module: Decide 370
582$      BRANCH,      1:
              If, A TransitTime <= 180, 2412$, Yes:
              Else, 2413$, Yes
2412$      ASSIGN:      Decide 370.NumberOut True=Decide 370.NumberOut True +
1:NEXT(718$)
2413$      ASSIGN:      Decide 370.NumberOut False=Decide 370.NumberOut False +
1:NEXT(583$)

```

```

Model statements for module: Decide 371
583$      BRANCH,      1:
                If,A Tmp>6,2414$,Yes:
                Else,2415$,Yes
2414$      ASSIGN:      Decide 371.NumberOut True=Decide 371.NumberOut True +
1:NEXT(584$)
2415$      ASSIGN:      Decide 371.NumberOut False=Decide 371.NumberOut False +
1:NEXT(585$)
Model statements for module: ReadWrite 72
584$      WRITE,      Log de Erros,"%1.8g. TT CONTINGENCIA de %1.8g. Trem tipo
%1.8g. Trecho %1.8g. Coluna %1.8g.\n":
                TNOW,
                A TransitTime,
                A Tipo,
                A TmpTrecho,
                A Tmp - 6:NEXT(586$)
Model statements for module: Assign 441
586$      ASSIGN:      A TransitTime=180:NEXT(718$)
Model statements for module: ReadWrite 73
585$      WRITE,      Log de Erros,"%1.8g. TT NORMAL de %1.8g. Trem tipo
%1.8g. Trecho %1.8g. Coluna %1.8g.\n":
                TNOW,
                A TransitTime,
                A Tipo,
                A TmpTrecho,
                A Tmp:NEXT(586$)
Model statements for module: Decide 513
718$      BRANCH,      1:
                If,A Node <> 0,2416$,Yes:
                Else,2417$,Yes
2416$      ASSIGN:      Saindo de entroncamento para SB?.NumberOut True=Saindo
de entroncamento para SB?.NumberOut True + 1
                :NEXT(720$)
2417$      ASSIGN:      Saindo de entroncamento para SB?.NumberOut False=
Saindo de entroncamento para SB?.NumberOut False +
1:NEXT(212$)
Model statements for module: Decide 536
720$      BRANCH,      1:
                If,A NodeS == 1,722$,Yes:
                If,A nodeS == 2,723$,Yes:
                Else,724$,Yes
Model statements for module: Dispose 78
724$      ASSIGN:      Dispose 78.NumberOut=Dispose 78.NumberOut + 1
2420$      DISPOSE:      Yes
Model statements for module: Record 15
722$      COUNT:      Record 15,A IDTrem == Rastreio:NEXT(719$)
Model statements for module: Assign 625
719$      ASSIGN:      V Node ( A Node, 1)=0:
                A Node=0:
                A NodeS=0:NEXT(212$)
Model statements for module: Record 16
723$      COUNT:      Record 16,A IDTrem == Rastreio:NEXT(721$)
Model statements for module: Assign 626
721$      ASSIGN:      V Node ( A Node, 2)=0:
                A Node=0:
                A NodeS=0:NEXT(212$)
Model statements for module: Decide 337
212$      BRANCH,      1:
                If,V Destino ( A PtAtual,1 + (A TmpSentido==0)) ==
0,2421$,Yes:
                Else,2422$,Yes
2421$      ASSIGN:      Nao esta ultrapassando neste patio via SB?.NumberOut
True=
                Nao esta ultrapassando neste patio via SB?.NumberOut
True + 1:NEXT(213$)

```



```

2422$      ASSIGN:      Nao esta ultrapassando neste patio via SB?.NumberOut
False=
                                Nao esta ultrapassando neste patio via SB?.NumberOut
False + 1:NEXT(38$)
      Model statements for module:  Assign 381
213$      ASSIGN:      V Ocpt ( A PtAtual , 1+ (A Sentido == 0))=0:
                                V PrioPatio ( A PtAtual , 1+ (A Sentido ==
0))=0:NEXT(38$)
      Model statements for module:  Assign 92
38$      ASSIGN:      IndLinha=1:
                                A Tmp=V Trecho ( A TmpTrecho , 4):
                                A TransitTime=A TransitTime * (1 * (A SouHelper==0) + E
Helpers(3,1) * (A SouHelper==1)):NEXT(192$)
      Model statements for module:  Assign 346
192$      ASSIGN:      Temp2=(4 * ( A TmpSentido == 0)) * (V Trecho(A
TmpTrecho,5) > 1):NEXT(340$)
      Model statements for module:  Assign 708
340$      ASSIGN:      V ocbl ( A TmpTrecho , 1 + Temp2)=1:NEXT(114$)
      Model statements for module:  Decide 126
114$      BRANCH,      1:
                                If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2423$,Yes:
                                Else,2424$,Yes
2423$      ASSIGN:      Ponto de checagem 6.NumberOut True=Ponto de checagem
6.NumberOut True + 1:NEXT(115$)
2424$      ASSIGN:      Ponto de checagem 6.NumberOut False=Ponto de checagem
6.NumberOut False + 1:NEXT(50$)
      Model statements for module:  Separate 15
115$      DUPLICATE,    100 - 50:
                                1,2427$,50:NEXT(2426$)
2426$      ASSIGN:      Separate 15.NumberOut Orig=Separate 15.NumberOut Orig +
1:NEXT(116$)
2427$      ASSIGN:      Separate 15.NumberOut Dup=Separate 15.NumberOut Dup +
1:NEXT(50$)
      Model statements for module:  Hold 23
116$      QUEUE,      Hold ENTRADA SBs.Queue:DETACH
      Model statements for module:  Separate 5
50$      DUPLICATE,    100 - 50:
                                1,2430$,50:NEXT(2429$)
2429$      ASSIGN:      Separate 5.NumberOut Orig=Separate 5.NumberOut Orig +
1:NEXT(98$)
2430$      ASSIGN:      Separate 5.NumberOut Dup=Separate 5.NumberOut Dup +
1:NEXT(39$)
      Model statements for module:  Decide 108
98$      BRANCH,      1:
                                If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2431$,Yes:
                                Else,2432$,Yes
2431$      ASSIGN:      Ponto de checagem 5.NumberOut True=Ponto de checagem
5.NumberOut True + 1:NEXT(238$)
2432$      ASSIGN:      Ponto de checagem 5.NumberOut False=Ponto de checagem
5.NumberOut False + 1:NEXT(51$)
      Model statements for module:  Separate 58
238$      DUPLICATE,    100 - 50:
                                1,2435$,50:NEXT(2434$)
2434$      ASSIGN:      Separate 58.NumberOut Orig=Separate 58.NumberOut Orig +
1:NEXT(96$)
2435$      ASSIGN:      Separate 58.NumberOut Dup=Separate 58.NumberOut Dup +
1:NEXT(51$)
      Model statements for module:  Hold 17
96$      QUEUE,      Hold SB ATUAL.Queue:DETACH
      Model statements for module:  Hold 11
51$      QUEUE,      E Filas Bloq( A TmpTrecho+( A TmpSentido == 0) ,
IndLinha + (4 * ( A TmpSentido == 0)) )
      WAIT:      A IDtrem * 100,1:NEXT(53$)

```

```

Model statements for module: Dispose 17
53$      ASSIGN:      Dispose 17.NumberOut=Dispose 17.NumberOut + 1
2436$    DISPOSE:      Yes
Model statements for module: Delay 9
39$      DELAY:      (A TransitTime)/ A Tmp,,Other:NEXT(41$)
Model statements for module: Decide 75
41$      BRANCH,      1:
                        If,IndLinha >= A Tmp,2437$,Yes:
                        Else,2438$,Yes
2437$    ASSIGN:      Chegou na ultima secao ?.NumberOut True=Chegou na ultima
secao ?.NumberOut True + 1:NEXT(42$)
2438$    ASSIGN:      Chegou na ultima secao ?.NumberOut False=Chegou na
ultima secao ?.NumberOut False + 1:NEXT(68$)
Model statements for module: Decide 76
42$      BRANCH,      1:
                        If,V Rot (A IndiRota + 1, A Rota) <> 0,2439$,Yes:
                        Else,2440$,Yes
2439$    ASSIGN:      O roteiro nao acabou aqui ?.NumberOut True=O roteiro nao
acabou aqui ?.NumberOut True + 1
                        :NEXT(236$)
2440$    ASSIGN:      O roteiro nao acabou aqui ?.NumberOut False=O roteiro
nao acabou aqui ?.NumberOut False + 1
                        :NEXT(43$)
Model statements for module: Decide 363
236$    BRANCH,      1:
                        If,chechar >= 0 .AND. chechar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2441$,Yes:
                        Else,2442$,Yes
2441$    ASSIGN:      Ponto de checagem 23.NumberOut True=Ponto de checagem
23.NumberOut True + 1:NEXT(237$)
2442$    ASSIGN:      Ponto de checagem 23.NumberOut False=Ponto de checagem
23.NumberOut False + 1:NEXT(40$)
Model statements for module: Separate 57
237$    DUPLICATE,    100 - 50:
                        1,2445$,50:NEXT(2444$)
2444$    ASSIGN:      Separate 57.NumberOut Orig=Separate 57.NumberOut Orig +
1:NEXT(234$)
2445$    ASSIGN:      Separate 57.NumberOut Dup=Separate 57.NumberOut Dup +
1:NEXT(40$)
Model statements for module: Hold 68
234$    QUEUE,        Hold 68.Queue:DETACH
Model statements for module: Assign 94
40$      ASSIGN:      A Patio=A PtAtual:
                        A OldSentido=A Sentido:
                        A Sentido=X Sentido(Indice2):
                        A Trecho=X Trecho(Indice2):
                        A PtAtual=X Patio(Indice2):NEXT(310$)
Model statements for module: Decide 581
310$    BRANCH,      1:
                        If,A Restricao2 == 1,2446$,Yes:
                        Else,2447$,Yes
2446$    ASSIGN:      Trem com restricao em SB ?.NumberOut True=Trem com
restricao em SB ?.NumberOut True + 1:NEXT(308$)
2447$    ASSIGN:      Trem com restricao em SB ?.NumberOut False=Trem com
restricao em SB ?.NumberOut False + 1
                        :NEXT(171$)
Model statements for module: Decide 580
308$    BRANCH,      1:
                        If,A PtAtual == A PtDestino,2448$,Yes:
                        Else,2449$,Yes
2448$    ASSIGN:      Patio adiante do restritivo destino final ?.NumberOut
True=
                        Patio adiante do restritivo destino final ?.NumberOut
True + 1:NEXT(309$)

```

```

2449$      ASSIGN:      Patio adiante do restritivo destino final ?.NumberOut
False=
                                Patio adiante do restritivo destino final ?.NumberOut
False + 1:NEXT(319$)
      Model statements for module:  Assign 689
309$      ASSIGN:      A Direto=1:NEXT(692$)
      Model statements for module:  Decide 226
692$      BRANCH,      1:
                                If,((E Verifica entroncamento .AND. V Ativa
Entroncamento) .OR. A Direto==1) .AND. A Restricao2==0,
                                2450$,Yes:
                                Else,2451$,Yes
2450$      ASSIGN:      Atingiu um entroncamento ou entrando direto?.NumberOut
True=
                                Atingiu um entroncamento ou entrando direto?.NumberOut
True + 1:NEXT(693$)
2451$      ASSIGN:      Atingiu um entroncamento ou entrando direto?.NumberOut
False=
                                Atingiu um entroncamento ou entrando direto?.NumberOut
False + 1:NEXT(697$)
      Model statements for module:  Decide 517
693$      BRANCH,      1:
                                If,A Direto==0,2452$,Yes:
                                Else,2453$,Yes
2452$      ASSIGN:      Nao entrando direto ?.NumberOut True=Nao entrando direto
?.NumberOut True + 1:NEXT(704$)
2453$      ASSIGN:      Nao entrando direto ?.NumberOut False=Nao entrando
direto ?.NumberOut False + 1:NEXT(694$)
      Model statements for module:  Assign 658
704$      ASSIGN:      Temp1=V ROT (Indice2 + A Indirota , A Rota):NEXT(711$)
      Model statements for module:  Assign 721
711$      ASSIGN:      A MarcaJump=TNOW:NEXT(708$)
      Model statements for module:  Decide 601
708$      BRANCH,      1:
                                If,V DROT (Indice2 + A Indirota, A Rota)==1,2454$,Yes:
                                Else,2455$,Yes
2454$      ASSIGN:      Identifica sentido a ser checado.NumberOut
True=Identifica sentido a ser checado.NumberOut True + 1
                                :NEXT(702$)
2455$      ASSIGN:      Identifica sentido a ser checado.NumberOut False=
Identifica sentido a ser checado.NumberOut False +
1:NEXT(703$)
      Model statements for module:  Decide 563
702$      BRANCH,      1:
                                If,V Node ( V Trecho (Temp1 ,1) , 1) == 1.
AND.0,2456$,Yes:
                                Else,2457$,Yes
2456$      ASSIGN:      Caminho crescente do entroncamento esta ocupado
SB?.NumberOut True=
                                Caminho crescente do entroncamento esta ocupado
SB?.NumberOut True + 1:NEXT(709$)
2457$      ASSIGN:      Caminho crescente do entroncamento esta ocupado
SB?.NumberOut False=
                                Caminho crescente do entroncamento esta ocupado
SB?.NumberOut False + 1:NEXT(700$)
      Model statements for module:  Decide 618
709$      BRANCH,      1:
                                If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2458$,Yes:
                                Else,2459$,Yes
2458$      ASSIGN:      Ponto de checagem 39.NumberOut True=Ponto de checagem
39.NumberOut True + 1:NEXT(712$)
2459$      ASSIGN:      Ponto de checagem 39.NumberOut False=Ponto de checagem
39.NumberOut False + 1:NEXT(712$)
      Model statements for module:  Decide 652

```

```

712$          BRANCH,          1:
                                If, (TNOW - A MarcaJump) > V Salto .AND. V ocpt ( A
PtAtual , 2 - (A Sentido == 0) ) == 1,2460$,Yes:
                                Else,2461$,Yes
2460$          ASSIGN:          Ficou travado3 ?.NumberOut True=Ficou travado3
?.NumberOut True + 1:NEXT(713$)
2461$          ASSIGN:          Ficou travado3 ?.NumberOut False=Ficou travado3
?.NumberOut False + 1:NEXT(705$)
          Model statements for module: Assign 720
713$          ASSIGN:          V OCBL(A TmpTrecho,1)=0:
                                V OCBL(A TmpTrecho,2)=0:
                                V OCBL(A TmpTrecho,3)=0:
                                V OCBL(A TmpTrecho,4)=0:
                                V Sentido(A TmpTrecho)=2:
                                A MarcaJump=TNOW+1000:
                                ContaPulos=ContaPulos+1:NEXT(714$)
          Model statements for module: ReadWrite 152
714$          WRITE,          Log de Erros,
                                "%1.8g. SALTO3 no trecho %1.8g do trem %1.8g. Tipo
%1.8g. Roteiro %1.8g. Com destino a %1.8g. Roteiro terminado em %1.8g sem encontrar
patio de destino. Trem abortado.\n":
                                TNOW,
                                A TmpTrecho,
                                A IDtrem,
                                A Tipo,
                                A Rota,
                                A PtDestino,
                                A PtAtual:NEXT(705$)
          Model statements for module: Delay 52
705$          DELAY:          unif(0.2,1),,Other:NEXT(702$)
          Model statements for module: Record 27
700$          COUNT:          Record 27,A IDTrem == Rastreo:NEXT(698$)
          Model statements for module: Assign 651
698$          ASSIGN:          V Node (V Trecho (Temp1,1) , 1)=1:
                                sh16=V Node (V Trecho (Temp1,1) , 1):NEXT(696$)
          Model statements for module: Assign 646
696$          ASSIGN:          A Node=V Trecho (Temp1,1):
                                A NodeS=1:NEXT(317$)
          Model statements for module: Decide 564
703$          BRANCH,          1:
                                If,V Node ( V Trecho (Temp1 ,1) , 2) == 1 .AND.
0,2462$,Yes:
                                Else,2463$,Yes
2462$          ASSIGN:          Caminho decrescente do entroncamento esta ocupado
SB?.NumberOut True=
                                Caminho decrescente do entroncamento esta ocupado
SB?.NumberOut True + 1:NEXT(710$)
2463$          ASSIGN:          Caminho decrescente do entroncamento esta ocupado
SB?.NumberOut False=
                                Caminho decrescente do entroncamento esta ocupado
SB?.NumberOut False + 1:NEXT(701$)
          Model statements for module: Decide 619
710$          BRANCH,          1:
                                If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreo.AND.Rastreo2<TNOW),2464$,Yes:
                                Else,2465$,Yes
2464$          ASSIGN:          Ponto de checagem 40.NumberOut True=Ponto de checagem
40.NumberOut True + 1:NEXT(715$)
2465$          ASSIGN:          Ponto de checagem 40.NumberOut False=Ponto de checagem
40.NumberOut False + 1:NEXT(715$)
          Model statements for module: Decide 653
715$          BRANCH,          1:
                                If, (TNOW - A MarcaJump) > V Salto .AND. V ocpt ( A
PtAtual , 2 - (A Sentido == 0) ) == 1,2466$,Yes:
                                Else,2467$,Yes

```

```

2466$      ASSIGN:      Ficou travado4?.NumberOut True=Ficou travado4?.NumberOut
True + 1:NEXT(716$)
2467$      ASSIGN:      Ficou travado4?.NumberOut False=Ficou
travado4?.NumberOut False + 1:NEXT(706$)
      Model statements for module: Assign 722
716$      ASSIGN:      V OCBL(A TmpTrecho,1)=0:
V OCBL(A TmpTrecho,2)=0:
V OCBL(A TmpTrecho,3)=0:
V OCBL(A TmpTrecho,4)=0:
V Sentido(A TmpTrecho)=2:
A MarcaJump=TNOW+1000:
ContaPulos=ContaPulos+1:NEXT(717$)
      Model statements for module: ReadWrite 153
717$      WRITE,      Log de Erros,
"%1.8g. SALTO3 no trecho %1.8g do trem %1.8g. Tipo
%1.8g. Roteiro %1.8g. Com destino a %1.8g. Roteiro terminado em %1.8g sem encontrar
patio de destino. Trem abortado.\n":
      TNOW,
      A TmpTrecho,
      A IDtrem,
      A Tipo,
      A Rota,
      A PtDestino,
      A PtAtual:NEXT(706$)
      Model statements for module: Delay 53
706$      DELAY:      unif(0.2,1),,Other:NEXT(703$)
      Model statements for module: Record 28
701$      COUNT:      Record 28,A IDTrem == Rastreio:NEXT(699$)
      Model statements for module: Assign 652
699$      ASSIGN:      V Node ( V Trecho (Temp1 ,1) , 2)=1:
sh17=V Node ( V Trecho (Temp1 ,1) , 2):NEXT(707$)
      Model statements for module: Assign 668
707$      ASSIGN:      A Node=V Trecho (Temp1 ,1):
A NodeS=2:NEXT(317$)
      Model statements for module: Assign 631
694$      ASSIGN:      Temp3=8300:
V Conta bypass=V Conta bypass + 1:NEXT(317$)
      Model statements for module: Decide 544
697$      BRANCH,      1:
If,E Verifica entroncamento,2468$,Yes:
Else,2469$,Yes
2468$      ASSIGN:      Patio adiante esta em entroncamento ?.NumberOut True=
Patio adiante esta em entroncamento ?.NumberOut True +
1:NEXT(694$)
2469$      ASSIGN:      Patio adiante esta em entroncamento ?.NumberOut False=
Patio adiante esta em entroncamento ?.NumberOut False +
1:NEXT(695$)
      Model statements for module: Assign 632
695$      ASSIGN:      V ocpt ( A PtAtual , 1 + (A Sentido == 0) )=1:
V PrioPatio ( A PtAtual , 1 + (A Sentido == 0) )=A
Prioridade:
Temp3=0:NEXT(317$)
      Model statements for module: Decide 584
317$      BRANCH,      1:
If,A Restricao2 == 1 .AND. E Verifica
entroncamento==0,2470$,Yes:
Else,2471$,Yes
2470$      ASSIGN:      Trem saindo com restricao em SB ?.NumberOut True=
Trem saindo com restricao em SB ?.NumberOut True +
1:NEXT(318$)
2471$      ASSIGN:      Trem saindo com restricao em SB ?.NumberOut False=
Trem saindo com restricao em SB ?.NumberOut False +
1:NEXT(343$)
      Model statements for module: Assign 693

```

```

318$      ASSIGN:          V Reserva (A TmpTrecho, 1+(A TmpSentido==0) )=V Reserva
(A TmpTrecho, 1+(A TmpSentido==0) ) - 1
          :NEXT(343$)
      Model statements for module: Decide 609
343$      BRANCH,          1:
          If, checar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2472$,Yes:
          Else,2473$,Yes
2472$      ASSIGN:          Ponto de checagem 33.NumberOut True=Ponto de checagem
33.NumberOut True + 1:NEXT(344$)
2473$      ASSIGN:          Ponto de checagem 33.NumberOut False=Ponto de checagem
33.NumberOut False + 1:NEXT(70$)
      Model statements for module: Separate 75
344$      DUPLICATE,       100 - 50:
          1,2476$,50:NEXT(2475$)
2475$      ASSIGN:          Separate 75.NumberOut Orig=Separate 75.NumberOut Orig +
1:NEXT(341$)
2476$      ASSIGN:          Separate 75.NumberOut Dup=Separate 75.NumberOut Dup +
1:NEXT(70$)
      Model statements for module: Hold 88
341$      QUEUE,           Hold 88.Queue:DETACH
      Model statements for module: Signal 10
70$      SIGNAL:          A IDtrem * 100,1:NEXT(48$)
      Model statements for module: Assign 99
48$      ASSIGN:          V ocbl ( A TmpTrecho , IndLinha + Temp2 )=0:
          A PassouSB=1:
          A Direto=0:NEXT(174$)
      Model statements for module: Decide 264
174$      BRANCH,          1:
          If,V Destino ( A PtAtual,1 + (A Sentido==0)) ==
0,2477$,Yes:
          Else,2478$,Yes
2477$      ASSIGN:          Nao vai fazer ultrapassagem ?.NumberOut True=Nao vai
fazer ultrapassagem ?.NumberOut True + 1
          :NEXT(47$)
2478$      ASSIGN:          Nao vai fazer ultrapassagem ?.NumberOut False=Nao vai
fazer ultrapassagem ?.NumberOut False + 1
          :NEXT(322$)
      Model statements for module: Route 5
47$      ROUTE:           0.000000000000000,ESTA ( A PtAtual,1 + ( A Sentido == 0)
)
      Model statements for module: Signal 30
322$      SIGNAL:          sorteio,1:NEXT(170$)
      Model statements for module: Assign 338
170$      ASSIGN:          Sorteio=Unif(21000,25000,4):
          A Ultrapassa=1:NEXT(169$)
      Model statements for module: Separate 31
169$      DUPLICATE,       100 - 50:
          1,2481$,50:NEXT(2480$)
2480$      ASSIGN:          Separate 31.NumberOut Orig=Separate 31.NumberOut Orig +
1:NEXT(167$)
2481$      ASSIGN:          Separate 31.NumberOut Dup=Separate 31.NumberOut Dup +
1:NEXT(47$)
      Model statements for module: Assign 337
167$      ASSIGN:          Picture=Picture.ultrapassa:
          A PtAtual=A Patio:
          A Sentido=A Sentido == 0:NEXT(168$)
      Model statements for module: Route 22
168$      ROUTE:           0.000000000000000,Station Anima Evento
      Model statements for module: Decide 585
319$      BRANCH,          1:
          If,Indice > Indice2,2482$,Yes:
          Else,2483$,Yes
2482$      ASSIGN:          Trem ainda nao chegou no patio sem restricao.NumberOut
True=

```

```

Trem ainda nao chegou no patio sem restricao.NumberOut
True + 1:NEXT(355$)
2483$      ASSIGN:      Trem ainda nao chegou no patio sem restricao.NumberOut
False=
Trem ainda nao chegou no patio sem restricao.NumberOut
False + 1:NEXT(171$)
  Model statements for module:  Assign 718
355$      ASSIGN:      A MarcaJump=TNOW:NEXT(311$)
  Model statements for module:  Decide 582
311$      BRANCH,      1:
                                If,V ocbl ( X Trecho(Indice2+1) , (1+Temp2)) ==
0,2484$,Yes:
                                Else,2485$,Yes
2484$      ASSIGN:      SB seguinte ao proximo patio ficou livre ?.NumberOut
True=
                                SB seguinte ao proximo patio ficou livre ?.NumberOut
True + 1:NEXT(317$)
2485$      ASSIGN:      SB seguinte ao proximo patio ficou livre ?.NumberOut
False=
                                SB seguinte ao proximo patio ficou livre ?.NumberOut
False + 1:NEXT(347$)
  Model statements for module:  Decide 616
347$      BRANCH,      1:
                                If, checar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2486$,Yes:
                                Else,2487$,Yes
2486$      ASSIGN:      Ponto de checagem 38.NumberOut True=Ponto de checagem
38.NumberOut True + 1:NEXT(764$)
2487$      ASSIGN:      Ponto de checagem 38.NumberOut False=Ponto de checagem
38.NumberOut False + 1:NEXT(764$)
  Model statements for module:  Decide 651
764$      BRANCH,      1:
                                If,(TNOW - A MarcaJump) > V Salto .AND. V ocpt ( A
PtAtual , 2 - (A Sentido == 0) ) == 1,2488$,Yes:
                                Else,2489$,Yes
2488$      ASSIGN:      Ficou travado2 ?.NumberOut True=Ficou travado2
?.NumberOut True + 1:NEXT(765$)
2489$      ASSIGN:      Ficou travado2 ?.NumberOut False=Ficou travado2
?.NumberOut False + 1:NEXT(312$)
  Model statements for module:  Assign 719
765$      ASSIGN:      V OCBL(A TmpTrecho,1)=0:
                                V OCBL(A TmpTrecho,2)=0:
                                V OCBL(A TmpTrecho,3)=0:
                                V OCBL(A TmpTrecho,4)=0:
                                V Sentido(A TmpTrecho)=2:
                                A MarcaJump=TNOW+1000:
                                ContaPulos=ContaPulos+1:NEXT(766$)
  Model statements for module:  ReadWrite 151
766$      WRITE,      Log de Erros,
                                "%1.8g. SALTO2 no trecho %1.8g do trem %1.8g. Tipo
%1.8g. Roteiro %1.8g. Com destino a %1.8g. Roteiro terminado em %1.8g sem encontrar
patio de destino. Trem abortado.\n":
                                TNOW,
                                A TmpTrecho,
                                A IDtrem,
                                A Tipo,
                                A Rota,
                                A PtDestino,
                                A PtAtual:NEXT(312$)
  Model statements for module:  Delay 55
312$      DELAY:      UNIF(0.5,2.7),,Other:NEXT(311$)
  Model statements for module:  Decide 262
171$      BRANCH,      1:
                                If,V Destino ( A PtAtual,1 + (A Sentido==0)) ==
0,2490$,Yes:

```

```

Else,2491$,Yes
2490$      ASSIGN:      Patio seguinte nao esta ocupado por trem em fila
?.NumberOut True=
                Patio seguinte nao esta ocupado por trem em fila
?.NumberOut True + 1:NEXT(301$)
2491$      ASSIGN:      Patio seguinte nao esta ocupado por trem em fila
?.NumberOut False=
                Patio seguinte nao esta ocupado por trem em fila
?.NumberOut False + 1:NEXT(172$)
    Model statements for module:  Assign 686
301$      ASSIGN:      A MarcaJump=TNOW:NEXT(260$)
    Model statements for module:  Decide 427
260$      BRANCH,      1:
                If,A PtAtual == A PtDestino,2492$,Yes:
                Else,2493$,Yes
2492$      ASSIGN:      Patio adiante destino final ?.NumberOut True=Patio
adiante destino final ?.NumberOut True + 1
                :NEXT(346$)
2493$      ASSIGN:      Patio adiante destino final ?.NumberOut False=Patio
adiante destino final ?.NumberOut False + 1
                :NEXT(71$)
    Model statements for module:  Decide 612
346$      BRANCH,      1:
                If,A Tipo <> 2 .OR. (A Patio <> 151 .AND. A Patio <> 7
.AND. A Patio <> 148),2494$,Yes:
                Else,2495$,Yes
2494$      ASSIGN:      Trem nao esta destinado a ponto zero ?.NumberOut True=
Trem nao esta destinado a ponto zero ?.NumberOut True +
1:NEXT(261$)
2495$      ASSIGN:      Trem nao esta destinado a ponto zero ?.NumberOut False=
Trem nao esta destinado a ponto zero ?.NumberOut False +
1:NEXT(71$)
    Model statements for module:  Assign 481
261$      ASSIGN:      A Direto=1:NEXT(692$)
    Model statements for module:  Decide 88
71$      BRANCH,      1:
                If,V ocpt ( A PtAtual , 1 + (A Sentido == 0) ) ==
0,2496$,Yes:
                Else,2497$,Yes
2496$      ASSIGN:      Patio adiante ficou livre ?.NumberOut True=Patio adiante
ficou livre ?.NumberOut True + 1
                :NEXT(692$)
2497$      ASSIGN:      Patio adiante ficou livre ?.NumberOut False=Patio
adiante ficou livre ?.NumberOut False + 1
                :NEXT(243$)
    Model statements for module:  Decide 376
243$      BRANCH,      1:
                If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2498$,Yes:
                Else,2499$,Yes
2498$      ASSIGN:      Ponto de checagem 27.NumberOut True=Ponto de checagem
27.NumberOut True + 1:NEXT(331$)
2499$      ASSIGN:      Ponto de checagem 27.NumberOut False=Ponto de checagem
27.NumberOut False + 1:NEXT(750$)
    Model statements for module:  Assign 703
331$      ASSIGN:      A Patio=A Patio:NEXT(750$)
    Model statements for module:  Decide 576
750$      BRANCH,      1:
                If,(TNOW - A MarcaJump) > V Salto .AND. V ocpt ( A
PtAtual , 2 - (A Sentido == 0) ) == 1,2500$,Yes:
                Else,2501$,Yes
2500$      ASSIGN:      Ficou travado ?.NumberOut True=Ficou travado ?.NumberOut
True + 1:NEXT(751$)
2501$      ASSIGN:      Ficou travado ?.NumberOut False=Ficou travado
?.NumberOut False + 1:NEXT(72$)

```



```

Model statements for module: Assign 685
751$      ASSIGN:      V OCBL(A TmpTrecho,1)=0:
                          V OCBL(A TmpTrecho,2)=0:
                          V OCBL(A TmpTrecho,3)=0:
                          V OCBL(A TmpTrecho,4)=0:
                          V Sentido(A TmpTrecho)=2:
                          A MarcaJump=TNOW+1000:
                          ContaPulos=ContaPulos+1:NEXT(752$)
Model statements for module: ReadWrite 110
752$      WRITE,      Log de Erros,
                          "%1.8g. SALTO no trecho %1.8g do trem %1.8g. Tipo %1.8g.
Roteiro %1.8g. Com destino a %1.8g. \n":
                          TNOW,
                          A TmpTrecho,
                          A IDtrem,
                          A Tipo,
                          A Rota,
                          A PtDestino,
                          A PtAtual:NEXT(72$)
Model statements for module: Delay 14
72$      DELAY:      UNIF(0.2,3),,Other:NEXT(71$)
Model statements for module: Decide 263
172$      BRANCH,      1:
                          If,V ocpt ( A PtAtual , 2 - (A Sentido == 0) ) ==
0,2502$,Yes:
                          Else,2503$,Yes
2502$      ASSIGN:      Patio adiante no sentido inverso ficou livre ?.NumberOut
True=
                          Patio adiante no sentido inverso ficou livre ?.NumberOut
True + 1:NEXT(261$)
2503$      ASSIGN:      Patio adiante no sentido inverso ficou livre ?.NumberOut
False=
                          Patio adiante no sentido inverso ficou livre ?.NumberOut
False + 1:NEXT(173$)
Model statements for module: Delay 39
173$      DELAY:      UNIF(0.2,3),,Other:NEXT(242$)
Model statements for module: Decide 374
242$      BRANCH,      1:
                          If,chechar >= 0 .AND. chechar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2504$,Yes:
                          Else,2505$,Yes
2504$      ASSIGN:      Ponto de checagem 26.NumberOut True=Ponto de checagem
26.NumberOut True + 1:NEXT(171$)
2505$      ASSIGN:      Ponto de checagem 26.NumberOut False=Ponto de checagem
26.NumberOut False + 1:NEXT(171$)
Model statements for module: ReadWrite 11
43$      WRITE,      Log de Erros,
                          "%1.8g. ERR5 Erro no trem %1.8g. Roteiro %1.8g
finalizado sem encontrar a estacao de destino %1.8g. Trem abortado no patio
%1.8g.\n":
                          TNOW,
                          A IDtrem,
                          A Rota,
                          A PtDestino,
                          A PtAtual:NEXT(44$)
Model statements for module: Release 7
44$      RELEASE:      ESTA_R ( A PtAtual,1 + (A Sentido==0)),1:NEXT(45$)
Model statements for module: Assign 95
45$      ASSIGN:      A Trecho=V ROT(A IndiRota - (A Sentido == 0), A Rota):
                          V Ocpt ( A Trecho , 1+ (A Sentido==0) )=0:
                          V ocbl ( A TmpTrecho , IndLinha - 1 + temp2)=0:NEXT(73$)
Model statements for module: Signal 11
73$      SIGNAL:      A IDtrem * 100,1:NEXT(180$)
Model statements for module: Route 27
180$      ROUTE:      0.000000000000000,Erro na movimentacao

```

```

Model statements for module: Decide 87
68$      BRANCH,      1:
          If,V ocbl ( A TmpTrecho , (IndLinha + 1 + Temp2)) ==
0,2506$,Yes:
          Else,2507$,Yes
2506$    ASSIGN:      SB seguinte ficou livre ?.NumberOut True=SB seguinte
ficou livre ?.NumberOut True + 1:NEXT(54$)
2507$    ASSIGN:      SB seguinte ficou livre ?.NumberOut False=SB seguinte
ficou livre ?.NumberOut False + 1:NEXT(345$)
Model statements for module: Signal 8
54$      SIGNAL:      A IDtrem * 100,1:NEXT(49$)
Model statements for module: Assign 100
49$      ASSIGN:      V ocbl ( A TmpTrecho , IndLinha + 1+ Temp2)=1:
          V ocbl ( A TmpTrecho , IndLinha + Temp2)=0:
          IndLinha=IndLinha + 1:NEXT(50$)
Model statements for module: Decide 610
345$     BRANCH,      1:
          If,chechar >= 0 .AND. chechar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2508$,Yes:
          Else,2509$,Yes
2508$    ASSIGN:      Ponto de checagem 37.NumberOut True=Ponto de checagem
37.NumberOut True + 1:NEXT(69$)
2509$    ASSIGN:      Ponto de checagem 37.NumberOut False=Ponto de checagem
37.NumberOut False + 1:NEXT(69$)
Model statements for module: Delay 13
69$      DELAY:      UNIF(0.5,2.7),,Other:NEXT(68$)
Model statements for module: Assign 135
102$     ASSIGN:      A Tmp=(1 + (A TmpSentido==0)) * Temp1:
          A TransitTime=TT Trechos ( A TmpTrecho , A
Tmp):NEXT(582$)
Model statements for module: Assign 705
332$     ASSIGN:      IndLinha=1:
          Indice2=1:NEXT(333$)
Model statements for module: Decide 634
353$     BRANCH,      1:
          If,V Gravar em texto == 1,2510$,Yes:
          Else,2511$,Yes
2510$    ASSIGN:      Decide 634.NumberOut True=Decide 634.NumberOut True +
1:NEXT(358$)
2511$    ASSIGN:      Decide 634.NumberOut False=Decide 634.NumberOut False +
1:NEXT(220$)
Especificação do arquivo de LOG:
1 - Identificador único do trem (A IDTrem)
2 - Pátio (A PtAtual)
3 - Momento da entrada no pátio (A Tmp)
4 - Momento da saída do trem (TNOW)
5 - Número do trem (A NTrem)
6 - Horário do trem (A Horario)
7 - Tipo do trem (A Tipo)
8 - Gráfico onde entra esta informação (E Patios (A PtAtual,9))
Model statements for module: ReadWrite 159
358$     WRITE,      Log de Movimentacao:
          Dia,
          A IDtrem,
          N Trem,
          A Horario,
          A PtAtual,
          A Tmp,
          TNOW,
          A Tipo,
          A Trecho,
          A Sentido,
          E Patios (A PtAtual,9),
          E Patios (A PtAtual,10),
          A PtDestino,

```

```

A Produto,
A Cliente,
tnow - marca:NEXT(334$)
Especificação do arquivo de LOG:
1 - Identificador único do trem (A IDTrem)
2 - Pátio (A PtAtual)
3 - Momento da entrada no pátio (A Tmp)
4 - Momento da saída do trem (TNOW)
5 - Número do trem (A NTrem)
6 - Horário do trem (A Horario)
7 - Tipo do trem (A Tipo)
8 - Gráfico onde entra esta informação (E Patios (A PtAtual,9))
Model statements for module: ReadWrite 67
220$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeMovimentacao):
          Dia,
          A IDtrem,
          N Trem,
          A Horario,
          A PtAtual,
          A Tmp,
          TNOW,
          A Tipo,
          A Trecho,
          A Sentido,
          E Patios (A PtAtual,9),
          E Patios (A PtAtual,10),
          A PtDestino,
          A Produto,
          A Cliente,
          tnow - marca:NEXT(334$)
Model statements for module: Assign 687
303$      ASSIGN:      A Restricao2=1:NEXT(89$)
Model statements for module: Decide 105
89$      BRANCH,      1:
          If,E Patios (A Patio , 6 ) == 1,2512$,Yes:
          Else,2513$,Yes
2512$      ASSIGN:      Patio esta ATIVO ?.NumberOut True=Patio esta ATIVO
?.NumberOut True + 1:NEXT(493$)
2513$      ASSIGN:      Patio esta ATIVO ?.NumberOut False=Patio esta ATIVO
?.NumberOut False + 1:NEXT(30$)
Model statements for module: Decide 514
493$      BRANCH,      1:
          If,A Patio == A PtDestino,2514$,Yes:
          Else,2515$,Yes
2514$      ASSIGN:      Entrando direto via normal?.NumberOut True=Entrando
direto via normal?.NumberOut True + 1
: NEXT(514$)
2515$      ASSIGN:      Entrando direto via normal?.NumberOut False=Entrando
direto via normal?.NumberOut False + 1
: NEXT(499$)
Model statements for module: Decide 615
514$      BRANCH,      1:
          If,A Tipo <> 2 .OR. (A Patio <> 151 .AND. A Patio <> 7
.AND. A Patio <> 148),2516$,Yes:
          Else,2517$,Yes
2516$      ASSIGN:      Trem em reserva nao esta destinado a ponto zero
?.NumberOut True=
          Trem em reserva nao esta destinado a ponto zero
?.NumberOut True + 1:NEXT(491$)
2517$      ASSIGN:      Trem em reserva nao esta destinado a ponto zero
?.NumberOut False=
          Trem em reserva nao esta destinado a ponto zero
?.NumberOut False + 1:NEXT(499$)
Model statements for module: Assign 307
491$      ASSIGN:      V Conta bypass=V Conta bypass + 1:

```

```

Temp3=8300:NEXT(30$)
Model statements for module: Decide 543
499$      BRANCH,      1:
If,E Verifica entroncamento A Patio .AND. V Ativa
Entroncamento,2518$,Yes:
Else,2519$,Yes
2518$      ASSIGN:      Esta em patio de entroncamento ?.NumberOut True=Esta em
patio de entroncamento ?.NumberOut True + 1
:NEXT(506$)
2519$      ASSIGN:      Esta em patio de entroncamento ?.NumberOut False=
Esta em patio de entroncamento ?.NumberOut False +
1:NEXT(495$)
Model statements for module: Decide 574
506$      BRANCH,      1:
If,A Restricao == 1,2520$,Yes:
Else,2521$,Yes
2520$      ASSIGN:      Trem com restricao ?.NumberOut True=Trem com restricao
?.NumberOut True + 1:NEXT(500$)
2521$      ASSIGN:      Trem com restricao ?.NumberOut False=Trem com restricao
?.NumberOut False + 1:NEXT(513$)
Model statements for module: Decide 554
500$      BRANCH,      1:
If,V DROT ((A Indirota + Indice2) , A Rota) ==
1,2522$,Yes:
Else,2523$,Yes
2522$      ASSIGN:      Descubre sentido para bloquear trecho.NumberOut True=
Descubre sentido para bloquear trecho.NumberOut True +
1:NEXT(497$)
2523$      ASSIGN:      Descubre sentido para bloquear trecho.NumberOut False=
Descubre sentido para bloquear trecho.NumberOut False +
1:NEXT(498$)
Model statements for module: Record 1
497$      COUNT:      Record 1,A IDTrem == Rastreio:NEXT(494$)
Model statements for module: Assign 608
494$      ASSIGN:      V Node ( V Trecho (V ROT ((A Indirota + Indice2) , A
Rota) ,1) , 1)=1:NEXT(504$)
Model statements for module: Decide 573
504$      BRANCH,      1:
If,chechar >= 0 .AND. chechar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW) ,2524$,Yes:
Else,2525$,Yes
2524$      ASSIGN:      Ponto de checagem 34.NumberOut True=Ponto de checagem
34.NumberOut True + 1:NEXT(505$)
2525$      ASSIGN:      Ponto de checagem 34.NumberOut False=Ponto de checagem
34.NumberOut False + 1:NEXT(491$)
Model statements for module: Separate 72
505$      DUPLICATE,    100 - 50:
1,2528$,50:NEXT(2527$)
2527$      ASSIGN:      Separate 72.NumberOut Orig=Separate 72.NumberOut Orig +
1:NEXT(502$)
2528$      ASSIGN:      Separate 72.NumberOut Dup=Separate 72.NumberOut Dup +
1:NEXT(491$)
Model statements for module: Hold 85
502$      QUEUE,      Hold 85.Queue:DETACH
Model statements for module: Record 2
498$      COUNT:      Record 2,A IDTrem == Rastreio:NEXT(496$)
Model statements for module: Assign 612
496$      ASSIGN:      V Node ( V Trecho (V ROT ((A Indirota + Indice2) , A
Rota) ,1) , 2)=1:NEXT(504$)
Model statements for module: Decide 597
513$      BRANCH,      1:
If,V DROT ((A Indirota + Indice2) , A Rota) ==
1,2529$,Yes:
Else,2530$,Yes

```

```

2529$      ASSIGN:      Descobre sentido para bloquear trecho sem
restricao.NumberOut True=
                                Descobre sentido para bloquear trecho sem
restricao.NumberOut True + 1:NEXT(510$)
2530$      ASSIGN:      Descobre sentido para bloquear trecho sem
restricao.NumberOut False=
                                Descobre sentido para bloquear trecho sem
restricao.NumberOut False + 1:NEXT(511$)
    Model statements for module: Record 33
510$      COUNT:      Record 33,A IDTrem == Rastreio:NEXT(508$)
    Model statements for module: Assign 679
508$      ASSIGN:      V Node ( V Trecho (V ROT ((A Indirota + Indice2) , A
Rota) ,1) , 1)=1:NEXT(507$)
    Model statements for module: Assign 678
507$      ASSIGN:      A Node=V Trecho (V ROT ((A Indirota + Indice2) , A Rota)
,1):
                                A NodeS=1:NEXT(504$)
    Model statements for module: Record 34
511$      COUNT:      Record 34,A IDTrem == Rastreio:NEXT(509$)
    Model statements for module: Assign 680
509$      ASSIGN:      V Node ( V Trecho (V ROT ((A Indirota + Indice2) , A
Rota) ,1) , 2)=1:NEXT(512$)
    Model statements for module: Assign 681
512$      ASSIGN:      A Node=V Trecho (V ROT ((A Indirota + Indice2) , A
Rota),1):
                                A NodeS=2:NEXT(504$)
    Model statements for module: Decide 515
495$      BRANCH,      1:
                                If,V Destino ( A Patio,1 + (A Sentido==0)) == 0 .AND. A
Restricao2<>1,2531$,Yes:
                                Else,2532$,Yes
2531$      ASSIGN:      Patio nao esta ocupado por trem em fila ?.NumberOut
True=
                                Patio nao esta ocupado por trem em fila ?.NumberOut True
+ 1:NEXT(492$)
2532$      ASSIGN:      Patio nao esta ocupado por trem em fila ?.NumberOut
False=
                                Patio nao esta ocupado por trem em fila ?.NumberOut
False + 1:NEXT(491$)
    Model statements for module: Assign 308
492$      ASSIGN:      V ocpt ( A Patio , 1+ ( A TmpSentido == 0) )=1:
                                V PrioPatio ( A Patio , 1+ ( A TmpSentido == 0) )=A
Prioridade:
                                Temp2=0:NEXT(30$)
    Model statements for module: ReadWrite 106
501$      WRITE,      Log de Erros,
                                "%1.8g. XXX Trem %1.8g Rot %1.8g. IndRt %1.8g PtAtual
%1.8g. APatio %1.8g Restr %1.8g. SB %1.8g. A Trecho %1.8g Trecho Indrt %1.8g Trecho
Indrt m1 %1.8g. Indice2 %1.8g\n":
                                TNOW,
                                A IDtrem,
                                A Rota,
                                A IndiRota,
                                A PtAtual,
                                A Patio,
                                A Restricao,
                                A PassouSB,
                                A Trecho,
                                V ROT (A Indirota, A Rota),
                                V ROT (A Indirota+1, A Rota),
                                Indice2:NEXT(491$)
    Model statements for module: Decide 70
30$      BRANCH,      1:
                                If,Indice2 >= Indice,2533$,Yes:
                                Else,2534$,Yes

```

```

2533$      ASSIGN:      Bloqueou todo o percurso ?.NumberOut True=Bloqueou todo
o percurso ?.NumberOut True + 1:NEXT(55$)
2534$      ASSIGN:      Bloqueou todo o percurso ?.NumberOut False=Bloqueou todo
o percurso ?.NumberOut False + 1
:      :NEXT(568$)
      Model statements for module:  Signal 9
55$        SIGNAL:      A IDtrem,1:NEXT(226$)
      Model statements for module:  Decide 359
226$        BRANCH,      1:
:      :If,A LinhaOcupada == A PtAtual,2535$,Yes:
:      :Else,2536$,Yes
2535$      ASSIGN:      Trem esta ocupando linha de estacionamento ?.NumberOut
True=
:      :Trem esta ocupando linha de estacionamento ?.NumberOut
True + 1:NEXT(227$)
2536$      ASSIGN:      Trem esta ocupando linha de estacionamento ?.NumberOut
False=
:      :Trem esta ocupando linha de estacionamento ?.NumberOut
False + 1:NEXT(214$)
      Model statements for module:  Assign 429
227$        ASSIGN:      V StatusPt (A PtAtual, 1)=V StatusPt (A PtAtual, 1) + 1:
A LinhaOcupada=0:NEXT(214$)
      Model statements for module:  Decide 338
214$        BRANCH,      1:
:      :If,V Destino ( A PtAtual,1 + (A Sentido==0)) == 0 .OR. A
ParEsp==A PtAtual,2537$,Yes:
:      :Else,2538$,Yes
2537$      ASSIGN:      Nao esta ultrapassando neste patio com
restricao?.NumberOut True=
:      :Nao esta ultrapassando neste patio com
restricao?.NumberOut True + 1:NEXT(57$)
2538$      ASSIGN:      Nao esta ultrapassando neste patio com
restricao?.NumberOut False=
:      :Nao esta ultrapassando neste patio com
restricao?.NumberOut False + 1:NEXT(175$)
      Model statements for module:  Assign 102
57$        ASSIGN:      V Ocpt ( A PtAtual , 1+ (A Sentido == 0))=0:
V PrioPatio ( A PtAtual , 1+ (A Sentido ==
0))=0:NEXT(175$)
      Model statements for module:  Assign 339
175$        ASSIGN:      IndLinha=1:
A Trecho=X Trecho(1):
A Sentido=X Sentido(1):
Indice2=1:
A PtAtual=X Patio(1):
Temp1=1*(A Tipo<4) + 2*(A Tipo==4) + 3*(A Tipo==5):
A TmpTrecho=A Trecho:
A TmpSentido=A Sentido:NEXT(109$)
      Model statements for module:  Release 10
109$        RELEASE:      Passo3,1:NEXT(266$)
      Model statements for module:  Decide 433
266$        BRANCH,      1:
:      :If,TNOW > (V Warmup * 1440),2539$,Yes:
:      :Else,2540$,Yes
2539$      ASSIGN:      Warmup do log de movimentacao 3.NumberOut True=Warmup do
log de movimentacao 3.NumberOut True + 1
:      :NEXT(219$)
2540$      ASSIGN:      Warmup do log de movimentacao 3.NumberOut False=Warmup
do log de movimentacao 3.NumberOut False + 1
:      :NEXT(285$)
      Model statements for module:  Decide 347
219$        BRANCH,      1:
:      :If,(tnow - marca) < 0.001,2541$,Yes:
:      :Else,2542$,Yes

```

```

2541$      ASSIGN:      Nao houve espera por problemas de transito ?.NumberOut
True=
                                     Nao houve espera por problemas de transito ?.NumberOut
True + 1:NEXT(285$)
2542$      ASSIGN:      Nao houve espera por problemas de transito ?.NumberOut
False=
                                     Nao houve espera por problemas de transito ?.NumberOut
False + 1:NEXT(352$)
      Model statements for module:  Assign 543
285$      ASSIGN:      Temp2=A Produto * (A Produto <> 2) + 1 * (A Produto ==
2):NEXT(669$)
      Model statements for module:  Decide 237
669$      BRANCH,      1:
                                     If,A ParEsp == 0,2543$,Yes:
                                     Else,2544$,Yes
2543$      ASSIGN:      V Destino nao usada por este trem ?.NumberOut True=
                                     V Destino nao usada por este trem ?.NumberOut True +
1:NEXT(742$)
2544$      ASSIGN:      V Destino nao usada por este trem ?.NumberOut False=
                                     V Destino nao usada por este trem ?.NumberOut False +
1:NEXT(672$)
      Model statements for module:  Decide 478
672$      BRANCH,      1:
                                     If,A CabFila == 1,2545$,Yes:
                                     Else,2546$,Yes
2545$      ASSIGN:      Trem encabeca esta fila ?.NumberOut True=Trem encabeca
esta fila ?.NumberOut True + 1:NEXT(670$)
2546$      ASSIGN:      Trem encabeca esta fila ?.NumberOut False=Trem encabeca
esta fila ?.NumberOut False + 1:NEXT(671$)
      Model statements for module:  Assign 327
670$      ASSIGN:      V Destino ( A ParEsp,1 + (A Sentido==0))=0:
                                     V DestinoP( A ParEsp,1 + (A Sentido==0))=0:
                                     A ParEsp=0:
                                     A CabFila=0:NEXT(674$)
      Model statements for module:  Decide 519
674$      BRANCH,      1:
                                     If,E Verifica entroncamento,2547$,Yes:
                                     Else,2548$,Yes
2547$      ASSIGN:      Saindo de fila de entroncamento para via em
restricao?.NumberOut True=
                                     Saindo de fila de entroncamento para via em
restricao?.NumberOut True + 1:NEXT(673$)
2548$      ASSIGN:      Saindo de fila de entroncamento para via em
restricao?.NumberOut False=
                                     Saindo de fila de entroncamento para via em
restricao?.NumberOut False + 1:NEXT(742$)
      Model statements for module:  Assign 611
673$      ASSIGN:      V NodeD ( V Trecho (V ROT(A Indirota + (A Sentido ==0),
A Rota),1) , 1 + (A Sentido==0))=0:
                                     V NodeDP ( V Trecho (V ROT(A Indirota + (A Sentido ==0),
A Rota),1) , 1 + (A Sentido==0))=0
                                     :NEXT(742$)
      Model statements for module:  Assign 541
671$      ASSIGN:      V Destino ( A PtAtual,1 + (A Sentido==0))=A PtDestino:
                                     V DestinoP ( A PtAtual,1 + (A
Sentido==0))=Temp2:NEXT(670$)
      Model statements for module:  Decide 568
742$      BRANCH,      1:
                                     If,E Verifica entroncamento .AND. V Ativa Entroncamento
.AND. A Restricao == 1,2549$,Yes:
                                     Else,2550$,Yes
2549$      ASSIGN:      Esta em patio de entroncamento RESTR?.NumberOut True=
                                     Esta em patio de entroncamento RESTR?.NumberOut True +
1:NEXT(749$)
2550$      ASSIGN:      Esta em patio de entroncamento RESTR?.NumberOut False=

```

```

Esta em patio de entroncamento RESTR?.NumberOut False +
1:NEXT(106$)
  Model statements for module: Decide 598
749$      BRANCH,      1:
          If,V DROT ((A Indirota + Indice2) , A Rota) ==
1,2551$,Yes:
          Else,2552$,Yes
2551$      ASSIGN:      Descubre o sentido.NumberOut True=Descubre o
sentido.NumberOut True + 1:NEXT(740$)
2552$      ASSIGN:      Descubre o sentido.NumberOut False=Descubre o
sentido.NumberOut False + 1:NEXT(741$)
  Model statements for module: Record 31
740$      COUNT:      Record 31,A IDTrem == Rastreo:NEXT(743$)
  Model statements for module: Assign 666
743$      ASSIGN:      A Node=V Trecho ( (V ROT(Indice2 + A Indirota, A Rota)),
1):
          A NodeS=1:NEXT(747$)
  Model statements for module: Decide 572
747$      BRANCH,      1:
          If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreo.AND.Rastreo2<TNOW),2553$,Yes:
          Else,2554$,Yes
2553$      ASSIGN:      Ponto de checagem 35.NumberOut True=Ponto de checagem
35.NumberOut True + 1:NEXT(748$)
2554$      ASSIGN:      Ponto de checagem 35.NumberOut False=Ponto de checagem
35.NumberOut False + 1:NEXT(106$)
  Model statements for module: Separate 71
748$      DUPLICATE,    100 - 50:
          1,2557$,50:NEXT(2556$)
2556$      ASSIGN:      Separate 71.NumberOut Orig=Separate 71.NumberOut Orig +
1:NEXT(745$)
2557$      ASSIGN:      Separate 71.NumberOut Dup=Separate 71.NumberOut Dup +
1:NEXT(106$)
  Model statements for module: Hold 84
745$      QUEUE,      Hold 84.Queue:DETACH
  Model statements for module: Record 32
741$      COUNT:      Record 32,A IDTrem == Rastreo:NEXT(744$)
  Model statements for module: Assign 667
744$      ASSIGN:      A Node=V Trecho ( (V ROT(Indice2 + A Indirota, A Rota)),
1):
          A NodeS=2:NEXT(747$)
  Model statements for module: Decide 111
106$      BRANCH,      1:
          With,(( V Trecho( A Trecho , 6 + Temp1
)/100,2558$,Yes:
          Else,2559$,Yes
2558$      ASSIGN:      Primeiro movimento tera contingencia de via ?.NumberOut
True=
          Primeiro movimento tera contingencia de via ?.NumberOut
True + 1:NEXT(107$)
2559$      ASSIGN:      Primeiro movimento tera contingencia de via ?.NumberOut
False=
          Primeiro movimento tera contingencia de via ?.NumberOut
False + 1:NEXT(105$)
  Model statements for module: Assign 138
107$      ASSIGN:      A Tmp=((1 + (A Sentido==0)) * Temp1 ) + 6:
          A TransitTime=TT Trechos ( A Trecho, A Tmp):NEXT(572$)
  Model statements for module: Decide 366
572$      BRANCH,      1:
          If,A TransitTime <= 180,2560$,Yes:
          Else,2561$,Yes
2560$      ASSIGN:      Decide 366.NumberOut True=Decide 366.NumberOut True +
1:NEXT(241$)
2561$      ASSIGN:      Decide 366.NumberOut False=Decide 366.NumberOut False +
1:NEXT(573$)

```



```

Model statements for module: Decide 367
573$      BRANCH,      1:
                If,A Tmp>6,2562$,Yes:
                Else,2563$,Yes
2562$      ASSIGN:      Decide 367.NumberOut True=Decide 367.NumberOut True +
1:NEXT(574$)
2563$      ASSIGN:      Decide 367.NumberOut False=Decide 367.NumberOut False +
1:NEXT(575$)
Model statements for module: ReadWrite 68
574$      WRITE,      Log de Erros,"%1.8g. TT CONTINGENCIA de %1.8g. Trem tipo
%1.8g. Trecho %1.8g. Coluna %1.8g.\n":
                TNOW,
                A TransitTime,
                A Tipo,
                A Trecho,
                A Tmp - 6:NEXT(576$)
Model statements for module: Assign 439
576$      ASSIGN:      A TransitTime=180:NEXT(241$)
Model statements for module: ReadWrite 69
575$      WRITE,      Log de Erros,"%1.8g. TT NORMAL de %1.8g. Trem tipo
%1.8g. Trecho %1.8g. Coluna %1.8g.\n":
                TNOW,
                A TransitTime,
                A Tipo,
                A Trecho,
                A Tmp:NEXT(576$)
Model statements for module: Assign 438
241$      ASSIGN:      A Tmp=0:
                A TransitTime=A TransitTime * (1 * (A SouHelper==0) + E
Helpers(3,1) * (A SouHelper==1)):
                A Direto=0:NEXT(158$)
Model statements for module: Decide 254
158$      BRANCH,      1:
                If,V Destino ( A PtAtual,1 + (A Sentido==0)) ==
0,2564$,Yes:
                Else,2565$,Yes
2564$      ASSIGN:      Proximo patio nao ocorrera ultrapassagem ?.NumberOut
True=
                Proximo patio nao ocorrera ultrapassagem ?.NumberOut
True + 1:NEXT(304$)
2565$      ASSIGN:      Proximo patio nao ocorrera ultrapassagem ?.NumberOut
False=
                Proximo patio nao ocorrera ultrapassagem ?.NumberOut
False + 1:NEXT(3$)
Model statements for module: Decide 578
304$      BRANCH,      1:
                If,A Restricao2 == 1 .AND. V Trecho ( A TmpTrecho ,
3)==1,2566$,Yes:
                Else,2567$,Yes
2566$      ASSIGN:      Trem restritivo vai por SB ?.NumberOut True=Trem
restritivo vai por SB ?.NumberOut True + 1
                :NEXT(305$)
2567$      ASSIGN:      Trem restritivo vai por SB ?.NumberOut False=Trem
restritivo vai por SB ?.NumberOut False + 1
                :NEXT(327$)
Model statements for module: Assign 688
305$      ASSIGN:      A EstAtual=M:NEXT(329$)
Model statements for module: Decide 604
329$      BRANCH,      1:
                If,V Destino ( A PtAtual,1 + (A TmpSentido==0)) ==
0,2568$,Yes:
                Else,2569$,Yes
2568$      ASSIGN:      Direto SB nao esta ultrapassando neste patio via
SB?.NumberOut True=

```

```

Direto SB nao esta ultrapassando neste patio via
SB?.NumberOut True + 1:NEXT(330$)
2569$ ASSIGN: Direto SB nao esta ultrapassando neste patio via
SB?.NumberOut False=
Direto SB nao esta ultrapassando neste patio via
SB?.NumberOut False + 1:NEXT(306$)
Model statements for module: Assign 702
330$ ASSIGN: V Ocpt ( A PtAtual , 1+ ( A Sentido == 0 ))=0:NEXT(306$)
Model statements for module: Route 32
306$ ROUTE: 0,Restricao por SB
Model statements for module: Decide 595
327$ BRANCH, 1:
If,E Verifica entroncamento,2570$,Yes:
Else,2571$,Yes
2570$ ASSIGN: Vai para entroncamento ?.NumberOut True=Vai para
entroncamento ?.NumberOut True + 1:NEXT(60$)
2571$ ASSIGN: Vai para entroncamento ?.NumberOut False=Vai para
entroncamento ?.NumberOut False + 1:NEXT(324$)
Model statements for module: Route 6
60$ ROUTE: A TransitTime,ESTA ( A PtAtual,1 + ( A Sentido == 0 ) )
Model statements for module: Assign 696
324$ ASSIGN: V Reserva ( A TmpTrecho, 1+(A TmpSentido==0) )=
V Reserva ( A TmpTrecho, 1+(A TmpSentido==0) ) - A
Restricao:NEXT(60$)
Model statements for module: Assign 334
3$ ASSIGN: Sorteio=Unif(21000,25000):
A Ultrapassa=1:NEXT(2$)
Model statements for module: Separate 29
2$ DUPLICATE, 100 - 50:
1,2574$,50:NEXT(2573$)
2573$ ASSIGN: Separate 29.NumberOut Orig=Separate 29.NumberOut Orig +
1:NEXT(0$)
2574$ ASSIGN: Separate 29.NumberOut Dup=Separate 29.NumberOut Dup +
1:NEXT(304$)
Model statements for module: Assign 333
0$ ASSIGN: Picture=Picture.ultrapassa:
A PtAtual=A Patio:
A Sentido=A Sentido == 0:NEXT(1$)
Model statements for module: Route 20
1$ ROUTE: 0.000000000000000,Station Anima Evento
Model statements for module: Assign 137
105$ ASSIGN: A Tmp=(1 + ( A Sentido==0 )) * Temp1:
A TransitTime=TT Trechos ( A Trecho , A Tmp ):NEXT(572$)
Model statements for module: Decide 633
352$ BRANCH, 1:
If,V Gravar em texto == 1,2575$,Yes:
Else,2576$,Yes
2575$ ASSIGN: Decide 633.NumberOut True=Decide 633.NumberOut True +
1:NEXT(359$)
2576$ ASSIGN: Decide 633.NumberOut False=Decide 633.NumberOut False +
1:NEXT(218$)
Especificação do arquivo de LOG:
1 - Identificador único do trem (A IDTrem)
2 - Pátio (A PtAtual)
3 - Momento da entrada no pátio (A Tmp)
4 - Momento da saída do trem (TNOW)
5 - Número do trem (A NTrem)
6 - Horário do trem (A Horario)
7 - Tipo do trem (A Tipo)
8 - Gráfico onde entra esta informação (E Patios (A PtAtual,9))
Model statements for module: ReadWrite 163
359$ WRITE, Log de Movimentacao:
Dia,
A IDtrem,
N Trem,

```

```

A Horario,
A PtAtual,
A Tmp,
TNOW,
A Tipo,
A Trecho,
A Sentido,
E Patios (A PtAtual,9),
E Patios (A PtAtual,10),
A PtDestino,
A Produto,
A Cliente,
tnow - marca:NEXT(285$)
Especificação do arquivo de LOG:
1 - Identificador único do trem (A IDTrem)
2 - Pátio (A PtAtual)
3 - Momento da entrada no pátio (A Tmp)
4 - Momento da saída do trem (TNOW)
5 - Número do trem (A NTrem)
6 - Horário do trem (A Horario)
7 - Tipo do trem (A Tipo)
8 - Gráfico onde entra esta informação (E Patios (A PtAtual,9))
Model statements for module: ReadWrite 66
218$      WRITE,          Log de MovimentacaoMDB,RECORDSET(LogdeMovimentacao):
          Dia,
          A IDtrem,
          N Trem,
          A Horario,
          A PtAtual,
          A Tmp,
          TNOW,
          A Tipo,
          A Trecho,
          A Sentido,
          E Patios (A PtAtual,9),
          E Patios (A PtAtual,10),
          A PtDestino,
          A Produto,
          A Cliente,
          tnow - marca:NEXT(285$)
          Model statements for module: Decide 303
568$      BRANCH,          1:
          If, checar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreo.AND.Rastreo2<TNOW),2577$,Yes:
          Else,2578$,Yes
2577$      ASSIGN:          Ponto de checagem 18.NumberOut True=Ponto de checagem
18.NumberOut True + 1:NEXT(569$)
2578$      ASSIGN:          Ponto de checagem 18.NumberOut False=Ponto de checagem
18.NumberOut False + 1:NEXT(31$)
          Model statements for module: Separate 43
569$      DUPLICATE,          100 - 50:
          1,2581$,50:NEXT(2580$)
2580$      ASSIGN:          Separate 43.NumberOut Orig=Separate 43.NumberOut Orig +
1:NEXT(570$)
2581$      ASSIGN:          Separate 43.NumberOut Dup=Separate 43.NumberOut Dup +
1:NEXT(31$)
          Model statements for module: Hold 52
570$      QUEUE,          Hold 52.Queue:DETACH
O pátio é parte integrante do trecho.
Portanto, Indice2 é incrementado, mas primeiro será
bloqueada a linha entre o pátio que acaba de ser bloqueado
e o próximo pátio.
Este valor só será usado no Assign "Proximo trecho",
para que se bloqueia o pátio seguinte .
Model statements for module: Assign 80

```

```

31$          ASSIGN:          Indice2=Indice2 + 1:NEXT(113$)
      Model statements for module:  Decide 594
326$          BRANCH,          1:
      If,E Verifica entroncamento A Patio,2582$,Yes:
      Else,2583$,Yes
2582$          ASSIGN:          Passando por entroncamento ?.NumberOut True=Passando por
entroncamento ?.NumberOut True + 1
      :NEXT(365$)
2583$          ASSIGN:          Passando por entroncamento ?.NumberOut False=Passando
por entroncamento ?.NumberOut False + 1
      :NEXT(32$)
      Model statements for module:  Assign 727
365$          ASSIGN:          V CronoPPT(A TmpTrecho,Temp1)=tnow:NEXT(29$)
      Model statements for module:  Assign 81
32$          ASSIGN:          V Sentido (A TmpTrecho)=A TmpSentido:
      V Reserva (A TmpTrecho, 1+(A TmpSentido==0) )=
      V Reserva (A TmpTrecho, 1+(A TmpSentido==0) ) + A

Restricao:
      V CronoPPT(A TmpTrecho,Temp1)=tnow:NEXT(29$)
      O pátio é parte integrante do trecho.
      Portanto, Indice2 é incrementado, mas primeiro será
      verificada a linha entre o pátio que acaba de ser checado
      e o próximo pátio.
      Este valor só será usado no Assign "Proximo trecho",
      para que se verifique o pátio seguinte .
      Model statements for module:  Assign 77
26$          ASSIGN:          Indice2=Indice2 + 1:NEXT(80$)
      Model statements for module:  Decide 280
189$          BRANCH,          1:
      If,A IDTrem <> Rastreio,2584$,Yes:
      Else,2585$,Yes
2584$          ASSIGN:          Nao atende patio.NumberOut True=Nao atende
patio.NumberOut True + 1:NEXT(197$)
2585$          ASSIGN:          Nao atende patio.NumberOut False=Nao atende
patio.NumberOut False + 1:NEXT(197$)
      Model statements for module:  Release 13
197$          RELEASE:          Passo3,1:NEXT(198$)
      Model statements for module:  Delay 40
198$          DELAY:          UNIF (0.5,2) + (A Prioridade2*4) ,, Other:NEXT(257$)
      Model statements for module:  Decide 425
257$          BRANCH,          1:
      If,A RestFila <> 0,2586$,Yes:
      Else,2587$,Yes
2586$          ASSIGN:          Restricao devido a Fila ?.NumberOut True=Restricao
devido a Fila ?.NumberOut True + 1:NEXT(258$)
2587$          ASSIGN:          Restricao devido a Fila ?.NumberOut False=Restricao
devido a Fila ?.NumberOut False + 1:NEXT(20$)
      Model statements for module:  Decide 426
258$          BRANCH,          1:
      If,V Destino ( A RestFila,1 + (A Sentido==0) ) <> A
PtDestino,2588$,Yes:
      Else,2589$,Yes
2588$          ASSIGN:          A fila que restringia andou ?.NumberOut True=A fila que
restringia andou ?.NumberOut True + 1
      :NEXT(259$)
2589$          ASSIGN:          A fila que restringia andou ?.NumberOut False=A fila que
restringia andou ?.NumberOut False + 1
      :NEXT(20$)
      Model statements for module:  Assign 480
259$          ASSIGN:          A Repescagem=1:NEXT(7$)
      Model statements for module:  Decide 73
36$          BRANCH,          1:
      If,
      ( V ocbl ( A TmpTrecho , 1 ) + V ocbl ( A TmpTrecho , 2
) + V ocbl ( A TmpTrecho , 3 ) + V ocbl ( A TmpTrecho , 4 ) ) == 0,

```

```

2590$,Yes:
Else,2591$,Yes
2590$      ASSIGN:      Todas as secoes de bloqueio estao livres ?.NumberOut
True=
                Todas as secoes de bloqueio estao livres ?.NumberOut
True + 1:NEXT(88$)
2591$      ASSIGN:      Todas as secoes de bloqueio estao livres ?.NumberOut
False=
                Todas as secoes de bloqueio estao livres ?.NumberOut
False + 1:NEXT(193$)
        Model statements for module:  Decide 292
193$      BRANCH,      1:
                If,A IDTrem <> Rastreio,2592$,Yes:
                Else,2593$,Yes
2592$      ASSIGN:      Nao atende SBs livres.NumberOut True=Nao atende SBs
livres.NumberOut True + 1:NEXT(197$)
2593$      ASSIGN:      Nao atende SBs livres.NumberOut False=Nao atende SBs
livres.NumberOut False + 1:NEXT(298$)
        Model statements for module:  Assign 647
298$      ASSIGN:      sh15=
                ( V ocbl ( A TmpTrecho , 1 ) + V ocbl ( A TmpTrecho , 2
) + V ocbl ( A TmpTrecho , 3 ) + V ocbl ( A TmpTrecho , 4 ) )
                :NEXT(197$)
        Model statements for module:  Decide 289
190$      BRANCH,      1:
                If,V Trecho(A TmpTrecho,5) > 1,2594$,Yes:
                Else,2595$,Yes
2594$      ASSIGN:      Esta linha eh dupla ?.NumberOut True=Esta linha eh dupla
?.NumberOut True + 1:NEXT(191$)
2595$      ASSIGN:      Esta linha eh dupla ?.NumberOut False=Esta linha eh
dupla ?.NumberOut False + 1:NEXT(110$)
        Model statements for module:  Decide 290
191$      BRANCH,      1:
                If,V ocbl ( A TmpTrecho , 1 + ( 4 * ( A TmpSentido == 0) )
) == 0,2596$,Yes:
                Else,2597$,Yes
2596$      ASSIGN:      Primeira secao de bloqueio livre neste
sentido?.NumberOut True=
                Primeira secao de bloqueio livre neste
sentido?.NumberOut True + 1:NEXT(88$)
2597$      ASSIGN:      Primeira secao de bloqueio livre neste
sentido?.NumberOut False=
                Primeira secao de bloqueio livre neste
sentido?.NumberOut False + 1:NEXT(187$)
        Model statements for module:  Decide 278
187$      BRANCH,      1:
                If,A IDTrem <> Rastreio,2598$,Yes:
                Else,2599$,Yes
2598$      ASSIGN:      Nao atende SB.NumberOut True=Nao atende SB.NumberOut
True + 1:NEXT(197$)
2599$      ASSIGN:      Nao atende SB.NumberOut False=Nao atende SB.NumberOut
False + 1:NEXT(197$)
        Model statements for module:  Decide 113
110$      BRANCH,      1:
                If,V ocbl ( A TmpTrecho , 1 ) == 0,2600$,Yes:
                Else,2601$,Yes
2600$      ASSIGN:      Primeira secao de bloqueio livre ?.NumberOut True=
                Primeira secao de bloqueio livre ?.NumberOut True +
1:NEXT(393$)
2601$      ASSIGN:      Primeira secao de bloqueio livre ?.NumberOut False=
                Primeira secao de bloqueio livre ?.NumberOut False +
1:NEXT(187$)
        Model statements for module:  Decide 605
393$      BRANCH,      1:

```

```

If,A Restricao==0 .AND. (A Patio == A
PtDestino),2602$,Yes:
Else,2603$,Yes
2602$      ASSIGN:      Trem sem restricao tem destino no proximo patio ? Neste
caso pode ir em frente.NumberOut True=
Trem sem restricao tem destino no proximo patio ? Neste
caso pode ir em frente.NumberOut True + 1
:NEXT(25$)
2603$      ASSIGN:      Trem sem restricao tem destino no proximo patio ? Neste
caso pode ir em frente.NumberOut False=
Trem sem restricao tem destino no proximo patio ? Neste
caso pode ir em frente.NumberOut False + 1
:NEXT(370$)
1 = CRESCENTE
Próximo Pátio: Indirota+1
Trecho usado: Indirota
0 = DECRESCENTE
Próximo Pátio: Indirota+1
Trecho usado: Indirota+1
Model statements for module: Assign 362
370$      ASSIGN:      Indice3=0:
Temp1=0:NEXT(382$)
Model statements for module: Decide 314
382$      BRANCH,      1:
If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2604$,Yes:
Else,2605$,Yes
2604$      ASSIGN:      Ponto de checagem 19.NumberOut True=Ponto de checagem
19.NumberOut True + 1:NEXT(383$)
2605$      ASSIGN:      Ponto de checagem 19.NumberOut False=Ponto de checagem
19.NumberOut False + 1:NEXT(381$)
Model statements for module: Separate 45
383$      DUPLICATE,    100 - 50:
1,2608$,50:NEXT(2607$)
2607$      ASSIGN:      Separate 45.NumberOut Orig=Separate 45.NumberOut Orig +
1:NEXT(384$)
2608$      ASSIGN:      Separate 45.NumberOut Dup=Separate 45.NumberOut Dup +
1:NEXT(381$)
Model statements for module: Hold 54
384$      QUEUE,      Hold 54.Queue:DETACH
Model statements for module: Decide 313
381$      BRANCH,      1:
If,V ROT(Indicel+Indice2+1, A Rota) <> 0,2609$,Yes:
Else,2610$,Yes
2609$      ASSIGN:      Roteiro nao acaba no proximo patio ?.NumberOut True=
Roteiro nao acaba no proximo patio ?.NumberOut True +
1:NEXT(371$)
2610$      ASSIGN:      Roteiro nao acaba no proximo patio ?.NumberOut False=
Roteiro nao acaba no proximo patio ?.NumberOut False +
1:NEXT(390$)
1 = CRESCENTE
Próximo Pátio: Indirota+1
Trecho usado: Indirota
0 = DECRESCENTE
Próximo Pátio: Indirota+1
Trecho usado: Indirota+1
Model statements for module: Assign 363
371$      ASSIGN:      A BalSentido=V DROT( Indicel + indice3, A Rota):
A BalTrecho=V ROT(Indicel + indice3 + (A BalSentido==0),
A Rota):
A BalPatio=V Trecho (V ROT(Indicel+Indice3+1, A Rota) ,
2 ):NEXT(394$)
Model statements for module: Decide 680
394$      BRANCH,      1:
If,V Trecho(A BalTrecho,5)>1,2611$,Yes:

```

```

Else,2612$,Yes
2611$      ASSIGN:      Trecho duplo ?.NumberOut True=Trecho duplo ?.NumberOut
True + 1:NEXT(25$)
2612$      ASSIGN:      Trecho duplo ?.NumberOut False=Trecho duplo ?.NumberOut
False + 1:NEXT(372$)
      Model statements for module:  Decide 308
372$      BRANCH,      1:
If,V Sentido (A BalTrecho) <> A BalSentido .AND. V
Sentido (A BalTrecho) <> 2,2613$,Yes:
Else,2614$,Yes
2613$      ASSIGN:      Este trecho esta no sentido contrario ?.NumberOut True=
Este trecho esta no sentido contrario ?.NumberOut True +
1:NEXT(376$)
2614$      ASSIGN:      Este trecho esta no sentido contrario ?.NumberOut False=
Este trecho esta no sentido contrario ?.NumberOut False
+ 1:NEXT(373$)
      Model statements for module:  Decide 310
376$      BRANCH,      1:
If,Temp1 < Indice3,2615$,Yes:
Else,2616$,Yes
2615$      ASSIGN:      Pode seguir em frente sem deadlock ?.NumberOut True=
Pode seguir em frente sem deadlock ?.NumberOut True +
1:NEXT(386$)
2616$      ASSIGN:      Pode seguir em frente sem deadlock ?.NumberOut False=
Pode seguir em frente sem deadlock ?.NumberOut False +
1:NEXT(391$)
      Model statements for module:  Decide 315
386$      BRANCH,      1:
If,chechar >= 0 .AND. chechar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2617$,Yes:
Else,2618$,Yes
2617$      ASSIGN:      Ponto de checagem 20.NumberOut True=Ponto de checagem
20.NumberOut True + 1:NEXT(387$)
2618$      ASSIGN:      Ponto de checagem 20.NumberOut False=Ponto de checagem
20.NumberOut False + 1:NEXT(25$)
      Model statements for module:  Separate 46
387$      DUPLICATE,    100 - 50:
1,2621$,50:NEXT(2620$)
2620$      ASSIGN:      Separate 46.NumberOut Orig=Separate 46.NumberOut Orig +
1:NEXT(388$)
2621$      ASSIGN:      Separate 46.NumberOut Dup=Separate 46.NumberOut Dup +
1:NEXT(25$)
      Model statements for module:  Hold 55
388$      QUEUE,      Hold 55.Queue:DETACH
      Model statements for module:  Decide 317
391$      BRANCH,      1:
If,A IDTrem <> Rastreio,2622$,Yes:
Else,2623$,Yes
2622$      ASSIGN:      Deadlock detectado.NumberOut True=Deadlock
detectado.NumberOut True + 1:NEXT(188$)
2623$      ASSIGN:      Deadlock detectado.NumberOut False=Deadlock
detectado.NumberOut False + 1:NEXT(188$)
1 = CRESCENTE
Próximo Pátio: Indirota+1
Trecho usado: Indirota
0 = DECRESCENTE
Próximo Pátio: Indirota+1
Trecho usado: Indirota+1
      Model statements for module:  Assign 364
373$      ASSIGN:      Temp1=
Temp1 + V ocbl ( A BalTrecho , 1) + V ocbl ( A BalTrecho
, 2) + V ocbl ( A BalTrecho , 3) + V ocbl ( A BalTrecho , 4)
:NEXT(379$)
      Model statements for module:  Decide 311
379$      BRANCH,      1:

```

```

                If,
                E Patios (A BalPatio , 7 ) >= A Tam .OR. V Trecho(A
BalTrecho,5)>1 .OR. A BalPatio == A PtDestino,
                2624$,Yes:
                Else,2625$,Yes
2624$          ASSIGN:      Este patio esta ativado e o trem pode parar aqui
?.NumberOut True=
                Este patio esta ativado e o trem pode parar aqui
?.NumberOut True + 1:NEXT(380$)
2625$          ASSIGN:      Este patio esta ativado e o trem pode parar aqui
?.NumberOut False=
                Este patio esta ativado e o trem pode parar aqui
?.NumberOut False + 1:NEXT(378$)
                Model statements for module: Decide 312
380$          BRANCH,      1:
                If,
                (E Patios (A BalPatio , 8 ) .AND. A Tipo == 1) .OR. A
Tipo <> 1 .OR. V Trecho(A BalTrecho,5)>1 .OR. A BalPatio == A PtDestino .OR. A
Excecao==1,
                2626$,Yes:
                Else,2627$,Yes
2626$          ASSIGN:      Minerio carregado pode parar neste patio ?.NumberOut
True=
                Minerio carregado pode parar neste patio ?.NumberOut
True + 1:NEXT(377$)
2627$          ASSIGN:      Minerio carregado pode parar neste patio ?.NumberOut
False=
                Minerio carregado pode parar neste patio ?.NumberOut
False + 1:NEXT(378$)
                1 = CRESCENTE
                Próximo Pátio: Indirota+1
                Trecho usado: Indirota
                0 = DECRESCENTE
                Próximo Pátio: Indirota+1
                Trecho usado: Indirota+1
                Model statements for module: Assign 366
377$          ASSIGN:      Templ=Templ + V ocpt (A BalPatio,1+ ( A BalSentido == 0)
):NEXT(374$)
                Model statements for module: Assign 365
374$          ASSIGN:      Indice3=Indice3 + 1:
                A BalSentido=V DROT( Indicel + indice3, A
Rota):NEXT(375$)
                Model statements for module: Decide 309
375$          BRANCH,      1:
                If,V ROT(Indicel + indice3 + 1, A Rota) == 0,2628$,Yes:
                Else,2629$,Yes
2628$          ASSIGN:      Roteiro acaba antes deste trecho ?.NumberOut True=
                Roteiro acaba antes deste trecho ?.NumberOut True +
1:NEXT(386$)
2629$          ASSIGN:      Roteiro acaba antes deste trecho ?.NumberOut False=
                Roteiro acaba antes deste trecho ?.NumberOut False +
1:NEXT(392$)
                Model statements for module: Decide 318
392$          BRANCH,      1:
                If,A IDTrem <> Rastreio,2630$,Yes:
                Else,2631$,Yes
2630$          ASSIGN:      Verificou SB novamente.NumberOut True=Verificou SB
novamente.NumberOut True + 1:NEXT(371$)
2631$          ASSIGN:      Verificou SB novamente.NumberOut False=Verificou SB
novamente.NumberOut False + 1:NEXT(371$)
                1 = CRESCENTE
                Próximo Pátio: Indirota+1
                Trecho usado: Indirota
                0 = DECRESCENTE
                Próximo Pátio: Indirota+1

```



```

Trecho usado: Indirota+1
Model statements for module: Assign 367
378$    ASSIGN:      Templ=Templ + 1:NEXT(374$)
Model statements for module: Decide 316
390$    BRANCH,      1:
                          If,A IDTrem <> Rastreio,2632$,Yes:
                          Else,2633$,Yes
2632$    ASSIGN:      Nao verifica SB.NumberOut True=Nao verifica SB.NumberOut
True + 1:NEXT(25$)
2633$    ASSIGN:      Nao verifica SB.NumberOut False=Nao verifica
SB.NumberOut False + 1:NEXT(25$)
Model statements for module: Decide 279
188$    BRANCH,      1:
                          If,A IDTrem <> Rastreio,2634$,Yes:
                          Else,2635$,Yes
2634$    ASSIGN:      Nao atende deadlock.NumberOut True=Nao atende
deadlock.NumberOut True + 1:NEXT(197$)
2635$    ASSIGN:      Nao atende deadlock.NumberOut False=Nao atende
deadlock.NumberOut False + 1:NEXT(197$)
Model statements for module: Decide 277
186$    BRANCH,      1:
                          If,A IDTrem <> Rastreio,2636$,Yes:
                          Else,2637$,Yes
2636$    ASSIGN:      Nao atende linha.NumberOut True=Nao atende
linha.NumberOut True + 1:NEXT(197$)
2637$    ASSIGN:      Nao atende linha.NumberOut False=Nao atende
linha.NumberOut False + 1:NEXT(197$)
Model statements for module: Decide 276
185$    BRANCH,      1:
                          If,A IDTrem <> Rastreio,2638$,Yes:
                          Else,2639$,Yes
2638$    ASSIGN:      Nao atende sentido.NumberOut True=Nao atende
sentido.NumberOut True + 1:NEXT(197$)
2639$    ASSIGN:      Nao atende sentido.NumberOut False=Nao atende
sentido.NumberOut False + 1:NEXT(197$)
Model statements for module: Decide 503
283$    BRANCH,      1:
                          If,A FilaNode,2640$,Yes:
                          Else,2641$,Yes
2640$    ASSIGN:      Verifica se esta no entroncamento para sair da
fila.NumberOut True=
                          Verifica se esta no entroncamento para sair da
fila.NumberOut True + 1:NEXT(284$)
2641$    ASSIGN:      Verifica se esta no entroncamento para sair da
fila.NumberOut False=
                          Verifica se esta no entroncamento para sair da
fila.NumberOut False + 1:NEXT(157$)
Model statements for module: Assign 600
284$    ASSIGN:      V NodeD ( V Trecho (V ROT(A Indirota + (A Sentido ==0),
A Rota),1) , 1 + (A Sentido==0))=0:
                          V NodeDP ( V Trecho (V ROT(A Indirota + (A Sentido ==0),
A Rota),1) , 1 + (A Sentido==0))=0:
                          A FilaNode=0:NEXT(7$)
Model statements for module: Assign 330
157$    ASSIGN:      V Destino ( A PtAtual,1 + (A Sentido==0))=0:
                          V DestinoP ( A PtAtual,1 + (A Sentido==0))=0:
                          A FilaNode=0:NEXT(300$)
Model statements for module: Delay 54
300$    DELAY:      unif(.5,2)/2,,Other:NEXT(7$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)

```

```

5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saída de grade ou expresso
5=outras)
Model statements for module: ReadWrite 63
216$      WRITE,          Log de MovimentacaoMDB,RECORDSET(LogdeEvento):
          Dia,
          A IDtrem,
          A Ntrem,
          A Tipo,
          marca,
          A PtAtual,
          tnow,
          9 * (A Estado ==1) + 8 * (A Estado == 0),
          A Peso,
          A Dist,
          E FrotasT (A Ntrem, 1),
          E FrotasT (A Ntrem, 2),
          E FrotasT (A Ntrem, 3),
          E FrotasT (A Ntrem, 4),
          E FrotasT (A Ntrem, 5),
          A Estado,
          A Produto,
          A Cliente,
          A PtOrigem,
          A PtDestino,
          A PtZero:NEXT(217$)
Model statements for module: Search 19
295$      FINDJ,          1,50:V IndTrecho(J,2) == A Trecho + (A Sentido==0)
2642$      BRANCH,       1:
          If,J<>0,2643$,Yes:
          Else,2644$,Yes
2643$      DELAY:         0.0,,VA:NEXT(297$)
2644$      DELAY:         0.0,,VA:NEXT(294$)
Model statements for module: Assign 642
297$      ASSIGN:        X NodeS(Indice)=2:NEXT(293$)
Model statements for module: Assign 640
294$      ASSIGN:        X Node(Indice)=0:
          X NodeS(Indice)=0:NEXT(16$)
Model statements for module: Decide 496
273$      BRANCH,       1:
          If,E Verifica entroncamento A Patio,2645$,Yes:
          Else,2646$,Yes
2645$      ASSIGN:       A Patio em teste esta em entroncamento ?.NumberOut True=
          A Patio em teste esta em entroncamento ?.NumberOut True
+ 1:NEXT(274$)
2646$      ASSIGN:       A Patio em teste esta em entroncamento ?.NumberOut
False=
          A Patio em teste esta em entroncamento ?.NumberOut False
+ 1:NEXT(152$)
Model statements for module: Decide 497
274$      BRANCH,       1:
          If,V NodeD ( V Trecho (A Trecho,1) , 1 + (A Sentido==0))
== 0,2647$,Yes:
          Else,2648$,Yes
2647$      ASSIGN:       Nao existe fila neste entroncamento para o trecho de
destino ?.NumberOut True=
          Nao existe fila neste entroncamento para o trecho de
destino ?.NumberOut True + 1:NEXT(205$)
2648$      ASSIGN:       Nao existe fila neste entroncamento para o trecho de
destino ?.NumberOut False=
          Nao existe fila neste entroncamento para o trecho de
destino ?.NumberOut False + 1:NEXT(275$)

```

```

Model statements for module: Decide 499
275$      BRANCH,      1:
                If,V NodeD ( V Trecho (A Trecho,1) , 1 + (A Sentido==0))
== A PtDestino,2649$,Yes:
                Else,2650$,Yes
2649$      ASSIGN:      Este trem do entroncamento vai para o mesmo terminal
?.NumberOut True=
                Este trem do entroncamento vai para o mesmo terminal
?.NumberOut True + 1:NEXT(280$)
2650$      ASSIGN:      Este trem do entroncamento vai para o mesmo terminal
?.NumberOut False=
                Este trem do entroncamento vai para o mesmo terminal
?.NumberOut False + 1:NEXT(256$)
Model statements for module: Assign 598
280$      ASSIGN:      Temp2=A Produto * (A Produto <> 2) + 1 * (A Produto ==
2):NEXT(279$)
Model statements for module: Decide 501
279$      BRANCH,      1:
                If,V NodeDP ( V Trecho (A Trecho,1) , 1 + (A
Sentido==0)) ==Temp2,2651$,Yes:
                Else,2652$,Yes
2651$      ASSIGN:      Mesmo produto no entroncamento?.NumberOut True=Mesmo
produto no entroncamento?.NumberOut True + 1
                :NEXT(281$)
2652$      ASSIGN:      Mesmo produto no entroncamento?.NumberOut False=Mesmo
produto no entroncamento?.NumberOut False + 1
                :NEXT(246$)
Model statements for module: Decide 502
281$      BRANCH,      1:
                If,A FilaNode,2653$,Yes:
                Else,2654$,Yes
2653$      ASSIGN:      Verifica se esta no entroncamento para sinalizar
fila.NumberOut True=
                Verifica se esta no entroncamento para sinalizar
fila.NumberOut True + 1:NEXT(282$)
2654$      ASSIGN:      Verifica se esta no entroncamento para sinalizar
fila.NumberOut False=
                Verifica se esta no entroncamento para sinalizar
fila.NumberOut False + 1:NEXT(276$)
Model statements for module: Assign 599
282$      ASSIGN:      V NodeD ( V Trecho (V ROT(A Indirota + (A Sentido ==0),
A Rota),1) , 1 + (A Sentido==0))=
                A PtDestino:
                A Entrou em Fila=1:
                marca=tnow:
                V NodeDP ( V Trecho (V ROT(A Indirota + (A Sentido ==0),
A Rota),1) , 1 + (A Sentido==0))=Temp2
                :NEXT(277$)
Model statements for module: Decide 500
277$      BRANCH,      1:
                If,V NodeD ( V Trecho (A Trecho,1) , 1 + (A Sentido==0))
<> A PtDestino,2655$,Yes:
                Else,2656$,Yes
2655$      ASSIGN:      Entroncamento fila andou ?.NumberOut True=Entroncamento
fila andou ?.NumberOut True + 1:NEXT(283$)
2656$      ASSIGN:      Entroncamento fila andou ?.NumberOut False=Entroncamento
fila andou ?.NumberOut False + 1
                :NEXT(278$)
Model statements for module: Delay 51
278$      DELAY:      unif(.5,2),,Other:NEXT(277$)
Model statements for module: Assign 596
276$      ASSIGN:      A Trecho=V ROT(A Indirota + (A Sentido ==0), A Rota):
                V Destino ( A PtAtual,1 + (A Sentido==0))=A PtDestino:
                A Entrou em Fila=1:
                marca=tnow:

```

```

V DestinoP ( A PtAtual,1 + (A
Sentido==0))=Temp2:NEXT(277$)
  Model statements for module: Decide 388
246$      BRANCH,      1:
                If,A Aviso == 0,2657$,Yes:
                Else,2658$,Yes
2657$      ASSIGN:      Ainda nao recebeu aviso ?.NumberOut True=Ainda nao
recebeu aviso ?.NumberOut True + 1:NEXT(245$)
2658$      ASSIGN:      Ainda nao recebeu aviso ?.NumberOut False=Ainda nao
recebeu aviso ?.NumberOut False + 1:NEXT(34$)
  Model statements for module: Assign 444
245$      ASSIGN:      A Aviso=A Patio:NEXT(34$)
  Model statements for module: Assign 89
34$      ASSIGN:      A Restricao=1:NEXT(286$)
  Model statements for module: Search 15
286$      FINDJ,      1,50:V IndTrecho(J,1) == A Trecho + (A Sentido==0)
2659$      BRANCH,      1:
                If,J<>0,2660$,Yes:
                Else,2661$,Yes
2660$      DELAY:      0.0,,VA:NEXT(290$)
2661$      DELAY:      0.0,,VA:NEXT(289$)
  Model statements for module: Assign 637
290$      ASSIGN:      X NodeS(Indice)=1:NEXT(287$)
  Model statements for module: Assign 635
287$      ASSIGN:      X Node(Indice)=J:NEXT(204$)
  Model statements for module: Assign 376
204$      ASSIGN:      X Sentido(Indice)=A Sentido:
                X Trecho(Indice)=A Trecho:
                X Patio(Indice)=A Patio:NEXT(11$)
  Model statements for module: Decide 57
11$      BRANCH,      1:
                If,V ROT(Indice1 + 1, A Rota) == 0,2662$,Yes:
                Else,2663$,Yes
2662$      ASSIGN:      Este foi o ultimo patio do roteiro ?.NumberOut True=
                Este foi o ultimo patio do roteiro ?.NumberOut True +
1:NEXT(228$)
2663$      ASSIGN:      Este foi o ultimo patio do roteiro ?.NumberOut False=
                Este foi o ultimo patio do roteiro ?.NumberOut False +
1:NEXT(325$)
  Model statements for module: Decide 361
228$      BRANCH,      1:
                If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2664$,Yes:
                Else,2665$,Yes
2664$      ASSIGN:      Ponto de checagem 22.NumberOut True=Ponto de checagem
22.NumberOut True + 1:NEXT(229$)
2665$      ASSIGN:      Ponto de checagem 22.NumberOut False=Ponto de checagem
22.NumberOut False + 1:NEXT(13$)
  Model statements for module: Separate 53
229$      DUPLICATE,      100 - 50:
                1,2668$,50:NEXT(2667$)
2667$      ASSIGN:      Separate 53.NumberOut Orig=Separate 53.NumberOut Orig +
1:NEXT(230$)
2668$      ASSIGN:      Separate 53.NumberOut Dup=Separate 53.NumberOut Dup +
1:NEXT(13$)
  Model statements for module: Hold 67
230$      QUEUE,      Hold 67.Queue:DETACH
  Model statements for module: ReadWrite 10
13$      WRITE,      Log de Erros,
                "%1.8g. ERR3 Erro no trem %1.8g. Tipo %1.8g. Numero
%1.8g. Roteiro %1.8g finalizado sem encontrar a estacao de destino %1.8g. Trem
abortado no patio %1.8g.\n":
                TNOW,
                A IDtrem,
                A Tipo,

```

```

A Ntrem,
A Rota,
A PtDestino,
A PtAtual:NEXT(121$)
Model statements for module: Signal 15
121$ SIGNAL: A IDtrem,1:NEXT(14$)
Model statements for module: Assign 72
14$ ASSIGN: A Trecho=V ROT(A IndiRota - (A Sentido == 0), A Rota):
V Ocpt ( A PtAtual , 1+ (A Sentido==0) )=0:NEXT(178$)
Model statements for module: Route 25
178$ ROUTE: 0.000000000000000,Erro na movimentacao
Model statements for module: Decide 593
325$ BRANCH, 1:
If,V ROT(Indicel + 2 , A Rota) == 0,2669$,Yes:
Else,2670$,Yes
2669$ ASSIGN: Proximo trecho do roteiro eh o ultimo ?.NumberOut True=
Proximo trecho do roteiro eh o ultimo ?.NumberOut True +
1:NEXT(15$)
2670$ ASSIGN: Proximo trecho do roteiro eh o ultimo ?.NumberOut False=
Proximo trecho do roteiro eh o ultimo ?.NumberOut False
+ 1:NEXT(119$)
Model statements for module: Assign 73
15$ ASSIGN: Indice=Indice + 1:
Indicel=Indicel + 1:NEXT(9$)
Model statements for module: Decide 141
119$ BRANCH, 1:
If,V Trecho ( V Rot (Indicel + 1, A Rota) , 2 ) == V
Trecho ( V Rot (Indicel + 2, A Rota) , 2 ),
2671$,Yes:
Else,2672$,Yes
2671$ ASSIGN: 4 Atingiu uma juncao de segmentos ?.NumberOut True=
4 Atingiu uma juncao de segmentos ?.NumberOut True +
1:NEXT(206$)
2672$ ASSIGN: 4 Atingiu uma juncao de segmentos ?.NumberOut False=
4 Atingiu uma juncao de segmentos ?.NumberOut False +
1:NEXT(15$)
Model statements for module: Assign 379
206$ ASSIGN: Indicel=Indicel + 1:
A Nsb=A Nsb + 1:NEXT(15$)
Model statements for module: Search 17
289$ FINDJ, 1,50:V IndTrecho(J,2) == A Trecho + (A Sentido==0)
2673$ BRANCH, 1:
If,J<>0,2674$,Yes:
Else,2675$,Yes
2674$ DELAY: 0.0,,VA:NEXT(291$)
2675$ DELAY: 0.0,,VA:NEXT(288$)
Model statements for module: Assign 638
291$ ASSIGN: X NodeS(Indice)=2:NEXT(287$)
Model statements for module: Assign 636
288$ ASSIGN: X Node(Indice)=0:
X NodeS(Indice)=0:NEXT(204$)
Model statements for module: Assign 479
256$ ASSIGN: A RestFila=A RestFila * (A RestFila<>0) + A Patio * (A
RestFila==0):
A RestFila=A RestFila:NEXT(34$)
Model statements for module: Decide 235
152$ BRANCH, 1:
If,V Destino ( A Patio,1 + (A Sentido==0) ) ==
0,2676$,Yes:
Else,2677$,Yes
2676$ ASSIGN: Patio nao esta ocupado por trem aguardando terminal
?.NumberOut True=
Patio nao esta ocupado por trem aguardando terminal
?.NumberOut True + 1:NEXT(205$)

```

```

2677$      ASSIGN:      Patio nao esta ocupado por trem aguardando terminal
?.NumberOut False=
                Patio nao esta ocupado por trem aguardando terminal
?.NumberOut False + 1:NEXT(249$)
    Model statements for module:  Decide 419
249$      BRANCH,      1:
                If,V Destino ( A Patio,1 + (A Sentido==0)) ==
6000,2678$,Yes:
                Else,2679$,Yes
2678$      ASSIGN:      Fila para FVR ?.NumberOut True=Fila para FVR ?.NumberOut
True + 1:NEXT(250$)
2679$      ASSIGN:      Fila para FVR ?.NumberOut False=Fila para FVR
?.NumberOut False + 1:NEXT(153$)
    Model statements for module:  Assign 469
250$      ASSIGN:      V Destino ( A PtAtual,1 + (A Sentido==0))=6000:
                marca=tnow:
                V DestinoP ( A PtAtual,1 + (A Sentido==0))=0:NEXT(251$)
    Model statements for module:  Decide 420
251$      BRANCH,      1:
                If,V Destino ( A Patio,1 + (A Sentido==0)) <>
6000,2680$,Yes:
                Else,2681$,Yes
2680$      ASSIGN:      Fila para FVR andou ?.NumberOut True=Fila para FVR andou
?.NumberOut True + 1:NEXT(157$)
2681$      ASSIGN:      Fila para FVR andou ?.NumberOut False=Fila para FVR
andou ?.NumberOut False + 1:NEXT(252$)
    Model statements for module:  Delay 44
252$      DELAY:      unif(.5,2),,Other:NEXT(251$)
    Model statements for module:  Decide 249
153$      BRANCH,      1:
                If,V Destino ( A Patio,1 + (A Sentido==0)) == A
PtDestino,2682$,Yes:
                Else,2683$,Yes
2682$      ASSIGN:      Este trem vai para o mesmo terminal ?.NumberOut True=
                Este trem vai para o mesmo terminal ?.NumberOut True +
1:NEXT(269$)
2683$      ASSIGN:      Este trem vai para o mesmo terminal ?.NumberOut False=
                Este trem vai para o mesmo terminal ?.NumberOut False +
1:NEXT(256$)
    Model statements for module:  Assign 542
269$      ASSIGN:      Temp2=A Produto * (A Produto <> 2) + 1 * (A Produto ==
2):NEXT(244$)
    Model statements for module:  Decide 386
244$      BRANCH,      1:
                If,V DestinoP ( A Patio,1 + (A Sentido==0))
==Temp2,2684$,Yes:
                Else,2685$,Yes
2684$      ASSIGN:      Mesmo produto ?.NumberOut True=Mesmo produto ?.NumberOut
True + 1:NEXT(154$)
2685$      ASSIGN:      Mesmo produto ?.NumberOut False=Mesmo produto
?.NumberOut False + 1:NEXT(246$)
    Model statements for module:  Assign 329
154$      ASSIGN:      V Destino ( A PtAtual,1 + (A Sentido==0))=A PtDestino:
                A Entrou em Fila=1:
                marca=tnow:
                V DestinoP ( A PtAtual,1 + (A
Sentido==0))=Temp2:NEXT(155$)
    Model statements for module:  Decide 250
155$      BRANCH,      1:
                If,V Destino ( A Patio,1 + (A Sentido==0)) <> A
PtDestino,2686$,Yes:
                Else,2687$,Yes
2686$      ASSIGN:      Fila andou ?.NumberOut True=Fila andou ?.NumberOut True
+ 1:NEXT(157$)

```

```

2687$      ASSIGN:      Fila andou ?.NumberOut False=Fila andou ?.NumberOut
False + 1:NEXT(156$)
      Model statements for module:  Delay 38
156$      DELAY:      unif(.5,2),,Other:NEXT(155$)
      Model statements for module:  Decide 55
8$        BRANCH,      1:
                        If,E Patios (A Patio , 7 ) >= A Tam,2688$,Yes:
                        Else,2689$,Yes
2688$      ASSIGN:      Patio tem tamanho suficiente para este trem ?.NumberOut
True=
                        Patio tem tamanho suficiente para este trem ?.NumberOut
True + 1:NEXT(10$)
2689$      ASSIGN:      Patio tem tamanho suficiente para este trem ?.NumberOut
False=
                        Patio tem tamanho suficiente para este trem ?.NumberOut
False + 1:NEXT(194$)
      Model statements for module:  Decide 56
10$        BRANCH,      1:
                        If,(E Patios (A Patio , 8 ) .AND. A Tipo == 1) .OR. A
Tipo <> 1,2690$,Yes:
                        Else,2691$,Yes
2690$      ASSIGN:      Pode parar neste patio se for trem minerio carregado
?.NumberOut True=
                        Pode parar neste patio se for trem minerio carregado
?.NumberOut True + 1:NEXT(273$)
2691$      ASSIGN:      Pode parar neste patio se for trem minerio carregado
?.NumberOut False=
                        Pode parar neste patio se for trem minerio carregado
?.NumberOut False + 1:NEXT(34$)
      Model statements for module:  Decide 293
194$        BRANCH,      1:
                        If,A IDTrem <> Rastreio .or. A PtAtual<>30,2692$,Yes:
                        Else,2693$,Yes
2692$      ASSIGN:      Bloqueio por patio pequeno.NumberOut True=Bloqueio por
patio pequeno.NumberOut True + 1:NEXT(34$)
2693$      ASSIGN:      Bloqueio por patio pequeno.NumberOut False=Bloqueio por
patio pequeno.NumberOut False + 1:NEXT(34$)
      Model statements for module:  Assign 473
255$      ASSIGN:      A SgDestino=0:NEXT(7$)
      Model statements for module:  Decide 179
124$        BRANCH,      1:
                        If,V Trecho (A Trecho, 9) == 1,2694$,Yes:
                        Else,2695$,Yes
2694$      ASSIGN:      CPTM via Bras ?.NumberOut True=CPTM via Bras ?.NumberOut
True + 1:NEXT(125$)
2695$      ASSIGN:      CPTM via Bras ?.NumberOut False=CPTM via Bras
?.NumberOut False + 1:NEXT(127$)
      Model statements for module:  Hold 25
125$        QUEUE,      Aguarda recurso CPTM Bras ter capacidade 1.Queue
SCAN:      MR (CPTM Bras) > 0:NEXT(471$)
      Model statements for module:  Hold 26
127$        QUEUE,      Aguarda recurso CPTM Suzano ter capacidade 1.Queue
SCAN:      MR (CPTM Suzano) > 0:NEXT(471$)
      Model statements for module:  ReadWrite 35
144$        WRITE,      Log de Erros,
                        "%1.8g. ERR11 Erro no trem %1.8g. Tipo %1.8g. Roteiro
%1.8g finalizado sem encontrar a estacao de destino %1.8g. Verificar orientacao do
roteiro. Trem abortado no patio %1.8g. NTrem %1.8g.\n":
                        TNOW,
                        A IDtrem,
                        A Tipo,
                        A Rota,
                        A PtDestino,
                        A PtAtual,
                        A Ntrem:NEXT(146$)

```

```

Model statements for module: Signal 20
146$      SIGNAL:      A IDtrem,1:NEXT(145$)
Model statements for module: Assign 256
145$      ASSIGN:      A Trecho=V ROT(A IndiRota - (A Sentido == 0), A Rota):
          V Ocpt ( A PtAtual , 1+ (A Sentido==0) )=0:NEXT(177$)
Model statements for module: Route 24
177$      ROUTE:      0.000000000000000,Erro na movimentacao
Model statements for module: Assign 728
366$      ASSIGN:      V NewPPT(A TmpTrecho, (1+((temp1==2)*2)))=V NewPPT(A
TmpTrecho, (1+((temp1==2)*2))) + 1:NEXT(58$)
Model statements for module: Decide 80
58$       BRANCH,      1:
          If,
          (V ocbl ( A TmpTrecho , 1) + V ocbl ( A TmpTrecho , 2) +
V ocbl ( A TmpTrecho , 3) + V ocbl ( A TmpTrecho , 4) ) > 0 .AND. A PassouSB==1,
          239$,Yes:
          If,V Trecho(A TmpTrecho,5)>1,363$,Yes:
          Else,320$,Yes
Model statements for module: Decide 586
320$      BRANCH,      1:
          If,A Restricao2 == 1,2698$,Yes:
          Else,2699$,Yes
2698$     ASSIGN:      Trem chegando com restricao em SB ?.NumberOut True=
          Trem chegando com restricao em SB ?.NumberOut True +
1:NEXT(321$)
2699$     ASSIGN:      Trem chegando com restricao em SB ?.NumberOut False=
          Trem chegando com restricao em SB ?.NumberOut False +
1:NEXT(364$)
Model statements for module: Decide 587
321$      BRANCH,      1:
          If,V Reserva (A TmpTrecho, 1+(A TmpSentido==0) ) >
0,2700$,Yes:
          Else,2701$,Yes
2700$     ASSIGN:      Algum trem ainda reserva o trecho ?.NumberOut True=
          Algum trem ainda reserva o trecho ?.NumberOut True +
1:NEXT(239$)
2701$     ASSIGN:      Algum trem ainda reserva o trecho ?.NumberOut False=
          Algum trem ainda reserva o trecho ?.NumberOut False +
1:NEXT(364$)
Model statements for module: Decide 365
239$      BRANCH,      1:
          If,A PassouSB == 1,2702$,Yes:
          Else,2703$,Yes
2702$     ASSIGN:      Veio de SB ?.NumberOut True=Veio de SB ?.NumberOut True
+ 1:NEXT(118$)
2703$     ASSIGN:      Veio de SB ?.NumberOut False=Veio de SB ?.NumberOut
False + 1:NEXT(240$)
Model statements for module: Assign 150
118$      ASSIGN:      Temp1=1*(A Tipo<4) + 2*(A Tipo==4) + 3*(A Tipo==5):
          A PassouSB=0:NEXT(149$)
Model statements for module: Decide 220
149$      BRANCH,      1:
          With,(E ContPatios( A PtAtual , Temp1))/100,2704$,Yes:
          Else,2705$,Yes
2704$     ASSIGN:      Este patio tera contingencia ?.NumberOut True=Este patio
tera contingencia ?.NumberOut True + 1
          :NEXT(396$)
2705$     ASSIGN:      Este patio tera contingencia ?.NumberOut False=Este
patio tera contingencia ?.NumberOut False + 1
          :NEXT(587$)
Model statements for module: Assign 223
396$      ASSIGN:      Sorteio=Unif(1,4000)+Unif(1,350):
          marca=tnow:NEXT(397$)
Model statements for module: Separate 22
397$      DUPLICATE,    100 - 50:

```



```

1,2708$,50:NEXT(2707$)
2707$      ASSIGN:      Separate 22.NumberOut Orig=Separate 22.NumberOut Orig +
1:NEXT(398$)
2708$      ASSIGN:      Separate 22.NumberOut Dup=Separate 22.NumberOut Dup +
1:NEXT(409$)
    Model statements for module:  Assign 258
398$      ASSIGN:      Picture=Picture.contingencia:NEXT(399$)
    Model statements for module:  Hold 35
399$      QUEUE,      E FilasATV ( A PtAtual,1 + (A Sentido==0))
          WAIT:      sorteio,1:NEXT(401$)
    Model statements for module:  Dispose 48
401$      ASSIGN:      Dispose 48.NumberOut=Dispose 48.NumberOut + 1
2709$     DISPOSE:     Yes
    Model statements for module:  Decide 218
409$     BRANCH,      1:
          If,A SouHelper == 1,395$,Yes:
          If,A Tipo == 4,408$,Yes:
          If,A Tipo == 5,410$,Yes:
          Else,411$,Yes
    Model statements for module:  Assign 273
411$     ASSIGN:      V LocosNow(1)=V LocosNow(1) - E FrotasT (A Ntrem, 1):
          V LocosNow(2)=V LocosNow(2) - E FrotasT (A Ntrem, 2):
          V LocosNow(3)=V LocosNow(3) - E FrotasT (A Ntrem, 3):
          V LocosNow(4)=V LocosNow(4) - E FrotasT (A Ntrem, 4):
          V LocosNow(5)=V LocosNow(5) - E FrotasT (A Ntrem,
5):NEXT(395$)
    Model statements for module:  Delay 21
395$     DELAY:      E ContPatrios( A PtAtual , (3 + Temp1)),,Other:NEXT(405$)
    Model statements for module:  Decide 219
405$     BRANCH,      1:
          If,A SouHelper == 1,402$,Yes:
          If,A Tipo == 4,404$,Yes:
          If,A Tipo == 5,406$,Yes:
          Else,407$,Yes
    Model statements for module:  Assign 276
407$     ASSIGN:      V LocosNow(1)=V LocosNow(1) + E FrotasT (A Ntrem, 1):
          V LocosNow(2)=V LocosNow(2) + E FrotasT (A Ntrem, 2):
          V LocosNow(3)=V LocosNow(3) + E FrotasT (A Ntrem, 3):
          V LocosNow(4)=V LocosNow(4) + E FrotasT (A Ntrem, 4):
          V LocosNow(5)=V LocosNow(5) + E FrotasT (A Ntrem,
5):NEXT(402$)
    Model statements for module:  Signal 21
402$     SIGNAL:      sorteio,1:NEXT(413$)
    Model statements for module:  Decide 436
413$     BRANCH,      1:
          If,TNOW > (V Warmup * 1440),2714$,Yes:
          Else,2715$,Yes
2714$     ASSIGN:      Warmup do log de eventos 4.NumberOut True=Warmup do log
de eventos 4.NumberOut True + 1:NEXT(414$)
2715$     ASSIGN:      Warmup do log de eventos 4.NumberOut False=Warmup do log
de eventos 4.NumberOut False + 1
          :NEXT(587$)
    Model statements for module:  Decide 629
414$     BRANCH,      1:
          If,V Gravar em texto == 1,2716$,Yes:
          Else,2717$,Yes
2716$     ASSIGN:      Decide 629.NumberOut True=Decide 629.NumberOut True +
1:NEXT(415$)
2717$     ASSIGN:      Decide 629.NumberOut False=Decide 629.NumberOut False +
1:NEXT(403$)
    Especificação do arquivo de LOG de eventos:
    1 - Dia do evento
    2- Identificador único do trem (A IDTrem)
    3 - Número do trem (A NTrem)
    4 - Tipo do trem (A Tipo)

```

5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário - grade/expresso)  
 6 - Pátio (A PtAtual)  
 7 - Momento da saída do trem (TNOW)  
 8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso 5=outras)

Model statements for module: ReadWrite 156

```
415$      WRITE,          Log de Eventos:
                          Dia,
                          A IDtrem,
                          A Ntrem,
                          A Tipo,
                          marca,
                          A PtAtual,
                          tnow,
                          3,
                          0,
                          0,
                          0,
                          0,
                          0,
                          0,
                          0,
                          A Estado,
                          A Produto,
                          A Cliente,
                          A PtOrigem,
                          A PtDestino,
                          A PtZero:NEXT(587$)
```

Especificação do arquivo de LOG de eventos:

1 - Dia do evento  
 2- Identificador único do trem (A IDTrem)  
 3 - Número do trem (A NTrem)  
 4 - Tipo do trem (A Tipo)  
 5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário - grade/expresso)  
 6 - Pátio (A PtAtual)  
 7 - Momento da saída do trem (TNOW)  
 8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso 5=outras)

Model statements for module: ReadWrite 39

```
403$      WRITE,          Log de MovimentacaoMDB,RECORDSET(LogdeEvento):
                          Dia,
                          A IDtrem,
                          A Ntrem,
                          A Tipo,
                          marca,
                          A PtAtual,
                          tnow,
                          3,
                          0,
                          0,
                          0,
                          0,
                          0,
                          0,
                          0,
                          0,
                          A Estado,
                          A Produto,
                          A Cliente,
                          A PtOrigem,
                          A PtDestino,
                          A PtZero:NEXT(587$)
```

Model statements for module: Assign 274

```
404$      ASSIGN:        V LocosNow(1)=V LocosNow(1) + E FrotasG (A Ntrem, 1):
```

```

V LocosNow(2)=V LocosNow(2) + E FrotasG (A Ntrem, 2):
V LocosNow(3)=V LocosNow(3) + E FrotasG (A Ntrem, 3):
V LocosNow(4)=V LocosNow(4) + E FrotasG (A Ntrem, 4):
V LocosNow(5)=V LocosNow(5) + E FrotasG (A Ntrem,
5):NEXT(402$)
  Model statements for module: Assign 275
406$      ASSIGN:      V LocosNow(1)=V LocosNow(1) + E FrotasE (A Ntrem, 1):
V LocosNow(2)=V LocosNow(2) + E FrotasE (A Ntrem, 2):
V LocosNow(3)=V LocosNow(3) + E FrotasE (A Ntrem, 3):
V LocosNow(4)=V LocosNow(4) + E FrotasE (A Ntrem, 4):
V LocosNow(5)=V LocosNow(5) + E FrotasE (A Ntrem,
5):NEXT(402$)
  Model statements for module: Assign 271
408$      ASSIGN:      V LocosNow(1)=V LocosNow(1) - E FrotasG (A Ntrem, 1):
V LocosNow(2)=V LocosNow(2) - E FrotasG (A Ntrem, 2):
V LocosNow(3)=V LocosNow(3) - E FrotasG (A Ntrem, 3):
V LocosNow(4)=V LocosNow(4) - E FrotasG (A Ntrem, 4):
V LocosNow(5)=V LocosNow(5) - E FrotasG (A Ntrem,
5):NEXT(395$)
  Model statements for module: Assign 272
410$      ASSIGN:      V LocosNow(1)=V LocosNow(1) - E FrotasE (A Ntrem, 1):
V LocosNow(2)=V LocosNow(2) - E FrotasE (A Ntrem, 2):
V LocosNow(3)=V LocosNow(3) - E FrotasE (A Ntrem, 3):
V LocosNow(4)=V LocosNow(4) - E FrotasE (A Ntrem, 4):
V LocosNow(5)=V LocosNow(5) - E FrotasE (A Ntrem,
5):NEXT(395$)
  Model statements for module: Station 11
412$      STATION,      Station Anima Evento
2720$     DELAY:        0.0,,VA:NEXT(399$)
  Model statements for module: Decide 394
587$      BRANCH,      1:
If,A Tipo < 4 .AND. A Entrada==0,2721$,Yes:
Else,2722$,Yes
2721$     ASSIGN:      Este trem tabela ?.NumberOut True=Este trem tabela
?.NumberOut True + 1:NEXT(588$)
2722$     ASSIGN:      Este trem tabela ?.NumberOut False=Este trem tabela
?.NumberOut False + 1:NEXT(605$)
  Model statements for module: Decide 395
588$      BRANCH,      1:
If,
A PtAtual == 36 .AND. A Sentido==0 .AND. (E Patios (A
PtDestino,10)==10 .OR. E Patios (A PtDestino,10)==13),
600$,Yes:
If,A PtAtual == 38 .AND. A Sentido==0,592$,Yes:
Else,605$,Yes
  Model statements for module: Decide 401
605$      BRANCH,      1:
If,(A PtAtual==30 .OR. A PtAtual==45) .AND. A
FVR==1,2725$,Yes:
Else,2726$,Yes
2725$     ASSIGN:      Esta passando pelos extremos do trecho de FVR
?.NumberOut True=
Esta passando pelos extremos do trecho de FVR
?.NumberOut True + 1:NEXT(615$)
2726$     ASSIGN:      Esta passando pelos extremos do trecho de FVR
?.NumberOut False=
Esta passando pelos extremos do trecho de FVR
?.NumberOut False + 1:NEXT(56$)
  Model statements for module: Decide 416
615$      BRANCH,      1:
If,A PtAtual <> A PtDestino,2727$,Yes:
Else,2728$,Yes
2727$     ASSIGN:      Nao esta no patio de destino ?.NumberOut True=Nao esta
no patio de destino ?.NumberOut True + 1
:NEXT(611$)

```

```

2728$      ASSIGN:      Nao esta no patio de destino ?.NumberOut False=Nao esta
no patio de destino ?.NumberOut False + 1
                        :NEXT(56$)
      Model statements for module: Search 3
611$      FINDJ,      1,4:V ROT(A Indirota+J, A Rota) == 0
2729$      BRANCH,      1:
                        If,J<>0,2730$,Yes:
                        Else,2731$,Yes
2730$      DELAY:      0.0,,VA:NEXT(612$)
2731$      DELAY:      0.0,,VA:NEXT(613$)
      Model statements for module: Assign 463
612$      ASSIGN:      Templ=J-1:NEXT(614$)
      Model statements for module: Search 4
614$      FINDJ,      1,Templ:E Patios ( V Trecho (V ROT(A Indirota+J, A
Rota), 2) ,10) <> 12
2732$      BRANCH,      1:
                        If,J<>0,2733$,Yes:
                        Else,2734$,Yes
2733$      DELAY:      0.0,,VA:NEXT(607$)
2734$      DELAY:      0.0,,VA:NEXT(56$)
      Model statements for module: Decide 412
607$      BRANCH,      1:
                        If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2735$,Yes:
                        Else,2736$,Yes
2735$      ASSIGN:      Ponto de checagem 31.NumberOut True=Ponto de checagem
31.NumberOut True + 1:NEXT(608$)
2736$      ASSIGN:      Ponto de checagem 31.NumberOut False=Ponto de checagem
31.NumberOut False + 1:NEXT(606$)
      Model statements for module: Separate 61
608$      DUPLICATE,      100 - 50:
                        1,2739$,50:NEXT(2738$)
2738$      ASSIGN:      Separate 61.NumberOut Orig=Separate 61.NumberOut Orig +
1:NEXT(609$)
2739$      ASSIGN:      Separate 61.NumberOut Dup=Separate 61.NumberOut Dup +
1:NEXT(606$)
      Model statements for module: Hold 74
609$      QUEUE,      Hold 74.Queue:DETACH
      Model statements for module: Assign 458
606$      ASSIGN:      V ContagemFVR=V ContagemFVR - 1:
                        A FVR=0:NEXT(56$)
      Model statements for module: Assign 464
613$      ASSIGN:      Templ=4:NEXT(614$)
      Model statements for module: Decide 400
600$      BRANCH,      1:
                        If,
                        (V ocpt(44,2) + V ocpt(43,2) + V ocpt(42,2) + V
ocpt(41,2) + V ocpt(40,2) + V ocpt(39,2) + V ocpt(38,2) + E SentidosFGI) > 4,
2740$,Yes:
                        Else,2741$,Yes
2740$      ASSIGN:      Deve esperar 2?.NumberOut True=Deve esperar 2?.NumberOut
True + 1:NEXT(603$)
2741$      ASSIGN:      Deve esperar 2?.NumberOut False=Deve esperar
2?.NumberOut False + 1:NEXT(56$)
      Model statements for module: Seize 7
603$      QUEUE,      Ocupa controle de FGI 2.Queue
SEIZE,      2,Other:
                        Controle de FGI,1:NEXT(2743$)
2743$      DELAY:      0.0,,VA:NEXT(601$)
      Model statements for module: Hold 71
601$      QUEUE,      Aguarda espaco no trecho 2.Queue
SCAN:
                        (V ocpt(44,2) + V ocpt(43,2) + V ocpt(42,2) + V
ocpt(41,2) + V ocpt(40,2) + V ocpt(39,2) + V ocpt(38,2) + E SentidosFGI) <= 2
                        :NEXT(591$)

```

```

Model statements for module: Assign 446
591$      ASSIGN:      V ContagemFGI=V ContagemFGI + 1:NEXT(599$)
Model statements for module: Release 15
599$      RELEASE:     Controle de FGI,1:NEXT(56$)
Model statements for module: Assign 447
592$      ASSIGN:      V ContagemFGI=V ContagemFGI - 1:NEXT(56$)
Model statements for module: Decide 399
593$      BRANCH,      1:
                          If,V ContagemFGI >10,2744$,Yes:
                          Else,2745$,Yes
2744$      ASSIGN:      Deve esperar ?.NumberOut True=Deve esperar ?.NumberOut
True + 1:NEXT(597$)
2745$      ASSIGN:      Deve esperar ?.NumberOut False=Deve esperar ?.NumberOut
False + 1:NEXT(594$)
Model statements for module: Seize 6
597$      QUEUE,       Ocupa controle de FGI.Queue
          SEIZE,       2,Other:
                          Controle de FGI,1:NEXT(2747$)
2747$      DELAY:      0.0,,VA:NEXT(589$)
Model statements for module: Hold 70
589$      QUEUE,       Aguarda espaco no trecho.Queue
          SCAN:        V ContagemFGI < 11:NEXT(591$)
Model statements for module: Assign 449
594$      ASSIGN:      V ContagemFGI=V ContagemFGI + 1:NEXT(56$)
Model statements for module: Assign 451
595$      ASSIGN:      V Destino ( A PtAtual,1 + (A Sentido==0))=A PtDestino:
                          marca=tnow:
                          V DestinoP ( A PtAtual,1 + (A Sentido==0))=A
Produto:NEXT(597$)
Model statements for module: Assign 452
596$      ASSIGN:      V Destino ( A PtAtual,1 + (A Sentido==0))=0:
                          V DestinoP ( A PtAtual,1 + (A Sentido==0))=0:NEXT(599$)
Model statements for module: Assign 437
240$      ASSIGN:      Temp2=(4 * ( A TmpSentido == 0)) * ( V Trecho(A
TmpTrecho,5)>1):
                          V ocb1 ( A TmpTrecho , 1+temp2)=0:NEXT(118$)
Model statements for module: Decide 656
364$      BRANCH,      1:
                          If,V CronoPPT(A TmpTrecho,Temp1) ==0,2748$,Yes:
                          Else,2749$,Yes
2748$      ASSIGN:      Cronometro zerado neste trecho?.NumberOut
True=Cronometro zerado neste trecho?.NumberOut True + 1
                          :NEXT(239$)
2749$      ASSIGN:      Cronometro zerado neste trecho?.NumberOut
False=Cronometro zerado neste trecho?.NumberOut False + 1
                          :NEXT(59$)
Model statements for module: Assign 103
59$       ASSIGN:      V Sentido ( A TmpTrecho )=2:
                          V NewPPT(A TmpTrecho, (2+((temp1==2)*2)))=TNOW - V
CronoPPT(A TmpTrecho,Temp1):
                          V CronoPPT(A TmpTrecho,Temp1)=0:NEXT(239$)
Model statements for module: Decide 655
363$      BRANCH,      1:
                          If,V CronoPPT(A TmpTrecho,Temp1) ==0,2750$,Yes:
                          Else,2751$,Yes
2750$      ASSIGN:      Cronometro zerado ?.NumberOut True=Cronometro zerado
?.NumberOut True + 1:NEXT(239$)
2751$      ASSIGN:      Cronometro zerado ?.NumberOut False=Cronometro zerado
?.NumberOut False + 1:NEXT(362$)
Model statements for module: Assign 724
362$      ASSIGN:      V NewPPT(A TmpTrecho, (2+((temp1==2)*2)))=TNOW - V
CronoPPT(A TmpTrecho,Temp1):
                          V CronoPPT(A TmpTrecho,Temp1)=0:NEXT(239$)
Model statements for module: Decide 354
223$      BRANCH,      1:

```

```

                If,NR(ESTA_R ( A PtAtual,1 + (A
Sentido==0)))==0,2752$,Yes:
                Else,2753$,Yes
2752$          ASSIGN:      Decide 354.NumberOut True=Decide 354.NumberOut True +
1:NEXT(5$)
2753$          ASSIGN:      Decide 354.NumberOut False=Decide 354.NumberOut False +
1:NEXT(150$)
                Model statements for module:  Seize 2
5$            QUEUE,      Ocupa o recurso Patio X ou Y.Queue
                SEIZE,      2,Other:
                ESTA_R ( A PtAtual,1 + (A Sentido==0)),1:NEXT(2755$)
2755$          DELAY:      0.0,,VA:NEXT(150$)
                Model statements for module:  Signal 26
166$          SIGNAL:      sorteio,1:NEXT(150$)
                Model statements for module:  Assign 82
33$           ASSIGN:      V ocbl ( A TmpTrecho , 1 )=1:NEXT(29$)
                Model statements for module:  Station 7
111$          STATION,    Estacoes X Set.Station
2756$          ASSIGN:      Attribute 1=MEMIDX(Estacoes X Set.Station,M)
2758$          DELAY:      0.0,,VA:NEXT(159$)
                Model statements for module:  Decide 213
143$          BRANCH,     1:
                If,V ROT((Indice+Indice2), A Rota) > 0,2759$,Yes:
                Else,2760$,Yes
2759$          ASSIGN:      Este patio existe ?.NumberOut True=Este patio existe
?.NumberOut True + 1:NEXT(79$)
2760$          ASSIGN:      Este patio existe ?.NumberOut False=Este patio existe
?.NumberOut False + 1:NEXT(140$)
                Model statements for module:  ReadWrite 34
140$          WRITE,      Log de Erros,
                "%1.8g. ERR10 Erro no trem %1.8g. Numero %1.8g.Tipo
%1.8g. Roteiro %1.8g finalizado sem encontrar a estacao de destino %1.8g. Verificar
orientacao do roteiro. Trem abortado no patio %1.8g.\n":
                TNOW,
                A IDtrem,
                A Ntrem,
                A Tipo,
                A Rota,
                A PtDestino,
                A PtAtual:NEXT(142$)
                Model statements for module:  Signal 18
142$          SIGNAL:      A IDtrem,1:NEXT(141$)
                Model statements for module:  Assign 253
141$          ASSIGN:      A Sentido=X Sentido(Indice2):
                V Ocpt ( A PtAtual , 1+ (A Sentido==0) )=0:NEXT(199$)
                Model statements for module:  Release 14
199$          RELEASE:    Passo3,1:NEXT(233$)
                Model statements for module:  Separate 56
233$          DUPLICATE,  100 - 50:
                1,2763$,50:NEXT(2762$)
2762$          ASSIGN:      Separate 56.NumberOut Orig=Separate 56.NumberOut Orig +
1:NEXT(207$)
2763$          ASSIGN:      Separate 56.NumberOut Dup=Separate 56.NumberOut Dup +
1:NEXT(179$)
                Model statements for module:  Hold 61
207$          QUEUE,      Hold 61.Queue:DETACH
                Model statements for module:  Route 26
179$          ROUTE:      0.0000000000000000,Erro na movimentacao
                Model statements for module:  ReadWrite 36
147$          WRITE,      Log de Mov traduzido,
                "Trem de ID %1.8g de numero %1.8g do horario %1.8g
chegou ao patio %1.8g aos %1.8g min. e saiu aos %1.8g min.\n":
                A IDtrem,
                A Ntrem,
                A Horario,

```

```

        A PtAtual,
        A Tmp,
        TNOW:NEXT(21$)
    Model statements for module: Station 13
176$      STATION,      Erro na movimentacao
2766$     DELAY:        0.0,,VA:NEXT(181$)
    Model statements for module: Decide 269
181$     BRANCH,       1:
                        If,A LinhaOcupada<>0,2767$,Yes:
                        Else,2768$,Yes
2767$     ASSIGN:      Trem removido esta ocupando um patio de atividade
?.NumberOut True=
                        Trem removido esta ocupando um patio de atividade
?.NumberOut True + 1:NEXT(182$)
2768$     ASSIGN:      Trem removido esta ocupando um patio de atividade
?.NumberOut False=
                        Trem removido esta ocupando um patio de atividade
?.NumberOut False + 1:NEXT(201$)
    Model statements for module: Assign 340
182$     ASSIGN:      V StatusPt (A LinhaOcupada, 1)=V StatusPt (A
LinhaOcupada, 1) + 1:
                        A ColPatio=0:NEXT(201$)
    Model statements for module: Dispose 53
201$     ASSIGN:      Dispose 53.NumberOut=Dispose 53.NumberOut + 1
2769$     DISPOSE:     Yes
    Model statements for module: Hold 46
183$     QUEUE,        Hold 46.Queue:DETACH
    Model statements for module: Station 15
307$     STATION,      Restricao por SB
2772$     DELAY:        0.0,,VA:NEXT(313$)
    Model statements for module: Assign 691
313$     ASSIGN:      Entity.Station=A EstAtual:NEXT(38$)
    Model statements for module: Assign 744
367$     ASSIGN:      Indice=tria(9,10,40):NEXT(368$)
    Model statements for module: Assign 745
368$     ASSIGN:      Indice=tria(9,10,40):NEXT(369$)
    Model statements for module: Dispose 87
369$     ASSIGN:      Dispose 87.NumberOut=Dispose 87.NumberOut + 1
2773$     DISPOSE:     Yes
    Model statements for module: Create 3
2774$     CREATE,      1,HoursToBaseTime(0.0),Entity
1:HoursToBaseTime(EXPO(1)),1:NEXT(2775$)
2775$     ASSIGN:      Cria as rotas na variavel V ROT.NumberOut=Cria as rotas
na variavel V ROT.NumberOut + 1:NEXT(789$)
    Model statements for module: Assign 166
789$     ASSIGN:      Picture=Picture.Marrom:
                        Picture=Picture.Cinza:
                        Picture=Picture.AzulClaro:
                        Picture=Picture.Laranja:
                        Picture=Picture.Cinzaescuro:NEXT(788$)
    Model statements for module: ReadWrite 15
788$     WRITE,        Log de Erros,"LOG DE ERROS INICIALIZADO\n":NEXT(773$)
    Model statements for module: Assign 119
773$     ASSIGN:      Indicel=0:NEXT(768$)
    Model statements for module: Assign 115
768$     ASSIGN:      IndLinha=IndLinha+1:
                        IndColuna=1:
                        passo=0:
                        Temp1=0:
                        Temp2=0:NEXT(793$)
    Model statements for module: Decide 486
793$     BRANCH,       1:
                        If,IndLinha > 110,2778$,Yes:
                        Else,2779$,Yes
2778$     ASSIGN:      Acabaram os roteiros do usuario ?.NumberOut True=

```

```

Acabaram os roteiros do usuario ?.NumberOut True +
1:NEXT(776$)
2779$      ASSIGN:      Acabaram os roteiros do usuario ?.NumberOut False=
Acabaram os roteiros do usuario ?.NumberOut False +
1:NEXT(767$)
  Model statements for module:  Assign 121
776$      ASSIGN:      Indice=0:NEXT(777$)
  Model statements for module:  Assign 122
777$      ASSIGN:      Indice=indice+1:NEXT(778$)
  Model statements for module:  ReadWrite 13
778$      WRITE,      Verifica," %1.8g-%1.8g %1.8g-%1.8g %1.8g-%1.8g %1.8g-
%1.8g %1.8g-%1.8g\n":
V ROT(Indice,1),
V DROT(Indice,1),
V ROT(Indice,2),
V DROT(Indice,2),
V ROT(Indice,3),
V DROT(Indice,3),
V ROT(Indice,4),
V DROT(Indice,4),
V ROT(Indice,53),
V DROT(Indice,53):NEXT(779$)
  Model statements for module:  Decide 97
779$      BRANCH,      1:
If,Indice>150,2780$,Yes:
Else,2781$,Yes
2780$      ASSIGN:      Decide 97.NumberOut True=Decide 97.NumberOut True +
1:NEXT(792$)
2781$      ASSIGN:      Decide 97.NumberOut False=Decide 97.NumberOut False +
1:NEXT(777$)
  Model statements for module:  Decide 181
792$      BRANCH,      1:
If,A PtAtual == 1,2782$,Yes:
Else,2783$,Yes
2782$      ASSIGN:      Houve algum erro de roteiro ?.NumberOut True=Houve algum
erro de roteiro ?.NumberOut True + 1
:NEXT(790$)
2783$      ASSIGN:      Houve algum erro de roteiro ?.NumberOut False=Houve
algum erro de roteiro ?.NumberOut False + 1
:NEXT(780$)
790$      VBA:          1,vba:NEXT(780$)
  Model statements for module:  Dispose 19
780$      ASSIGN:      Dispose 19.NumberOut=Dispose 19.NumberOut + 1
2784$      DISPOSE:      Yes
  Model statements for module:  Decide 94
767$      BRANCH,      1:
If,V ROT Usuario(IndLinha,IndColuna)==0,2785$,Yes:
Else,2786$,Yes
2785$      ASSIGN:      Roteiro zerado ?.NumberOut True=Roteiro zerado
?.NumberOut True + 1:NEXT(773$)
2786$      ASSIGN:      Roteiro zerado ?.NumberOut False=Roteiro zerado
?.NumberOut False + 1:NEXT(781$)
  Model statements for module:  Decide 98
781$      BRANCH,      1:
If,V DROT(1,IndLinha)==0,2787$,Yes:
Else,2788$,Yes
2787$      ASSIGN:      Qual o sentido do primeiro trecho ?.NumberOut True=
Qual o sentido do primeiro trecho ?.NumberOut True +
1:NEXT(782$)
2788$      ASSIGN:      Qual o sentido do primeiro trecho ?.NumberOut False=
Qual o sentido do primeiro trecho ?.NumberOut False +
1:NEXT(771$)
  Model statements for module:  Assign 123
782$      ASSIGN:      Temp1=0:
Fim=V INDTRECHO(V ROT Usuario(IndLinha,IndColuna),1):

```



```

                Inicio=V INDTRECHO(V ROT
Usuario(IndLinha,IndColuna),2)+Temp2:
                passo=(-1):
                Indicel=Indicel:
                A Peso=1:NEXT(772$)
    Model statements for module: Assign 118
772$          ASSIGN:      Indicel=Indicel+(A Peso==0):
                A Peso=0:
                Indicel=Indicel + (Indicel==0):NEXT(774$)
    Model statements for module: Assign 120
774$          ASSIGN:      V ROT(indicel,IndLinha)=Inicio:
                V DROT(indicel,IndLinha)=(passo==1):
                Inicio=Inicio + passo:NEXT(775$)
    Model statements for module: Decide 96
775$          BRANCH,      1:

If, (Inicio>fim.and.passo==1).or. (Inicio<fim.and.passo==(-1)),2789$,Yes:
                Else,2790$,Yes
2789$          ASSIGN:      Terminou o trecho ?.NumberOut True=Terminou o trecho
?.NumberOut True + 1:NEXT(769$)
2790$          ASSIGN:      Terminou o trecho ?.NumberOut False=Terminou o trecho
?.NumberOut False + 1:NEXT(772$)
    Model statements for module: Assign 116
769$          ASSIGN:      IndColuna=IndColuna+1:NEXT(770$)
    Model statements for module: Decide 95
770$          BRANCH,      1:
                If,V ROT Usuario ( IndLinha , IndColuna )==0,2791$,Yes:
                Else,2792$,Yes
2791$          ASSIGN:      Acabaram os trechos deste roteiro ?.NumberOut True=
                Acabaram os trechos deste roteiro ?.NumberOut True +
1:NEXT(773$)
2792$          ASSIGN:      Acabaram os trechos deste roteiro ?.NumberOut False=
                Acabaram os trechos deste roteiro ?.NumberOut False +
1:NEXT(784$)
    Model statements for module: Assign 124
784$          ASSIGN:      passo=V INDTRECHO (V ROT Usuario ( IndLinha , IndColuna
),1):NEXT(783$)
    Model statements for module: Decide 99
783$          BRANCH,      1:
                If,V TRECHO ( V ROT (Indicel , IndLinha) , 2) <> V
TRECHO(passo,2),2793$,Yes:
                Else,2794$,Yes
2793$          ASSIGN:      Primeiro patio do proximo segmento diferente deste
?.NumberOut True=
                Primeiro patio do proximo segmento diferente deste
?.NumberOut True + 1:NEXT(785$)
2794$          ASSIGN:      Primeiro patio do proximo segmento diferente deste
?.NumberOut False=
                Primeiro patio do proximo segmento diferente deste
?.NumberOut False + 1:NEXT(771$)
    Model statements for module: Assign 125
785$          ASSIGN:      Temp1=passo:
                passo=V INDTRECHO (V ROT Usuario ( IndLinha , IndColuna
),2):NEXT(786$)
    Model statements for module: Decide 100
786$          BRANCH,      1:
                If,V TRECHO ( V ROT (Indicel , IndLinha) , 2) <> V
TRECHO(passo,2),2795$,Yes:
                Else,2796$,Yes
2795$          ASSIGN:      Estes patios nao se conectam pois as duas pontas sao
diferentes ?.NumberOut True=
                Estes patios nao se conectam pois as duas pontas sao
diferentes ?.NumberOut True + 1:NEXT(791$)
2796$          ASSIGN:      Estes patios nao se conectam pois as duas pontas sao
diferentes ?.NumberOut False=

```

```

Estes patios nao se conectam pois as duas pontas sao
diferentes ?.NumberOut False + 1:NEXT(782$)
  Model statements for module: Assign 231
791$   ASSIGN:      A PtAtual=1:NEXT(787$)
  Model statements for module: ReadWrite 14
787$   WRITE,       Log de Erros,
                          "%1.8g. ERR1 Erro na Rota %1.8g. Trecho %1.8g nao tem
ligacao com o %1.8g no sentido indicado. Roteiro abortado.\n":
                          TNOW,
                          IndLinha,
                          V TRECHO ( V ROT (Indice1 , IndLinha) , 1),
                          V TRECHO(passo,1):NEXT(773$)
  Model statements for module: Assign 117
771$   ASSIGN:      Inicio=V INDTRECHO(V ROT
Usuario(IndLinha,IndColuna),1)+Templ:
                          Fim=V INDTRECHO(V ROT Usuario(IndLinha,IndColuna),2):
                          passo=1:
                          Indice1=Indice1:
                          A Peso=1:NEXT(772$)
  Model statements for module: Create 4
2797$   CREATE,     1,MinutesToBaseTime(0.01),Entity
1:MinutesToBaseTime(EXPO(19)),1:NEXT(2798$)
2798$   ASSIGN:      Cria trem grade e expresso matriz.NumberOut=Cria trem
grade e expresso matriz.NumberOut + 1
                          :NEXT(827$)
  Model statements for module: Decide 260
827$   BRANCH,     1:
                          If,V Grade( 1, 2) > 0,2801$,Yes:
                          Else,2802$,Yes
2801$   ASSIGN:      Grade nao vazia ?.NumberOut True=Grade nao vazia
?.NumberOut True + 1:NEXT(803$)
2802$   ASSIGN:      Grade nao vazia ?.NumberOut False=Grade nao vazia
?.NumberOut False + 1:NEXT(828$)
  Model statements for module: Assign 139
803$   ASSIGN:      Indice=Indice + 1:
                          A Ntrem=Indice:
                          A PtOrigem=V Grade( Indice, 2):
                          A PtDestino=V Grade( Indice, 3):
                          A Rota=V Grade( Indice, 4):
                          A Tam=V Grade( Indice, 1):
                          A NLocos=0:
                          Entity.Picture=Desenhos ( V Grade( Indice, 6) ):
                          A Horario=V Grade( Indice , 5):
                          A Tipo=4:
                          A Cliente=10:
                          A Prioridade=5:NEXT(804$)
  Model statements for module: Separate 10
804$   DUPLICATE,  100 - 50:
                          1,2805$,50:NEXT(2804$)
2804$   ASSIGN:      Cria matriz deste trem grade.NumberOut Orig=Cria matriz
deste trem grade.NumberOut Orig + 1
                          :NEXT(807$)
2805$   ASSIGN:      Cria matriz deste trem grade.NumberOut Dup=Cria matriz
deste trem grade.NumberOut Dup + 1
                          :NEXT(805$)
  Model statements for module: Delay 19
807$   DELAY:      A Horario,,Other:NEXT(808$)
  Model statements for module: Separate 11
808$   DUPLICATE,  100 - 50:
                          1,2808$,50:NEXT(2807$)
2807$   ASSIGN:      Cria trem que vai entrar na malha.NumberOut Orig=
                          Cria trem que vai entrar na malha.NumberOut Orig +
1:NEXT(857$)
2808$   ASSIGN:      Cria trem que vai entrar na malha.NumberOut Dup=Cria
trem que vai entrar na malha.NumberOut Dup + 1

```

```

: NEXT(809$)
Model statements for module: Seize 10
857$    QUEUE,      Reserva ID para trem CG.Queue
        SEIZE,      2,Other:
                ResourceID,1:NEXT(2810$)
2810$    DELAY:      0.0,,VA:NEXT(859$)
Model statements for module: Assign 683
859$    ASSIGN:      ContaID=ContaID + 1:
                A IDtrem=ContaID:NEXT(860$)
Model statements for module: Release 25
860$    RELEASE:     ResourceID,1:NEXT(794$)
Model statements for module: Assign 126
794$    ASSIGN:      A IndiRota=1:
                A PtAtual=A PtOrigem:
                A Estado=1:
                A IndiAtiv=1:
                A Peso=E PesosT(A Ntrem, 2+(A Tipo==5)):
                Temp1=100 - (V Supress( A NTrem, (1*(A Tipo==4)+2*(A
Tipo==5) ) ) *100:NEXT(830$)
Model statements for module: Decide 306
830$    BRANCH,      1:
                With, (Temp1)/100,2811$,Yes:
                Else,2812$,Yes
2811$    ASSIGN:      Supressao do trem grade ou expresso.NumberOut True=
                Supressao do trem grade ou expresso.NumberOut True +
1:NEXT(865$)
2812$    ASSIGN:      Supressao do trem grade ou expresso.NumberOut False=
                Supressao do trem grade ou expresso.NumberOut False +
1:NEXT(831$)
Model statements for module: Record 35
865$    COUNT:      Conta entrando,1:NEXT(800$)
Model statements for module: Assign 128
800$    ASSIGN:      A Sentido=V DROT(A IndiRota, A Rota):
                A Trecho=V ROT(A IndiRota, A Rota):
                Sh7=A trecho:NEXT(796$)
Model statements for module: Decide 101
796$    BRANCH,      1:
                If,V Trecho ( A Trecho , 2 ) == A PtOrigem,2813$,Yes:
                Else,2814$,Yes
2813$    ASSIGN:      Patio de origem do trem esta neste trecho ?.NumberOut
True=
                Patio de origem do trem esta neste trecho ?.NumberOut
True + 1:NEXT(872$)
2814$    ASSIGN:      Patio de origem do trem esta neste trecho ?.NumberOut
False=
                Patio de origem do trem esta neste trecho ?.NumberOut
False + 1:NEXT(795$)
Model statements for module: Assign 737
872$    ASSIGN:      Temp2=tnow:NEXT(870$)
Model statements for module: Decide 675
870$    BRANCH,      1:
                If,V Ocpt ( A PtOrigem , 1+ (A Sentido==0) ) ==
0,2815$,Yes:
                Else,2816$,Yes
2815$    ASSIGN:      Patio no sentido correto esta livre ?.NumberOut True=
                Patio no sentido correto esta livre ?.NumberOut True +
1:NEXT(825$)
2816$    ASSIGN:      Patio no sentido correto esta livre ?.NumberOut False=
                Patio no sentido correto esta livre ?.NumberOut False +
1:NEXT(873$)
Model statements for module: Decide 676
873$    BRANCH,      1:
                If, (TNOW - Temp2) > 190,2817$,Yes:
                Else,2818$,Yes

```

```

2817$      ASSIGN:      Tentando ha mais de 2 horas ?.NumberOut True=Tentando ha
mais de 2 horas ?.NumberOut True + 1
                        :NEXT(874$)
2818$      ASSIGN:      Tentando ha mais de 2 horas ?.NumberOut False=Tentando
ha mais de 2 horas ?.NumberOut False + 1
                        :NEXT(871$)
      Model statements for module:  ReadWrite 194
874$      WRITE,      Log de Erros,
                        "%1.8g. TREM CANCELADO ERR19. Trem tipo  %1.8g numero
%1.8g. Patio de origem %1.8g. \n":
                        TNOW,
                        A Tipo,
                        A Ntrem,
                        A PtOrigem:NEXT(875$)
      Model statements for module:  Dispose 86
875$      ASSIGN:      Dispose 86.NumberOut=Dispose 86.NumberOut + 1
2819$      DISPOSE:    Yes
      Model statements for module:  Delay 57
871$      DELAY:      SecondsToBaseTime(norm(90,10)),,Other:NEXT(870$)
      Model statements for module:  Assign 313
825$      ASSIGN:      A PtAtual=V Trecho ( A Trecho , 2 ):
                        MarcaCiclo=tnow:NEXT(866$)
      Model statements for module:  Record 36
866$      COUNT:      Conta saindo,1:NEXT(826$)
      Model statements for module:  Assign 314
826$      ASSIGN:      A PtAtual=V Trecho ( A Trecho , 2 ):
                        A Tmp=8123:
                        MarcaCT=TNOW:NEXT(849$)
      Model statements for module:  Decide 485
849$      BRANCH,      1:
                        If,A SouHelper == 1,2820$,Yes:
                        Else,2821$,Yes
2820$      ASSIGN:      Trem helper nao pede locomotivas nem grava log.NumberOut
True=
                        Trem helper nao pede locomotivas nem grava log.NumberOut
True + 1:NEXT(801$)
2821$      ASSIGN:      Trem helper nao pede locomotivas nem grava log.NumberOut
False=
                        Trem helper nao pede locomotivas nem grava log.NumberOut
False + 1:NEXT(837$)
      Model statements for module:  Station 6
801$      STATION,      Station Origem de todos
2824$      DELAY:      0.0,,VA:NEXT(835$)
      Model statements for module:  Decide 335
835$      BRANCH,      1:
                        If,Tnow > 1530 .AND. A PtOrigem==81,2825$,Yes:
                        Else,2826$,Yes
2825$      ASSIGN:      Decide 335.NumberOut True=Decide 335.NumberOut True +
1:NEXT(832$)
2826$      ASSIGN:      Decide 335.NumberOut False=Decide 335.NumberOut False +
1:NEXT(802$)
      Model statements for module:  Separate 50
832$      DUPLICATE,    100 - 50:
                        1,2829$,50:NEXT(2828$)
2828$      ASSIGN:      Separate 50.NumberOut Orig=Separate 50.NumberOut Orig +
1:NEXT(833$)
2829$      ASSIGN:      Separate 50.NumberOut Dup=Separate 50.NumberOut Dup +
1:NEXT(802$)
      Model statements for module:  Hold 62
833$      QUEUE,      Hold 62.Queue:DETACH
      Model statements for module:  Route 8
802$      ROUTE:      0.000000000000000,ESTA ( A PtAtual,1 + ( A Sentido == 0)
)
      Model statements for module:  Decide 430
837$      BRANCH,      1:

```

```

                If,TNOW > (V Warmup * 1440),2830$,Yes:
                Else,2831$,Yes
2830$          ASSIGN:      Warmup do log de eventos.NumberOut True=Warmup do log de
eventos.NumberOut True + 1:NEXT(867$)
2831$          ASSIGN:      Warmup do log de eventos.NumberOut False=Warmup do log
de eventos.NumberOut False + 1:NEXT(836$)
                Model statements for module:  Decide 626
867$           BRANCH,      1:
                If,V Gravar em texto == 1,2832$,Yes:
                Else,2833$,Yes
2832$          ASSIGN:      Decide 626.NumberOut True=Decide 626.NumberOut True +
1:NEXT(868$)
2833$          ASSIGN:      Decide 626.NumberOut False=Decide 626.NumberOut False +
1:NEXT(824$)
                Especificação do arquivo de LOG de eventos:
                1 - Dia do evento
                2- Identificador único do trem (A IDTrem)
                3 - Número do trem (A NTrem)
                4 - Tipo do trem (A Tipo)
                5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
                6 - Pátio (A PtAtual)
                7 - Momento da saída do trem (TNOW)
                8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
                Model statements for module:  ReadWrite 160
868$           WRITE,       Log de Eventos:
                Dia,
                A IDtrem,
                A Ntrem,
                A Tipo,
                marca,
                A PtAtual,
                tnow,
                6,
                A Peso,
                A Dist,
                E FrotasT (A Ntrem, 1),
                E FrotasT (A Ntrem, 2),
                E FrotasT (A Ntrem, 3),
                E FrotasT (A Ntrem, 4),
                E FrotasT (A Ntrem, 5),
                A Estado,
                A Produto,
                A Cliente:NEXT(836$)
                Model statements for module:  Assign 474
836$           ASSIGN:      V TUDia=V TUDia + A Peso:NEXT(822$)
                Model statements for module:  Decide 215
822$           BRANCH,      1:
                If,A Tipo == 4,2834$,Yes:
                Else,2835$,Yes
2834$          ASSIGN:      Trem grade ?.NumberOut True=Trem grade ?.NumberOut True
+ 1:NEXT(821$)
2835$          ASSIGN:      Trem grade ?.NumberOut False=Trem grade ?.NumberOut
False + 1:NEXT(823$)
                Model statements for module:  Assign 262
821$           ASSIGN:      V LocosNow(1)=V LocosNow(1) + E FrotasG (A Ntrem, 1):
                V LocosNow(2)=V LocosNow(2) + E FrotasG (A Ntrem, 2):
                V LocosNow(3)=V LocosNow(3) + E FrotasG (A Ntrem, 3):
                V LocosNow(4)=V LocosNow(4) + E FrotasG (A Ntrem,4):
                V LocosNow(5)=V LocosNow(5) + E FrotasG (A Ntrem,
5):NEXT(851$)
                Model statements for module:  Assign 589
851$           ASSIGN:      V Trens na Malha(4)=V Trens na Malha(4) + 1:NEXT(801$)
                Model statements for module:  Assign 263

```

```

823$      ASSIGN:      V LocosNow(1)=V LocosNow(1) + E FrotasE (A Ntrem, 1):
                          V LocosNow(2)=V LocosNow(2) + E FrotasE (A Ntrem, 2):
                          V LocosNow(3)=V LocosNow(3) + E FrotasE (A Ntrem, 3):
                          V LocosNow(4)=V LocosNow(4) + E FrotasE (A Ntrem, 4):
                          V LocosNow(5)=V LocosNow(5) + E FrotasE (A Ntrem,
5):NEXT(852$)
      Model statements for module: Assign 590
852$      ASSIGN:      V Trens na Malha(5)=V Trens na Malha(5) + 1:NEXT(801$)
      Especificação do arquivo de LOG de eventos:
      1 - Dia do evento
      2- Identificador único do trem (A IDTrem)
      3 - Número do trem (A NTrem)
      4 - Tipo do trem (A Tipo)
      5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
      6 - Pátio (A PtAtual)
      7 - Momento da saída do trem (TNOW)
      8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
      Model statements for module: ReadWrite 49
824$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeEvento):
                          Dia,
                          A IDtrem,
                          A Ntrem,
                          A Tipo,
                          marca,
                          A PtAtual,
                          tnow,
                          6,
                          A Peso,
                          A Dist,
                          E FrotasT (A Ntrem, 1),
                          E FrotasT (A Ntrem, 2),
                          E FrotasT (A Ntrem, 3),
                          E FrotasT (A Ntrem, 4),
                          E FrotasT (A Ntrem, 5),
                          A Estado,
                          A Produto,
                          A Cliente:NEXT(836$)
      Model statements for module: Assign 127
795$      ASSIGN:      A IndiRota=A IndiRota + 1:NEXT(797$)
      Model statements for module: Decide 102
797$      BRANCH,      1:
                          If,V ROT(A IndiRota, A Rota) == 0,2836$,Yes:
                          Else,2837$,Yes
2836$      ASSIGN:      Acabaram os trechos do roteiro sem achar a estacao
?.NumberOut True=
                          Acabaram os trechos do roteiro sem achar a estacao
?.NumberOut True + 1:NEXT(798$)
2837$      ASSIGN:      Acabaram os trechos do roteiro sem achar a estacao
?.NumberOut False=
                          Acabaram os trechos do roteiro sem achar a estacao
?.NumberOut False + 1:NEXT(800$)
      Model statements for module: ReadWrite 16
798$      WRITE,      Log de Erros,
                          "%1.8g. ERR2 Erro na Rota %1.8g. Trem tipo %1.8g numero
%1.8g. Nao encontrado patio numero %1.8g de origem do trem. Trem abortado.\n":
                          TNOW,
                          A Rota,
                          A Tipo,
                          A Ntrem,
                          A PtOrigem:NEXT(799$)
      Model statements for module: Dispose 20
799$      ASSIGN:      Dispose 20.NumberOut=Dispose 20.NumberOut + 1
2838$      DISPOSE:      Yes

```

```

Model statements for module: Dispose 54
831$      ASSIGN:      Trem suprimido.NumberOut=Trem suprimido.NumberOut + 1
2839$     DISPOSE:     Yes
Model statements for module: Delay 20
809$     DELAY:       1440.0000000000000000,,Other:NEXT(808$)
Model statements for module: Decide 114
805$     BRANCH,      1:
                        If,Indice == 50,828$,Yes:
                        If,V Grade( Indice + 1, 1) == 0,828$,Yes:
                        Else,803$,Yes
Model statements for module: Decide 261
828$     BRANCH,      1:
                        If,V Expresso( 1, 2) > 0.AND.1,2842$,Yes:
                        Else,2843$,Yes
2842$     ASSIGN:     Expressos nao vazios ?.NumberOut True=Expressos nao
vazios ?.NumberOut True + 1:NEXT(813$)
2843$     ASSIGN:     Expressos nao vazios ?.NumberOut False=Expressos nao
vazios ?.NumberOut False + 1:NEXT(829$)
Model statements for module: Assign 164
813$     ASSIGN:     A Tipo=5:
                        Indice=1:
                        A Prioridade=4:NEXT(810$)
Model statements for module: Assign 163
810$     ASSIGN:     Indice=Indice + 1:
                        A Ntrem=Indice:
                        A PtOrigem=V Expresso( Indice, 2):
                        A PtDestino=V Expresso( Indice, 3):
                        A Rota=V Expresso( Indice, 4):
                        A Tam=V Expresso( Indice, 1):
                        A NLocos=0:
                        Entity.Picture=Desenhos ( V Grade( Indice, 6) ):
                        A Horario=V Expresso( Indice , 5):
                        A Cliente=11:NEXT(811$)
Model statements for module: Separate 17
811$     DUPLICATE,   100 - 50:
                        1,2846$,50:NEXT(2845$)
2845$     ASSIGN:     Cria matriz deste trem expresso.NumberOut Orig=Cria
matriz deste trem expresso.NumberOut Orig + 1
                        :NEXT(807$)
2846$     ASSIGN:     Cria matriz deste trem expresso.NumberOut Dup=Cria
matriz deste trem expresso.NumberOut Dup + 1
                        :NEXT(812$)
Model statements for module: Decide 153
812$     BRANCH,      1:
                        If,Indice == 50,806$,Yes:
                        If,V Expresso( Indice + 1, 1) == 0,806$,Yes:
                        Else,810$,Yes
Model statements for module: Dispose 23
806$     ASSIGN:     Descarta trem matriz principal.NumberOut=Descarta trem
matriz principal.NumberOut + 1
2849$     DISPOSE:     Yes
Model statements for module: Dispose 51
829$     ASSIGN:     Dispose 51.NumberOut=Dispose 51.NumberOut + 1
2850$     DISPOSE:     Yes
Model statements for module: Create 7
2851$     CREATE,      1,MinutesToBaseTime(0.01),Entity
1:MinutesToBaseTime(EXPO(19)),1:NEXT(2852$)
2852$     ASSIGN:     Cria trem unitario matriz.NumberOut=Cria trem unitario
matriz.NumberOut + 1:NEXT(814$)
Model statements for module: Assign 165
814$     ASSIGN:     Indice=Indice + 1:
                        A Ntrem=Indice:
                        A PtOrigem=E Tabelas( Indice, 1):
                        A PtDestino=E Tabelas( Indice, 1):
                        A Rota=0:

```

```

A NLocos=0:
Entity.Picture=Picture.Cinzaescuro:
A Tipo=E Tabelas ( Indice , 4):
A Produto=A Tipo:
A PtZero=A PtOrigem:NEXT(815$)
Model statements for module: Separate 18
815$      DUPLICATE,      100 - 50:
          E Tabelas( Indice , 2),2857$,50:NEXT(2856$)
2856$      ASSIGN:      Cria matriz deste trem unitario.NumberOut Orig=Cria
matriz deste trem unitario.NumberOut Orig + 1
          :NEXT(816$)
2857$      ASSIGN:      Cria matriz deste trem unitario.NumberOut Dup=Cria
matriz deste trem unitario.NumberOut Dup + 1
          :NEXT(817$)
Model statements for module: Decide 154
816$      BRANCH,      1:
          If,Indice == 30,820$,Yes:
          If,E Tabelas( Indice + 1, 1) == 0,820$,Yes:
          Else,814$,Yes
Model statements for module: Dispose 30
820$      ASSIGN:      Dispose 30.NumberOut=Dispose 30.NumberOut + 1
2860$      DISPOSE:      Yes
Model statements for module: Assign 167
817$      ASSIGN:      A Tam=E Tabelas (Indice, 3) * V CVT(Indice):
          A Tipo=2 * (A Tipo==1.OR.A Tipo==2) + 3 * (A Tipo > 2):
          A Peso=E Tabelas (Indice, 3) * E PesosT(indice,1):
          A Prioridade=2:NEXT(839$)
Model statements for module: Assign 518
839$      ASSIGN:      A Grupo=E Tabelas (A NTrem, 5):NEXT(853$)
Model statements for module: Seize 9
853$      QUEUE,      Reserva ID para trem tabela.Queue
          SEIZE,      2,Other:
          ResourceID,1:NEXT(2862$)
2862$      DELAY:      0.0,,VA:NEXT(855$)
Model statements for module: Assign 682
855$      ASSIGN:      ContaID=ContaID + 1:
          A IDtrem=ContaID:NEXT(856$)
Model statements for module: Release 24
856$      RELEASE:      ResourceID,1:NEXT(818$)
Model statements for module: Assign 168
818$      ASSIGN:      ContaID=ContaID + 1:
          A IDtrem=ContaID:
          A IndiRota=1:
          A PtAtual=A PtOrigem:
          Entity.Station=Station Origem de todos:
          A Tmp=8123:
          A Entrada=A Produto == 3:NEXT(838$)
Model statements for module: Assign 484
838$      ASSIGN:      V LocosNow(1)=V LocosNow(1) + E FrotasT (A Ntrem, 1):
          V LocosNow(2)=V LocosNow(2) + E FrotasT (A Ntrem, 2):
          V LocosNow(3)=V LocosNow(3) + E FrotasT (A Ntrem, 3):
          V LocosNow(4)=V LocosNow(4) + E FrotasT (A Ntrem,4):
          V LocosNow(5)=V LocosNow(5) + E FrotasT (A Ntrem,
5):NEXT(819$)
O envio é realizado com um tempo de espera para cada
entidade, de modo que não cheguem todas juntas,
configurando uma situação irreal.
Model statements for module: Route 9
819$      ROUTE:      UNIF(0,100),Direcionamento
Model statements for module: Station 14
840$      STATION,      Retorno de helpers
2865$      DELAY:      0.0,,VA:NEXT(841$)
Model statements for module: Assign 561
841$      ASSIGN:      A SouHelper=1:NEXT(842$)
Model statements for module: Decide 483

```



```

842$          BRANCH,          1:
                                If,A PtAtual == 21,2866$,Yes:
                                Else,2867$,Yes
2866$          ASSIGN:         Helper chegando em FOJ ?.NumberOut True=Helper chegando
em FOJ ?.NumberOut True + 1:NEXT(843$)
2867$          ASSIGN:         Helper chegando em FOJ ?.NumberOut False=Helper chegando
em FOJ ?.NumberOut False + 1:NEXT(848$)
                                Model statements for module: Assign 562
843$          ASSIGN:         V ComboioH=V ComboioH + A NLocos:NEXT(844$)
                                Model statements for module: Decide 484
844$          BRANCH,          1:
                                If,V ComboioH >= E Helpers(4,1),2868$,Yes:
                                Else,2869$,Yes
2868$          ASSIGN:         Montou o comboio ?.NumberOut True=Montou o comboio
?.NumberOut True + 1:NEXT(846$)
2869$          ASSIGN:         Montou o comboio ?.NumberOut False=Montou o comboio
?.NumberOut False + 1:NEXT(845$)
                                Model statements for module: Assign 563
846$          ASSIGN:         A NLocos=V ComboioH:
                                V ComboioH=0:NEXT(848$)
                                Model statements for module: Assign 565
848$          ASSIGN:         A Ntrem=9000:
                                A PtOrigem=A PtAtual:
                                A PtDestino=A RetornoH:
                                A Tam=10:
                                Picture=Picture.helper:
                                A Horario=0:
                                A Tipo=4:
                                A Cliente=10:NEXT(850$)
                                Model statements for module: Assign 566
850$          ASSIGN:         A Rota=
                                101*(A RetornoH==131) + 102*(A RetornoH==143) + 103*(A
RetornoH==144) + 104*(A RetornoH==1) + 105*(A RetornoH==7) + 106*(A RetornoH==30) +
107*(A RetornoH==37) + 108*(A RetornoH==75)
                                :NEXT(861$)
                                Model statements for module: Seize 11
861$          QUEUE,          Reserva ID para trem helper.Queue
                                SEIZE,          2,Other:
                                ResourceID,1:NEXT(2871$)
2871$          DELAY:         0.0,,VA:NEXT(863$)
                                Model statements for module: Assign 684
863$          ASSIGN:         ContaID=ContaID + 1:
                                A IDtrem=ContaID:NEXT(864$)
                                Model statements for module: Release 26
864$          RELEASE:        ResourceID,1:NEXT(847$)
                                Model statements for module: Assign 564
847$          ASSIGN:         A IndiRota=1:
                                A Estado=1:
                                A IndiAtiv=1:
                                A Peso=0:NEXT(865$)
                                Model statements for module: Dispose 65
845$          ASSIGN:         Dispose 65.NumberOut=Dispose 65.NumberOut + 1
2872$          DISPOSE:        Yes
                                Model statements for module: Dispose 88
869$          ASSIGN:         Dispose 88.NumberOut=Dispose 88.NumberOut + 1
2873$          DISPOSE:        Yes
                                Model statements for module: Station 9
884$          STATION,        Direcionamento
2876$          DELAY:         0.0,,VA:NEXT(953$)
                                Model statements for module: Decide 450
953$          BRANCH,          1:
                                If,TNOW > (V Warmup * 1440),2877$,Yes:
                                Else,2878$,Yes
2877$          ASSIGN:         Warmup do log de ciclos etapa 3.NumberOut True=Warmup do
log de ciclos etapa 3.NumberOut True + 1

```

```

: NEXT(964$)
2878$      ASSIGN:      Warmup do log de ciclos etapa 3.NumberOut False=Warmup
do log de ciclos etapa 3.NumberOut False + 1
: NEXT(885$)
      Model statements for module:  Decide 635
964$      BRANCH,      1:
                        If,V Gravar em texto == 1,2879$,Yes:
                        Else,2880$,Yes
2879$      ASSIGN:      Decide 635.NumberOut True=Decide 635.NumberOut True +
1:NEXT(966$)
2880$      ASSIGN:      Decide 635.NumberOut False=Decide 635.NumberOut False +
1:NEXT(942$)
      Especificação do arquivo de LOG de eventos:
      1 - Dia do evento
      2- Identificador único do trem (A IDTrem)
      3 - Número do trem (A NTrem)
      4 - Tipo do trem (A Tipo)
      5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
      6 - Pátio (A PtAtual)
      7 - Momento da saída do trem (TNOW)
      8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outtras)
      Model statements for module:  ReadWrite 161
966$      WRITE,      Log de Ciclos:
                        Dia,
                        A IDtrem,
                        A Ntrem,
                        A Tipo,
                        A PtOrigem,
                        A PtAtual,
                        tnow - MarcaCiclo,
                        A Produto,
                        A Estado,
                        A Cliente,
                        A PtZero,
                        3,
                        A Tcarga:NEXT(885$)
      Model statements for module:  Decide 158
885$      BRANCH,      1:
                        If,A Tmp == 8123,2881$,Yes:
                        Else,2882$,Yes
2881$      ASSIGN:      Trem foi criado agora ?.NumberOut True=Trem foi criado
agora ?.NumberOut True + 1:NEXT(886$)
2882$      ASSIGN:      Trem foi criado agora ?.NumberOut False=Trem foi criado
agora ?.NumberOut False + 1:NEXT(890$)
      Model statements for module:  Seize 4
886$      QUEUE,      Filas de entrada.Queue(A Produto)
      SEIZE,      2,Other:
                        Recursos Bloqueadores(A Produto),1:NEXT(2884$)
2884$      DELAY:      0.0,,VA:NEXT(888$)
      Model statements for module:  Delay 23
888$      DELAY:      300.00000000000000,,Other:NEXT(959$)
      Model statements for module:  Assign 592
959$      ASSIGN:      V Trens na Malha(2)=V Trens na Malha(2) + 1:NEXT(889$)
      Model statements for module:  Release 12
889$      RELEASE:      Recursos Bloqueadores(A Produto),1:NEXT(890$)
151 = FPK
7 = FJC
148 = FDM
      Model statements for module:  Decide 160
890$      BRANCH,      1:
                        If,A PtZero == 151 .OR. A PtZero == 7 .OR. A PtZero ==
148,2885$,Yes:
                        Else,2886$,Yes

```

```

2885$      ASSIGN:      Separa por ponto de origem.NumberOut True=Separa por
ponto de origem.NumberOut True + 1:NEXT(955$)
2886$      ASSIGN:      Separa por ponto de origem.NumberOut False=Separa por
ponto de origem.NumberOut False + 1
                        :NEXT(960$)
      Model statements for module: Assign 519
955$      ASSIGN:      Shift=(A Grupo * 12) - 12:
                        A Tmp=1*(A PtZero == 151) + 2*(A PtZero == 7) + 3*(A
PtZero == 148):
                        aTemp3=0:
                        Indice=1:NEXT(892$)

Busca:
MAXIMIZAR a expressão
MAX((1/((E APCg(J,2)==0)+E APCg(J,2))) +
(1 / ((E TempoCg(A Tmp,J-Shift)==0)+
E TempoCg(A Tmp,J-Shift))) +
(1 / ( (V DA(J)==0) + (V DA(J)/( (V DO(J)==0)+V DO(J)) ) ) ) *
(V DA(J) < V DO(J)) -
(10000 * (E TempoCg(A Tmp,J-Shift)==0))
- 100 * (V TD(J-Shift)))
      Model statements for module: Search 1
892$      FINDJ,      1+Shift,12+Shift:
                        MAX((1/((E ExtraPcg(J,2)==0) + E ExtraPcg(J,2))) + (1 /
((E TempoCg(A Tmp,J-Shift)==0)+E TempoCg(A Tmp,J-Shift))) + (1 / ( (V DA(J)==0) + (V
DA(J)/( (V DO(J)==0)+V DO(J)) ) ) ) * (V DA(J) < V DO(J)) - (900000 * (E
ExtraPcg(J,1)==0))- 100 * (V TD(J - ( (A Grupo-1) * 12))))
2887$      BRANCH,      1:
                        If,J<>0,2888$,Yes:
                        Else,2889$,Yes
2888$      DELAY:      0.0,,VA:NEXT(971$)
2889$      DELAY:      0.0,,VA:NEXT(897$)
      Model statements for module: Assign 747
971$      ASSIGN:      A aTemp3=J:NEXT(898$)
      Model statements for module: Assign 177
898$      ASSIGN:      A Rota=E RotZero (A Tmp, A aTemp3-Shift):
                        A PtDestino=V IndicePtCg (A aTemp3-Shift):
                        A PtOrigem=A PtZero:
                        A PtAtual=A PtOrigem:
                        V DA(A aTemp3)=V DA(A aTemp3) + 1:
                        A Estado=0:
                        A IndiRota=1:
                        A aTemp3=J:
                        V TD ( A aTemp3 - ( (A Grupo-1) * 12) )=
                        V TD(A aTemp3 - ( (A Grupo-1) * 12)) + (1/E Patios (A
PtDestino, 2)):
                        A Peso=0:
                        A Cliente=0:NEXT(946$)
      Model statements for module: Decide 398
946$      BRANCH,      1:
                        If,A Rota <> 0,2890$,Yes:
                        Else,2891$,Yes
2890$      ASSIGN:      Este roteiro existe mesmo?.NumberOut True=Este roteiro
existe mesmo?.NumberOut True + 1:NEXT(920$)
2891$      ASSIGN:      Este roteiro existe mesmo?.NumberOut False=Este roteiro
existe mesmo?.NumberOut False + 1
                        :NEXT(947$)
      Model statements for module: Decide 203
920$      BRANCH,      1:
                        If,chechar >= 0 .AND. checar < TNOW.OR.A
IDTrem==Rastreo,2892$,Yes:
                        Else,2893$,Yes
2892$      ASSIGN:      Ponto de checagem 9.NumberOut True=Ponto de checagem
9.NumberOut True + 1:NEXT(921$)
2893$      ASSIGN:      Ponto de checagem 9.NumberOut False=Ponto de checagem
9.NumberOut False + 1:NEXT(925$)

```



```

sh12=(900000 * (E ExtraPcg(J,1)==0)):
sh13=100 * (V TD(J - (A Grupo-1) * 12)):
sh14=V TD(J):NEXT(950$)
Model statements for module: Hold 75
950$    QUEUE,          Hold 75.Queue:DETACH
Model statements for module: Assign 187
901$    ASSIGN:        A Codigo=1:
                          A Cliente=1:NEXT(937$)
Model statements for module: Assign 416
937$    ASSIGN:        A EscolhaTMP=A Tmp:
                          A EscolhaCDG=A Codigo:
                          A Prioridade=2 + (A Tipo==3):NEXT(941$)
Model statements for module: Assign 417
941$    ASSIGN:        MarcaCiclo=tnow:
                          A Etapa=1:
                          Entity.Picture=Figuras Vazios(A Produto):
                          A Tcarga=A PtDestino:NEXT(957$)

151 = FPK
7 = FJC
148 = FDM
Model statements for module: Decide 493
957$    BRANCH,        1:
                          If,A PtZero == 151 .OR. A PtZero == 7 .OR. A PtZero ==
148,2901$,Yes:
                          Else,2902$,Yes
2901$    ASSIGN:        Ponto zero FPK ou FJC ou FDM ?.NumberOut True=Ponto zero
FPK ou FJC ou FDM ?.NumberOut True + 1
                          :NEXT(910$)
2902$    ASSIGN:        Ponto zero FPK ou FJC ou FDM ?.NumberOut False=Ponto
zero FPK ou FJC ou FDM ?.NumberOut False + 1
                          :NEXT(958$)
Model statements for module: Decide 166
910$    BRANCH,        1:
                          If,A Rota <> 0,2903$,Yes:
                          Else,2904$,Yes
2903$    ASSIGN:        Roteiro existe ?.NumberOut True=Roteiro existe
?.NumberOut True + 1:NEXT(935$)
2904$    ASSIGN:        Roteiro existe ?.NumberOut False=Roteiro existe
?.NumberOut False + 1:NEXT(911$)
Model statements for module: Decide 330
935$    BRANCH,        1:
                          If,A IDTrem==1.OR.1,2905$,Yes:
                          Else,2906$,Yes
2905$    ASSIGN:        Decide 330.NumberOut True=Decide 330.NumberOut True +
1:NEXT(929$)
2906$    ASSIGN:        Decide 330.NumberOut False=Decide 330.NumberOut False +
1:NEXT(936$)
Model statements for module: Decide 224
929$    BRANCH,        1:
                          If,chechar >= 0 .AND. chechar < TNOW.OR.A
IDTrem==Rastreio,2907$,Yes:
                          Else,2908$,Yes
2907$    ASSIGN:        Ponto de checagem 12.NumberOut True=Ponto de checagem
12.NumberOut True + 1:NEXT(926$)
2908$    ASSIGN:        Ponto de checagem 12.NumberOut False=Ponto de checagem
12.NumberOut False + 1:NEXT(881$)
Model statements for module: Separate 27
926$    DUPLICATE,     100 - 50:
                          1,2911$,50:NEXT(2910$)
2910$    ASSIGN:        Separate 27.NumberOut Orig=Separate 27.NumberOut Orig +
1:NEXT(927$)
2911$    ASSIGN:        Separate 27.NumberOut Dup=Separate 27.NumberOut Dup +
1:NEXT(881$)
Model statements for module: Hold 40
927$    QUEUE,          Hold 40.Queue:DETACH

```

```

Model statements for module: Assign 170
881$      ASSIGN:      A Sentido=V DROT(A IndiRota, A Rota):
                        A Trecho=V ROT(A IndiRota, A Rota):NEXT(877$)
Model statements for module: Decide 155
877$      BRANCH,      1:
                        If,V Trecho ( A Trecho , 2 ) == A PtOrigem,2912$,Yes:
                        Else,2913$,Yes
2912$      ASSIGN:      2 Patio de origem do trem esta neste trecho ?.NumberOut
True=
                        2 Patio de origem do trem esta neste trecho ?.NumberOut
True + 1:NEXT(963$)
2913$      ASSIGN:      2 Patio de origem do trem esta neste trecho ?.NumberOut
False=
                        2 Patio de origem do trem esta neste trecho ?.NumberOut
False + 1:NEXT(876$)
Model statements for module: Decide 614
963$      BRANCH,      1:
                        If,A Tipo <> 2 .OR. (A Patio <> 151 .AND. A Patio <> 7
                        .AND. A Patio <> 148),2914$,Yes:
                        Else,2915$,Yes
2914$      ASSIGN:      Tabela vazio nao esta saindo de ponto zero ?.NumberOut
True=
                        Tabela vazio nao esta saindo de ponto zero ?.NumberOut
True + 1:NEXT(974$)
2915$      ASSIGN:      Tabela vazio nao esta saindo de ponto zero ?.NumberOut
False=
                        Tabela vazio nao esta saindo de ponto zero ?.NumberOut
False + 1:NEXT(932$)
Model statements for module: Assign 738
974$      ASSIGN:      Temp2=tnow:NEXT(977$)
Model statements for module: Decide 679
977$      BRANCH,      1:
                        If,V Ocpt ( A PtOrigem , 1+ (A Sentido==0) ) ==
0,2916$,Yes:
                        Else,2917$,Yes
2916$      ASSIGN:      2 Patio no sentido correto esta livre ?.NumberOut True=
                        2 Patio no sentido correto esta livre ?.NumberOut True +
1:NEXT(975$)
2917$      ASSIGN:      2 Patio no sentido correto esta livre ?.NumberOut False=
                        2 Patio no sentido correto esta livre ?.NumberOut False
+ 1:NEXT(973$)
Model statements for module: Decide 678
975$      BRANCH,      1:
                        If,(TNOW - Temp2) > 120,2918$,Yes:
                        Else,2919$,Yes
2918$      ASSIGN:      Tabela tentando ha mais de 2 horas ?.NumberOut True=
                        Tabela tentando ha mais de 2 horas ?.NumberOut True +
1:NEXT(976$)
2919$      ASSIGN:      Tabela tentando ha mais de 2 horas ?.NumberOut False=
                        Tabela tentando ha mais de 2 horas ?.NumberOut False +
1:NEXT(932$)
Model statements for module: ReadWrite 195
976$      WRITE,      Log de Erros,
                        "%1.8g. TREM ATRASADO ERR20. Trem tipo %1.8g numero
%1.8g. Patio de origem %1.8g atrasou %1.8g minutos. \n":
                        TNOW,
                        A Tipo,
                        A Ntrem,
                        A PtOrigem,
                        TNOW - Temp2:NEXT(932$)
Model statements for module: Delay 59
973$      DELAY:      SecondsToBaseTime(norm(30,10)),,Other:NEXT(977$)
Model statements for module: Assign 315
932$      ASSIGN:      A PtAtual=V Trecho ( A Trecho , 2 ):
                        A Tmp=8123:NEXT(931$)

```

```

Model statements for module: Decide 233
931$      BRANCH,      1:
                If,E Verifica entroncamento .OR. A Direto==1,2920$,Yes:
                Else,2921$,Yes
2920$      ASSIGN:      Unitario vai entrar em um entroncamento ?.NumberOut
True=
                Unitario vai entrar em um entroncamento ?.NumberOut True
+ 1:NEXT(930$)
2921$      ASSIGN:      Unitario vai entrar em um entroncamento ?.NumberOut
False=
                Unitario vai entrar em um entroncamento ?.NumberOut
False + 1:NEXT(882$)
Model statements for module: Assign 312
930$      ASSIGN:      V Conta bypass=V Conta bypass + 1:NEXT(899$)
Model statements for module: Assign 178
899$      ASSIGN:      Entity.Station=Station Origem de todos:NEXT(952$)
Model statements for module: Decide 429
952$      BRANCH,      1:
                If,TNOW > (V Warmup * 1440),2922$,Yes:
                Else,2923$,Yes
2922$      ASSIGN:      Warmup do log de eventos 12.NumberOut True=Warmup do log
de eventos 12.NumberOut True + 1
                :NEXT(965$)
2923$      ASSIGN:      Warmup do log de eventos 12.NumberOut False=Warmup do
log de eventos 12.NumberOut False + 1
                :NEXT(918$)
Model statements for module: Decide 636
965$      BRANCH,      1:
                If,V Gravar em texto == 1,2924$,Yes:
                Else,2925$,Yes
2924$      ASSIGN:      Decide 636.NumberOut True=Decide 636.NumberOut True +
1:NEXT(967$)
2925$      ASSIGN:      Decide 636.NumberOut False=Decide 636.NumberOut False +
1:NEXT(924$)
    Especificação do arquivo de LOG de eventos:
    1 - Dia do evento
    2- Identificador único do trem (A IDTrem)
    3 - Número do trem (A NTrem)
    4 - Tipo do trem (A Tipo)
    5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
    6 - Pátio (A PtAtual)
    7 - Momento da saída do trem (TNOW)
    8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 162
967$      WRITE,      Log de Eventos:
                Dia,
                A IDtrem,
                A Ntrem,
                A Tipo,
                marca,
                A PtAtual,
                tnow,
                6,
                A Peso,
                A Dist,
                E FrotasT (A Ntrem, 1),
                E FrotasT (A Ntrem, 2),
                E FrotasT (A Ntrem, 3),
                E FrotasT (A Ntrem, 4),
                E FrotasT (A Ntrem, 5),
                A Estado,
                A Produto,
                A Cliente,

```

```

        A PtOrigem,
        A PtDestino,
        A PtZero:NEXT(918$)
Model statements for module: Decide 197
918$      BRANCH,      1:
                If,A ColPatio == 0,2926$,Yes:
                Else,2927$,Yes
2926$      ASSIGN:      Nao esta ocupando linha ?.NumberOut True=Nao esta
ocupando linha ?.NumberOut True + 1:NEXT(883$)
2927$      ASSIGN:      Nao esta ocupando linha ?.NumberOut False=Nao esta
ocupando linha ?.NumberOut False + 1:NEXT(919$)
Model statements for module: Route 10
883$      ROUTE:      0.000000000000000,ESTA ( A PtAtual,1 + ( A Sentido == 0)
)
Model statements for module: Assign 245
919$      ASSIGN:      V StatusPt (A PtAtual, A ColPatio)=V StatusPt (A
PtAtual, A ColPatio) + 1:NEXT(883$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 50
924$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeEvento):
                Dia,
                A IDtrem,
                A Ntrem,
                A Tipo,
                marca,
                A PtAtual,
                tnow,
                6,
                A Peso,
                A Dist,
                E FrotasT (A Ntrem, 1),
                E FrotasT (A Ntrem, 2),
                E FrotasT (A Ntrem, 3),
                E FrotasT (A Ntrem, 4),
                E FrotasT (A Ntrem, 5),
                A Estado,
                A Produto,
                A Cliente,
                A PtOrigem,
                A PtDestino,
                A PtZero:NEXT(918$)
Model statements for module: Assign 171
882$      ASSIGN:      A PtAtual=V Trecho ( A Trecho , 2 ):
                V Ocpt ( A PtAtual , 1+ (A Sentido == 0))=1:
                A Tmp=8123:NEXT(899$)
Model statements for module: Assign 169
876$      ASSIGN:      A IndiRota=A IndiRota + 1:NEXT(878$)
Model statements for module: Decide 156
878$      BRANCH,      1:
                If,V ROT(A IndiRota, A Rota) == 0,2928$,Yes:
                Else,2929$,Yes
2928$      ASSIGN:      2 Acabaram os trechos do roteiro sem achar a estacao
?.NumberOut True=
                2 Acabaram os trechos do roteiro sem achar a estacao
?.NumberOut True + 1:NEXT(879$)

```



```

2929$      ASSIGN:          2 Acabaram os trechos do roteiro sem achar a estacao
?.NumberOut False=
          2 Acabaram os trechos do roteiro sem achar a estacao
?.NumberOut False + 1:NEXT(881$)
      Model statements for module:  ReadWrite 20
879$      WRITE,          Log de Erros,
          "%1.8g. ERR2 Erro na Rota %1.8g. Nao encontrado patio
numero %1.8g de origem do trem. Trem abortado.\n":
          TNOW,
          A Rota,
          A PtOrigem:NEXT(880$)
      Model statements for module:  Dispose 26
880$      ASSIGN:          Dispose 26.NumberOut=Dispose 26.NumberOut + 1
2930$      DISPOSE:        Yes
      Model statements for module:  Dispose 56
936$      ASSIGN:          Dispose 56.NumberOut=Dispose 56.NumberOut + 1
2931$      DISPOSE:        Yes
      Model statements for module:  ReadWrite 21
911$      WRITE,          Log de Erros,
          "%1.8g. ERR6 Erro no trem tipo %1.8g e numero %1.8g.
Escolhida rota zerada em %1.8g. Pto Zero %1.8g. Com destino a %1.8g. Trem
abortado.\n":
          TNOW,
          A Tipo,
          A Ntrem,
          A PtOrigem,
          A PtZero,
          A PtDestino:NEXT(912$)
      Model statements for module:  Dispose 31
912$      ASSIGN:          Dispose 31.NumberOut=Dispose 31.NumberOut + 1
2932$      DISPOSE:        Yes
      Model statements for module:  Assign 582
958$      ASSIGN:          A Tcarga=A PtOrigem:NEXT(910$)
      Model statements for module:  Assign 188
902$      ASSIGN:          A Codigo=2:
          A Cliente=2:NEXT(937$)
      Model statements for module:  Assign 189
903$      ASSIGN:          A Codigo=3:
          A Cliente=3 + 5 * (A Tipo == 3):NEXT(945$)
      Model statements for module:  Decide 349
945$      BRANCH,         1:
          If,A Cliente == 8,2933$,Yes:
          Else,2934$,Yes
2933$      ASSIGN:          Trem de Bauxita ?.NumberOut True=Trem de Bauxita
?.NumberOut True + 1:NEXT(943$)
2934$      ASSIGN:          Trem de Bauxita ?.NumberOut False=Trem de Bauxita
?.NumberOut False + 1:NEXT(937$)
      Model statements for module:  Assign 422
943$      ASSIGN:          A Codigo=3:
          A Rota=E RotPcg ( (A Grupo - 15) + 12, 3):
          A PtDestino=48:
          A PtOrigem=A PtZero:
          A Cliente=8:NEXT(937$)
      Model statements for module:  Assign 190
904$      ASSIGN:          A Codigo=4:
          A Cliente=4:NEXT(937$)
      Model statements for module:  Assign 193
905$      ASSIGN:          A Codigo=5:
          A Cliente=5:NEXT(937$)
      Model statements for module:  Assign 194
906$      ASSIGN:          A Codigo=6:
          A Cliente=6:NEXT(937$)
      Model statements for module:  Assign 195
907$      ASSIGN:          A Codigo=7:
          A Cliente=7:NEXT(937$)

```

```

Model statements for module: Assign 196
908$      ASSIGN:      A Codigo=8:
                        A Cliente=9:NEXT(944$)
Model statements for module: Assign 423
944$      ASSIGN:      A Codigo=8:
                        A Rota=E RotPcg ( (A Grupo - 15) + 12, 8):
                        A PtDestino=233:
                        A PtOrigem=A PtZero:
                        A Cliente=9:NEXT(937$)
Model statements for module: Assign 197
909$      ASSIGN:      A Codigo=9:
                        A Cliente=13:NEXT(937$)
Model statements for module: Assign 198
913$      ASSIGN:      A Codigo=10:
                        A Cliente=130:NEXT(937$)
Model statements for module: Assign 199
914$      ASSIGN:      A Codigo=11:
                        A Cliente=14:NEXT(937$)
Model statements for module: Assign 201
915$      ASSIGN:      A Codigo=12:
                        A Cliente=15:NEXT(937$)
Model statements for module: Assign 202
916$      ASSIGN:      A Codigo=13:
                        A Cliente=16:NEXT(937$)
Model statements for module: Assign 204
917$      ASSIGN:      A Codigo=14:
                        A Cliente=17:NEXT(937$)
Model statements for module: Assign 521
956$      ASSIGN:      A Codigo=15:
                        A Cliente=12:NEXT(937$)
Model statements for module: ReadWrite 74
947$      WRITE,      Log de Erros,
                        "%1.8g. ERR12 Trem tipo %1.8g numero %1.8g. Rota zerada
em %1.8g. Pto Zero %1.8g. Com destino a %1.8g. Linha da var %1.8g.Trem abortado.\n":
                        TNOW,
                        A Tipo,
                        A Ntrem,
                        A PtOrigem,
                        A PtZero,
                        A PtDestino,
                        A aTemp3:NEXT(948$)
Model statements for module: Dispose 59
948$      ASSIGN:      Dispose 59.NumberOut=Dispose 59.NumberOut + 1
2935$     DISPOSE:      Yes
Model statements for module: Dispose 28
897$      ASSIGN:      Dispose 28.NumberOut=Dispose 28.NumberOut + 1
2936$     DISPOSE:      Yes
Model statements for module: Assign 593
960$      ASSIGN:      V Trens na Malha(1)=V Trens na Malha(1) + 1:NEXT(891$)
Model statements for module: Assign 172
891$      ASSIGN:      A Tmp=(A Grupo - 15) + 180:
                        A Estado=1:
                        A IndiRota=1:
                        A Peso=E Tabelas (A NTrem, 3) * E PesosT(A NTrem,1):
                        A Etapa=1:NEXT(900$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)

```

8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso  
5=outros)

```

Model statements for module: ReadWrite 60
942$      WRITE,          Log de MovimentacaoMDB,RECORDSET(LogdeCiclos):
                          Dia,
                          A IDtrem,
                          A Ntrem,
                          A Tipo,
                          A PtOrigem,
                          A PtAtual,
                          tnow - MarcaCiclo,
                          A Produto,
                          A Estado,
                          A Cliente,
                          A PtZero,
                          3,
                          A Tcarga:NEXT(885$)

Model statements for module: Create 8
2937$     CREATE,        1,DaysToBaseTime(0),Entity 1:DaysToBaseTime(V
DAD):NEXT(2938$)
2938$     ASSIGN:       Cria entidade atualizadora a cada V DAD dias.NumberOut=
                          Cria entidade atualizadora a cada V DAD dias.NumberOut +
1:NEXT(961$)

Model statements for module: Decide 520
961$     BRANCH,        1:
                          If, TNOW > (V Warmup * 1440),2941$,Yes:
                          Else,2942$,Yes

2941$     ASSIGN:       Warmup da atualizacao de demanda.NumberOut True=Warmup
da atualizacao de demanda.NumberOut True + 1
                          :NEXT(893$)
2942$     ASSIGN:       Warmup da atualizacao de demanda.NumberOut False=
                          Warmup da atualizacao de demanda.NumberOut False +
1:NEXT(962$)

Model statements for module: Assign 174
893$     ASSIGN:       Indice=1:NEXT(894$)

Model statements for module: Assign 175
894$     ASSIGN:       V DO(Indice)=V DO(Indice) + (E ExtraPCg(Indice,1)*V
DAD)/30 - V DA(Indice):
                          V DA(Indice)=0:
                          Indice=Indice + 1:NEXT(895$)

Model statements for module: Decide 161
895$     BRANCH,        1:
                          If,Indice>190,2943$,Yes:
                          Else,2944$,Yes

2943$     ASSIGN:       Terminaram as linhas ?.NumberOut True=Terminaram as
linhas ?.NumberOut True + 1:NEXT(896$)
2944$     ASSIGN:       Terminaram as linhas ?.NumberOut False=Terminaram as
linhas ?.NumberOut False + 1:NEXT(894$)

Model statements for module: Dispose 27
896$     ASSIGN:       Dispose 27.NumberOut=Dispose 27.NumberOut + 1
2945$     DISPOSE:      Yes

Model statements for module: Dispose 72
962$     ASSIGN:       Dispose 72.NumberOut=Dispose 72.NumberOut + 1
2946$     DISPOSE:      Yes

Model statements for module: Hold 50
933$     QUEUE,        Hold 50.Queue:DETACH

Model statements for module: Assign 739
968$     ASSIGN:       A Tmp=A aTemp3:
                          A Estado=0:
                          A IndiRota=1:
                          A Etapa=1:NEXT(938$)

Model statements for module: Decide 685
969$     BRANCH,        1:
                          If,E ExtraPcg(A aTemp3,1)>0,2947$,Yes:
                          Else,2948$,Yes

```

```

2947$      ASSIGN:      Decide 685.NumberOut True=Decide 685.NumberOut True +
1:NEXT(972$)
2948$      ASSIGN:      Decide 685.NumberOut False=Decide 685.NumberOut False +
1:NEXT(970$)
      Model statements for module:  Assign 748
972$      ASSIGN:      A Rota=E RotZero (A Tmp, A aTemp3-Shift):
      A PtDestino=V IndicePtCg (A aTemp3-Shift):
      A PtOrigem=A PtZero:
      A PtAtual=A PtOrigem:
      V DA(A aTemp3)=V DA(A aTemp3) + 1:
      A Estado=0:
      A IndiRota=1:
      V TD ( A aTemp3 - ( (A Grupo-1) * 12) )=
      V TD(A aTemp3 - ( (A Grupo-1) * 12)) + (1/E Patios (A
PtDestino, 2)):
      A Peso=0:
      A Cliente=0:NEXT(946$)
      Model statements for module:  Assign 746
970$      ASSIGN:      A aTemp3=anint (UNIF(1+Shift,12+Shift)):NEXT(969$)
      Model statements for module:  Station 10
978$      STATION,      Logica de Patios
2951$      DELAY:      0.0,,VA:NEXT(979$)
      Model statements for module:  Decide 167
979$      BRANCH,      1:
      If,A Tipo == 1,1108$,Yes:
      If,A Tipo == 2,1009$,Yes:
      If,A Tipo == 3,1006$,Yes:
      If,A Tipo == 4,1136$,Yes:
      If,A Tipo == 5,1136$,Yes:
      Else,980$,Yes
      Model statements for module:  Dispose 32
980$      ASSIGN:      Dispose 32.NumberOut=Dispose 32.NumberOut + 1
2954$      DISPOSE:      Yes
      Model statements for module:  Decide 438
1108$     BRANCH,      1:
      If,TNOW > (V Warmup * 1440),2955$,Yes:
      Else,2956$,Yes
2955$     ASSIGN:      Warmup do log de ciclo etapa 2.NumberOut True=Warmup do
log de ciclo etapa 2.NumberOut True + 1
      :NEXT(1161$)
2956$     ASSIGN:      Warmup do log de ciclo etapa 2.NumberOut False=Warmup do
log de ciclo etapa 2.NumberOut False + 1
      :NEXT(1156$)
      Model statements for module:  Decide 641
1161$     BRANCH,      1:
      If,V Gravar em texto == 1,2957$,Yes:
      Else,2958$,Yes
2957$     ASSIGN:      Decide 641.NumberOut True=Decide 641.NumberOut True +
1:NEXT(1167$)
2958$     ASSIGN:      Decide 641.NumberOut False=Decide 641.NumberOut False +
1:NEXT(1025$)
      Especificação do arquivo de LOG de eventos:
      1 - Dia do evento
      2- Identificador único do trem (A IDTrem)
      3 - Número do trem (A NTrem)
      4 - Tipo do trem (A Tipo)
      5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
      6 - Pátio (A PtAtual)
      7 - Momento da saída do trem (TNOW)
      8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
      Model statements for module:  ReadWrite 164
1167$     WRITE,      Log de Ciclos:
      Dia,

```

```

A IDtrem,
A Ntrem,
A Tipo,
A PtOrigem,
A PtAtual,
tnow - MarcaCiclo,
A Produto,
A Estado,
A Cliente,
A PtZero,
2,
A Tcarga:NEXT(1156$)
Model statements for module: Assign 588
1156$    ASSIGN:      V Trens na Malha(1)=V Trens na Malha(1) - 1:NEXT(1105$)
Model statements for module: Decide 418
1105$    BRANCH,      1:
                        If,E Patios (A PtAtual,10)==12,2959$,Yes:
                        Else,2960$,Yes
2959$    ASSIGN:      Trem saindo do trecho de FVR para descarregar
?.NumberOut True=
                        Trem saindo do trecho de FVR para descarregar
?.NumberOut True + 1:NEXT(1106$)
2960$    ASSIGN:      Trem saindo do trecho de FVR para descarregar
?.NumberOut False=
                        Trem saindo do trecho de FVR para descarregar
?.NumberOut False + 1:NEXT(1059$)
Model statements for module: Assign 466
1106$    ASSIGN:      V ContagemFVR=V ContagemFVR - 1:NEXT(1059$)
Model statements for module: Decide 339
1059$    BRANCH,      1:
                        If,chechar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2961$,Yes:
                        Else,2962$,Yes
2961$    ASSIGN:      Ponto de checagem 28.NumberOut True=Ponto de checagem
28.NumberOut True + 1:NEXT(1080$)
2962$    ASSIGN:      Ponto de checagem 28.NumberOut False=Ponto de checagem
28.NumberOut False + 1:NEXT(1020$)
Model statements for module: Separate 51
1080$    DUPLICATE,    100 - 50:
                        1,2965$,50:NEXT(2964$)
2964$    ASSIGN:      Separate 51.NumberOut Orig=Separate 51.NumberOut Orig +
1:NEXT(1081$)
2965$    ASSIGN:      Separate 51.NumberOut Dup=Separate 51.NumberOut Dup +
1:NEXT(1020$)
Model statements for module: Hold 64
1081$    QUEUE,        Hold 64.Queue:DETACH
Model statements for module: Assign 267
1020$    ASSIGN:      V LocosNow(1)=V LocosNow(1) - E FrotasT (A Ntrem, 1):
V LocosNow(2)=V LocosNow(2) - E FrotasT (A Ntrem, 2):
V LocosNow(3)=V LocosNow(3) - E FrotasT (A Ntrem, 3):
V LocosNow(4)=V LocosNow(4) - E FrotasT (A Ntrem, 4):
V LocosNow(5)=V LocosNow(5) - E FrotasT (A Ntrem, 5):
MarcaCT=TNOW:
marca=TNOW:
A Entrada=0:NEXT(1109$)
Model statements for module: Decide 439
1109$    BRANCH,      1:
                        If, TNOW > (V Warmup * 1440),2966$,Yes:
                        Else,2967$,Yes
2966$    ASSIGN:      Warmup do log de eventos 6.NumberOut True=Warmup do log
de eventos 6.NumberOut True + 1:NEXT(1160$)
2967$    ASSIGN:      Warmup do log de eventos 6.NumberOut False=Warmup do log
de eventos 6.NumberOut False + 1
                        :NEXT(1010$)
Model statements for module: Decide 640

```

```

1160$      BRANCH,          1:
                                If,V Gravar em texto == 1,2968$,Yes:
                                Else,2969$,Yes
2968$      ASSIGN:          Decide 640.NumberOut True=Decide 640.NumberOut True +
1:NEXT(1168$)
2969$      ASSIGN:          Decide 640.NumberOut False=Decide 640.NumberOut False +
1:NEXT(1022$)
    Especificação do arquivo de LOG de eventos:
    1 - Dia do evento
    2- Identificador único do trem (A IDTrem)
    3 - Número do trem (A NTrem)
    4 - Tipo do trem (A Tipo)
    5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
    6 - Pátio (A PtAtual)
    7 - Momento da saída do trem (TNOW)
    8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
    Model statements for module:  ReadWrite 165
1168$      WRITE,          Log de Eventos:
                                Dia,
                                A IDtrem,
                                A Ntrem,
                                A Tipo,
                                marca,
                                A PtAtual,
                                tnow,
                                4,
                                A Peso,
                                A Dist,
                                E FrotasT (A Ntrem, 1),
                                E FrotasT (A Ntrem, 2),
                                E FrotasT (A Ntrem, 3),
                                E FrotasT (A Ntrem, 4),
                                E FrotasT (A Ntrem, 5),
                                A Estado,
                                A Produto,
                                A Cliente,
                                A Tcarga,
                                A PtAtual,
                                A PtZero:NEXT(1010$)
    Model statements for module:  Decide 194
1010$      BRANCH,          1:
                                If,A ColPatio == 1,2970$,Yes:
                                Else,2971$,Yes
2970$      ASSIGN:          Trem reservou linha de espera para descarga?.NumberOut
True=
                                Trem reservou linha de espera para descarga?.NumberOut
True + 1:NEXT(1091$)
2971$      ASSIGN:          Trem reservou linha de espera para descarga?.NumberOut
False=
                                Trem reservou linha de espera para descarga?.NumberOut
False + 1:NEXT(1087$)
    Model statements for module:  Assign 436
1091$      ASSIGN:          Sorteio=Unif(4000,8000)+Unif(1,350):NEXT(1030$)
    Model statements for module:  Separate 24
1030$      DUPLICATE,      100 - 50:
                                1,2974$,50:NEXT(2973$)
2973$      ASSIGN:          Separate 24.NumberOut Orig=Separate 24.NumberOut Orig +
1:NEXT(1089$)
2974$      ASSIGN:          Separate 24.NumberOut Dup=Separate 24.NumberOut Dup +
1:NEXT(1129$)
    Model statements for module:  Assign 435
1089$      ASSIGN:          Picture=Picture.espdesc:NEXT(1033$)
    Model statements for module:  Route 16

```



```

        tnow - marca,
        A Dist,
        E FrotasT (A Ntrem, 1),
        E FrotasT (A Ntrem, 2),
        E FrotasT (A Ntrem, 3),
        E FrotasT (A Ntrem, 4),
        E FrotasT (A Ntrem, 5),
        A Estado,
        A Produto,
        A Cliente,
        A Tcarga,
        A PtAtual,
        A PtZero:NEXT(1087$)
    Model statements for module: Assign 434
1087$    ASSIGN:      Sorteio=Unif(12000,18000)+Unif(1,350):
        V StatusPtD(A Cliente)=V StatusPtD(A Cliente) +
1:NEXT(1088$)
    Model statements for module: Separate 55
1088$    DUPLICATE,    100 - 50:
        1,2983$,50:NEXT(2982$)
2982$    ASSIGN:      Separate 55.NumberOut Orig=Separate 55.NumberOut Orig +
1:NEXT(1031$)
2983$    ASSIGN:      Separate 55.NumberOut Dup=Separate 55.NumberOut Dup +
1:NEXT(1047$)
    Model statements for module: Assign 291
1031$    ASSIGN:      Picture=Picture.descarga:NEXT(1033$)
    Model statements for module: Decide 298
1047$    BRANCH,      1:
        If,A Tipo == 3,2984$,Yes:
        Else,2985$,Yes
2984$    ASSIGN:      Tabela nao MFE ?.NumberOut True=Tabela nao MFE
?.NumberOut True + 1:NEXT(982$)
2985$    ASSIGN:      Tabela nao MFE ?.NumberOut False=Tabela nao MFE
?.NumberOut False + 1:NEXT(1048$)
    Model statements for module: Assign 206
982$     ASSIGN:      marca=tnow:
        A Estado=0:
        A Peso=0:NEXT(983$)
    Model statements for module: Delay 24
983$     DELAY:      E Tabelas (A NTrem, 6)/E PtProd(A PtAtual, (A Produto *
(A Produto <> 2) + 1 * (A Produto == 2))+5),,
        Other:NEXT(1114$)
    Model statements for module: Decide 444
1114$    BRANCH,      1:
        If, TNOW > (V Warmup * 1440),2986$,Yes:
        Else,2987$,Yes
2986$    ASSIGN:      Warmup do log de eventos 10.NumberOut True=Warmup do log
de eventos 10.NumberOut True + 1
        :NEXT(1158$)
2987$    ASSIGN:      Warmup do log de eventos 10.NumberOut False=Warmup do
log de eventos 10.NumberOut False + 1
        :NEXT(999$)
    Model statements for module: Decide 638
1158$    BRANCH,      1:
        If,V Gravar em texto == 1,2988$,Yes:
        Else,2989$,Yes
2988$    ASSIGN:      Decide 638.NumberOut True=Decide 638.NumberOut True +
1:NEXT(1170$)
2989$    ASSIGN:      Decide 638.NumberOut False=Decide 638.NumberOut False +
1:NEXT(1014$)
    Especificação do arquivo de LOG de eventos:
    1 - Dia do evento
    2- Identificador único do trem (A IDTrem)
    3 - Número do trem (A NTrem)
    4 - Tipo do trem (A Tipo)

```



5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário - grade/expresso)  
 6 - Pátio (A PtAtual)  
 7 - Momento da saída do trem (TNOW)  
 8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saída de grade ou expresso 5=outras)

Model statements for module: ReadWrite 167

```
1170$      WRITE,          Log de Eventos:
                          Dia,
                          A IDtrem,
                          A Ntrem,
                          A Tipo,
                          marca,
                          A PtAtual,
                          tnow,
                          2,
                          A Peso,
                          A Dist,
                          E FrotasT (A Ntrem, 1),
                          E FrotasT (A Ntrem, 2),
                          E FrotasT (A Ntrem, 3),
                          E FrotasT (A Ntrem, 4),
                          E FrotasT (A Ntrem, 5),
                          A Estado,
                          A Produto,
                          A Cliente,
                          A Tcarga,
                          A PtAtual,
                          A PtZero:NEXT(999$)
```

151 = FPK

7 = FJC

148 = FDM

Model statements for module: Decide 172

```
999$      BRANCH,        1:
                          If,A PtZero == 151,1000$,Yes:
                          If,A PtZero == 7,1001$,Yes:
                          If,A PtZero == 148,1002$,Yes:
                          If,A PtZero == 41,1003$,Yes:
                          If,A PtZero == 69,1004$,Yes:
                          If,A PtZero == 72,1005$,Yes:
                          If,A PtZero == 89,1055$,Yes:
                          If,A PtZero == 359,1126$,Yes:
                          If,A PtZero == 14,1127$,Yes:
                          If,A PtZero == 20,1128$,Yes:
                          Else,1038$,Yes
```

Model statements for module: Hold 38

```
1038$     QUEUE,          Hold 38.Queue:DETACH
```

Model statements for module: Assign 212

```
1000$     ASSIGN:        Temp1=1:
                          A Tipo=2:NEXT(998$)
```

Model statements for module: Assign 211

```
998$     ASSIGN:        V StatusPtD(A Cliente)=V StatusPtD(A Cliente) - 1:
                          A Rota=E RotPDcg (A Codigo, Temp1):
                          A PtOrigem=A PtDestino:
                          A PtDestino=A PtZero:
                          A IndiRota=1:
                          A Estado=0:
                          A Dist=0:
                          A ColPatio=3:
                          A Etapa=3:
                          Temp2=A Produto * (A Produto <> 2) + 1 * (A Produto ==
2):
                          Entity.Picture=Figuras Vazios(A Produto):NEXT(1032$)
```

Model statements for module: Signal 23

```
1032$     SIGNAL:        sorteio,1:NEXT(994$)
```

```

Model statements for module: Assign 210
994$      ASSIGN:      Entity.Station=Station Origem de todos:
                                Temp2=A Produto * (A Produto <> 2) + 1 * (A Produto ==
2):
                                V StatusPtProd (A Ptatual, Temp2 + 5)=V StatusPtProd (A
Ptatual, Temp2 + 5) - 1:
                                V StatusPt (A PtAtual, A ColPatio)=V StatusPt (A
PtAtual, A ColPatio) + 1:
                                A ColPatio=0:
                                MarcaCiclo=tnow:
                                A Etapa=3:NEXT(1154$)
Model statements for module: Assign 586
1154$     ASSIGN:      V Trens na Malha(2)=V Trens na Malha(2) + 1:
                                A Prioridade=2 + (A Tipo==3):NEXT(1097$)
Model statements for module: Decide 385
1097$     BRANCH,      1:
                                If, checar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),2992$,Yes:
                                Else,2993$,Yes
2992$     ASSIGN:      Ponto de checagem 29.NumberOut True=Ponto de checagem
29.NumberOut True + 1:NEXT(1098$)
2993$     ASSIGN:      Ponto de checagem 29.NumberOut False=Ponto de checagem
29.NumberOut False + 1:NEXT(1053$)
Model statements for module: Separate 59
1098$     DUPLICATE,    100 - 50:
                                1,2996$,50:NEXT(2995$)
2995$     ASSIGN:      Separate 59.NumberOut Orig=Separate 59.NumberOut Orig +
1:NEXT(1099$)
2996$     ASSIGN:      Separate 59.NumberOut Dup=Separate 59.NumberOut Dup +
1:NEXT(1053$)
Model statements for module: Hold 69
1099$     QUEUE,        Hold 69.Queue:DETACH
Model statements for module: Decide 322
1053$     BRANCH,      1:
                                If,A PtOrigem <> A PtZero,2997$,Yes:
                                Else,2998$,Yes
2997$     ASSIGN:      Ponto de descarga diferente do ponto de direcionamento
?.NumberOut True=
                                Ponto de descarga diferente do ponto de direcionamento
?.NumberOut True + 1:NEXT(995$)
2998$     ASSIGN:      Ponto de descarga diferente do ponto de direcionamento
?.NumberOut False=
                                Ponto de descarga diferente do ponto de direcionamento
?.NumberOut False + 1:NEXT(1118$)
Model statements for module: Decide 171
995$      BRANCH,      1:
                                If,A Rota <> 0,2999$,Yes:
                                Else,3000$,Yes
2999$     ASSIGN:      Este Roteiro existe ?.NumberOut True=Este Roteiro existe
?.NumberOut True + 1:NEXT(989$)
3000$     ASSIGN:      Este Roteiro existe ?.NumberOut False=Este Roteiro
existe ?.NumberOut False + 1:NEXT(996$)
Model statements for module: Assign 208
989$      ASSIGN:      A Sentido=V DROT(A IndiRota, A Rota):
                                A Trecho=V ROT(A IndiRota, A Rota):NEXT(985$)
Model statements for module: Decide 168
985$      BRANCH,      1:
                                If,V Trecho ( A Trecho , 2 ) == A PtOrigem,3001$,Yes:
                                Else,3002$,Yes
3001$     ASSIGN:      3 Patio de origem do trem esta neste trecho ?.NumberOut
True=
                                3 Patio de origem do trem esta neste trecho ?.NumberOut
True + 1:NEXT(990$)
3002$     ASSIGN:      3 Patio de origem do trem esta neste trecho ?.NumberOut
False=

```

```

3 Patio de origem do trem esta neste trecho ?.NumberOut
False + 1:NEXT(984$)
  Model statements for module: Decide 170
990$      BRANCH,      1:
          If,V Ocpt ( A PtOrigem , 1+ (A Sentido==0) ) ==
0,3003$,Yes:
          Else,3004$,Yes
3003$      ASSIGN:      3 Patio no sentido correto esta livre ?.NumberOut True=
          3 Patio no sentido correto esta livre ?.NumberOut True +
1:NEXT(1052$)
3004$      ASSIGN:      3 Patio no sentido correto esta livre ?.NumberOut False=
          3 Patio no sentido correto esta livre ?.NumberOut False
+ 1:NEXT(992$)
  1 = CRESCENTE
  Próximo Pátio: Indirota+1
  Trecho usado: Indirota
  0 = DECRESCENTE
  Próximo Pátio: Indirota+1
  Trecho usado: Indirota+1
  Model statements for module: Assign 368
1052$      ASSIGN:      A TmpSentido=V DROT( A Indirota, A Rota):
          A TmpTrecho=V ROT(A Indirota + (A TmpSentido==0), A
Rota):
          A Patio=V Trecho (V ROT(A Indirota+1, A Rota) , 2 ):
          A Direto=1:NEXT(991$)
  Model statements for module: Assign 209
991$      ASSIGN:      A PtAtual=V Trecho ( A Trecho , 2 ):
          A Tmp=8123:NEXT(1019$)
  Model statements for module: Assign 266
1019$      ASSIGN:      V LocosNow(1)=V LocosNow(1) + E FrotasT (A Ntrem, 1):
          V LocosNow(2)=V LocosNow(2) + E FrotasT (A Ntrem, 2):
          V LocosNow(3)=V LocosNow(3) + E FrotasT (A Ntrem, 3):
          V LocosNow(4)=V LocosNow(4) + E FrotasT (A Ntrem, 4):
          V LocosNow(5)=V LocosNow(5) + E FrotasT (A Ntrem,
5):NEXT(1117$)
  Model statements for module: Decide 448
1117$      BRANCH,      1:
          If,TNOW > (V Warmup * 1440),3005$,Yes:
          Else,3006$,Yes
3005$      ASSIGN:      Warmup do log de eventos 11.NumberOut True=Warmup do log
de eventos 11.NumberOut True + 1
          :NEXT(1157$)
3006$      ASSIGN:      Warmup do log de eventos 11.NumberOut False=Warmup do
log de eventos 11.NumberOut False + 1
          :NEXT(1107$)
  Model statements for module: Decide 637
1157$      BRANCH,      1:
          If,V Gravar em texto == 1,3007$,Yes:
          Else,3008$,Yes
3007$      ASSIGN:      Decide 637.NumberOut True=Decide 637.NumberOut True +
1:NEXT(1171$)
3008$      ASSIGN:      Decide 637.NumberOut False=Decide 637.NumberOut False +
1:NEXT(1024$)
  Especificação do arquivo de LOG de eventos:
  1 - Dia do evento
  2- Identificador único do trem (A IDTrem)
  3 - Número do trem (A NTrem)
  4 - Tipo do trem (A Tipo)
  5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
  6 - Pátio (A PtAtual)
  7 - Momento da saída do trem (TNOW)
  8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
  Model statements for module: ReadWrite 168

```

```

1171$      WRITE,          Log de Eventos:
                          Dia,
                          A IDtrem,
                          A Ntrem,
                          A Tipo,
                          marca,
                          A PtAtual,
                          tnow,
                          6,
                          A Peso,
                          A Dist,
                          E FrotasT (A Ntrem, 1),
                          E FrotasT (A Ntrem, 2),
                          E FrotasT (A Ntrem, 3),
                          E FrotasT (A Ntrem, 4),
                          E FrotasT (A Ntrem, 5),
                          A Estado,
                          A Produto,
                          A Cliente,
                          A Tcarga,
                          A PtDestino,
                          A PtZero:NEXT(1107$)
      Model statements for module: Assign 475
1107$      ASSIGN:        V TUDia=V TUDia + A Peso:NEXT(993$)
      Model statements for module: Route 12
993$      ROUTE:         0.0000000000000000,ESTA ( A PtAtual,1 + ( A Sentido == 0)
)
      Especificação do arquivo de LOG de eventos:
      1 - Dia do evento
      2- Identificador único do trem (A IDTrem)
      3 - Número do trem (A NTrem)
      4 - Tipo do trem (A Tipo)
      5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
      6 - Pátio (A PtAtual)
      7 - Momento da saída do trem (TNOW)
      8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
      Model statements for module: ReadWrite 51
1024$     WRITE,          Log de MovimentacaoMDB,RECORDSET(LogdeEvento):
                          Dia,
                          A IDtrem,
                          A Ntrem,
                          A Tipo,
                          marca,
                          A PtAtual,
                          tnow,
                          6,
                          A Peso,
                          A Dist,
                          E FrotasT (A Ntrem, 1),
                          E FrotasT (A Ntrem, 2),
                          E FrotasT (A Ntrem, 3),
                          E FrotasT (A Ntrem, 4),
                          E FrotasT (A Ntrem, 5),
                          A Estado,
                          A Produto,
                          A Cliente,
                          A Tcarga,
                          A PtDestino,
                          A PtZero:NEXT(1107$)
      Model statements for module: Delay 25
992$      DELAY:         SecondsToBaseTime(norm(13,5)),,Other:NEXT(1135$)
      Model statements for module: Decide 474
1135$     BRANCH,       1:

```

```

                                If, checar >= 0 .AND. checar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),3009$,Yes:
                                Else,3010$,Yes
3009$          ASSIGN:          Decide 474.NumberOut True=Decide 474.NumberOut True +
1:NEXT(990$)
3010$          ASSIGN:          Decide 474.NumberOut False=Decide 474.NumberOut False +
1:NEXT(990$)
    Model statements for module: Assign 207
984$          ASSIGN:          A IndiRota=A IndiRota + 1:NEXT(986$)
    Model statements for module: Decide 169
986$          BRANCH,          1:
                                If,V ROT(A IndiRota, A Rota) == 0,3011$,Yes:
                                Else,3012$,Yes
3011$          ASSIGN:          3 Acabaram os trechos do roteiro sem achar a estacao
?.NumberOut True=
                                3 Acabaram os trechos do roteiro sem achar a estacao
?.NumberOut True + 1:NEXT(987$)
3012$          ASSIGN:          3 Acabaram os trechos do roteiro sem achar a estacao
?.NumberOut False=
                                3 Acabaram os trechos do roteiro sem achar a estacao
?.NumberOut False + 1:NEXT(989$)
    Model statements for module: ReadWrite 25
987$          WRITE,          Log de Erros,
                                "%1.8g. ERR2 Erro na Rota %1.8g. Nao encontrado patio
numero %1.8g de origem do trem. Trem abortado.\n":
                                TNOW,
                                A Rota,
                                A PtOrigem:NEXT(988$)
    Model statements for module: Dispose 34
988$          ASSIGN:          Dispose 34.NumberOut=Dispose 34.NumberOut + 1
3013$          DISPOSE:        Yes
    Model statements for module: ReadWrite 26
996$          WRITE,          Log de Erros,
                                "%1.8g. ERR7 Erro no trem tipo %1.8g e ID %1.8g.
Escolhida rota zerada em %1.8g. Temp1 %1.8g. A Codigo %1.8g. Trem abortado.\n":
                                TNOW,
                                A Tipo,
                                A IDtrem,
                                A PtOrigem,
                                Temp1,
                                A Codigo:NEXT(997$)
    Model statements for module: Dispose 35
997$          ASSIGN:          Dispose 35.NumberOut=Dispose 35.NumberOut + 1
3014$          DISPOSE:        Yes
    Model statements for module: Assign 486
1118$         ASSIGN:          V LocosNow(1)=V LocosNow(1) + E FrotasT (A Ntrem, 1):
                                V LocosNow(2)=V LocosNow(2) + E FrotasT (A Ntrem, 2):
                                V LocosNow(3)=V LocosNow(3) + E FrotasT (A Ntrem, 3):
                                V LocosNow(4)=V LocosNow(4) + E FrotasT (A Ntrem, 4):
                                V LocosNow(5)=V LocosNow(5) + E FrotasT (A Ntrem,
5):NEXT(1054$)
    Model statements for module: Route 30
1054$         ROUTE:          0.0000000000000000,Direcionamento
    Model statements for module: Assign 213
1001$         ASSIGN:          Temp1=2:
                                A Tipo=2:NEXT(998$)
    Model statements for module: Assign 214
1002$         ASSIGN:          Temp1=3:
                                A Tipo=2:NEXT(998$)
    Model statements for module: Assign 215
1003$         ASSIGN:          Temp1=4:NEXT(998$)
    Model statements for module: Assign 216
1004$         ASSIGN:          Temp1=6:NEXT(998$)
    Model statements for module: Assign 217
1005$         ASSIGN:          Temp1=7:NEXT(998$)

```

```

Model statements for module: Assign 371
1055$      ASSIGN:          Temp1=5:NEXT(998$)
Model statements for module: Assign 527
1126$      ASSIGN:          Temp1=8:NEXT(998$)
Model statements for module: Assign 530
1127$      ASSIGN:          Temp1=9:NEXT(998$)
Model statements for module: Assign 531
1128$      ASSIGN:          Temp1=10:NEXT(998$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 40
1014$      WRITE,          Log de MovimentacaoMDB,RECORDSET(LogdeEvento):
                          Dia,
                          A IDtrem,
                          A Ntrem,
                          A Tipo,
                          marca,
                          A PtAtual,
                          tnow,
                          2,
                          A Peso,
                          A Dist,
                          E FrotasT (A Ntrem, 1),
                          E FrotasT (A Ntrem, 2),
                          E FrotasT (A Ntrem, 3),
                          E FrotasT (A Ntrem, 4),
                          E FrotasT (A Ntrem, 5),
                          A Estado,
                          A Produto,
                          A Cliente,
                          A Tcarga,
                          A PtAtual,
                          A PtZero:NEXT(999$)
Model statements for module: Assign 349
1048$      ASSIGN:          marca=tnow:
                          A Estado=0:
                          A Tipo=2:
                          A Peso=0:NEXT(983$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 54
1026$      WRITE,          Log de MovimentacaoMDB,RECORDSET(LogdeEvento):
                          Dia,
                          A IDtrem,
                          A Ntrem,
                          A Tipo,
                          marca,

```

```

A PtAtual,
tnow,
9,
tnow - marca,
A Dist,
E FrotasT (A Ntrem, 1),
E FrotasT (A Ntrem, 2),
E FrotasT (A Ntrem, 3),
E FrotasT (A Ntrem, 4),
E FrotasT (A Ntrem, 5),
A Estado,
A Produto,
A Cliente,
A Tcarga,
A PtAtual,
A PtZero:NEXT(1087$)
Model statements for module: Delay 42
1094$      DELAY:      SecondsToBaseTime(norm(15,100)),,Other:NEXT(1129$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 46
1022$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeEvento):
Dia,
A IDtrem,
A Ntrem,
A Tipo,
marca,
A PtAtual,
tnow,
4,
A Peso,
A Dist,
E FrotasT (A Ntrem, 1),
E FrotasT (A Ntrem, 2),
E FrotasT (A Ntrem, 3),
E FrotasT (A Ntrem, 4),
E FrotasT (A Ntrem, 5),
A Estado,
A Produto,
A Cliente,
A Tcarga,
A PtAtual,
A PtZero:NEXT(1010$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 52
1025$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeCiclos):

```

```

Dia,
A IDtrem,
A Ntrem,
A Tipo,
A PtOrigem,
A PtAtual,
tnow - MarcaCiclo,
A Produto,
A Estado,
A Cliente,
A PtZero,
2,
A Tcarga:NEXT(1156$)
Model statements for module: Decide 174
1009$ BRANCH, 1:
If,A PtAtual<>151.AND.A PtAtual<>7.AND.A PtAtual<>148
.OR. A Etapa <> 3,3015$,Yes:
Else,3016$,Yes
3015$ ASSIGN: Chegou a um ponto de carga e nao ponto zero ?.NumberOut
True=
Chegou a um ponto de carga e nao ponto zero ?.NumberOut
True + 1:NEXT(1021$)
3016$ ASSIGN: Chegou a um ponto de carga e nao ponto zero ?.NumberOut
False=
Chegou a um ponto de carga e nao ponto zero ?.NumberOut
False + 1:NEXT(1040$)
Model statements for module: Assign 268
1021$ ASSIGN: V LocosNow(1)=V LocosNow(1) - E FrotasT (A Ntrem, 1):
V LocosNow(2)=V LocosNow(2) - E FrotasT (A Ntrem, 2):
V LocosNow(3)=V LocosNow(3) - E FrotasT (A Ntrem, 3):
V LocosNow(4)=V LocosNow(4) - E FrotasT (A Ntrem, 4):
V LocosNow(5)=V LocosNow(5) - E FrotasT (A Ntrem, 5):
A Dist=0:
marca=TNOW:
A Entrada=0:NEXT(1155$)
Model statements for module: Assign 587
1155$ ASSIGN: V Trens na Malha(2)=V Trens na Malha(2) - 1:NEXT(1103$)
Model statements for module: Decide 417
1103$ BRANCH, 1:
If,E Patios (A PtAtual,10)==12,3017$,Yes:
Else,3018$,Yes
3017$ ASSIGN: Trem saindo do trecho de FVR para carregar ?.NumberOut
True=
Trem saindo do trecho de FVR para carregar ?.NumberOut
True + 1:NEXT(1104$)
3018$ ASSIGN: Trem saindo do trecho de FVR para carregar ?.NumberOut
False=
Trem saindo do trecho de FVR para carregar ?.NumberOut
False + 1:NEXT(1110$)
Model statements for module: Assign 465
1104$ ASSIGN: V ContagemFVR=V ContagemFVR - 1:NEXT(1110$)
Model statements for module: Decide 440
1110$ BRANCH, 1:
If, TNOW > (V Warmup * 1440),3019$,Yes:
Else,3020$,Yes
3019$ ASSIGN: Warmup do log de eventos e ciclo etapa 1.NumberOut True=
Warmup do log de eventos e ciclo etapa 1.NumberOut True
+ 1:NEXT(1162$)
3020$ ASSIGN: Warmup do log de eventos e ciclo etapa 1.NumberOut
False=
Warmup do log de eventos e ciclo etapa 1.NumberOut False
+ 1:NEXT(1012$)
Model statements for module: Decide 642
1162$ BRANCH, 1:
If,V Gravar em texto == 1,3021$,Yes:

```



```

Else,3022$,Yes
3021$      ASSIGN:      Decide 642.NumberOut True=Decide 642.NumberOut True +
1:NEXT(1173$)
3022$      ASSIGN:      Decide 642.NumberOut False=Decide 642.NumberOut False +
1:NEXT(1078$)
    Especificação do arquivo de LOG de eventos:
    1 - Dia do evento
    2- Identificador único do trem (A IDTrem)
    3 - Número do trem (A NTrem)
    4 - Tipo do trem (A Tipo)
    5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
    6 - Pátio (A PtAtual)
    7 - Momento da saída do trem (TNOW)
    8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
    Model statements for module: ReadWrite 170
1173$      WRITE,      Log de Ciclos:
                        Dia,
                        A IDtrem,
                        A Ntrem,
                        A Tipo,
                        A PtOrigem,
                        A PtAtual,
                        tnow - MarcaCiclo,
                        A Produto,
                        A Estado,
                        A Cliente,
                        A PtZero,
                        1,
                        A Tcarga:NEXT(1172$)
    Especificação do arquivo de LOG de eventos:
    1 - Dia do evento
    2- Identificador único do trem (A IDTrem)
    3 - Número do trem (A NTrem)
    4 - Tipo do trem (A Tipo)
    5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
    6 - Pátio (A PtAtual)
    7 - Momento da saída do trem (TNOW)
    8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
    Model statements for module: ReadWrite 169
1172$      WRITE,      Log de Eventos:
                        Dia,
                        A IDtrem,
                        A Ntrem,
                        A Tipo,
                        marca,
                        A PtAtual,
                        tnow,
                        4,
                        A Peso,
                        A Dist,
                        E FrotasT (A Ntrem, 1),
                        E FrotasT (A Ntrem, 2),
                        E FrotasT (A Ntrem, 3),
                        E FrotasT (A Ntrem, 4),
                        E FrotasT (A Ntrem, 5),
                        A Estado,
                        A Produto,
                        A Cliente,
                        A Tcarga,
                        A PtAtual,
                        A PtZero:NEXT(1012$)

```

```

Model statements for module: Decide 196
1012$      BRANCH,      1:
                If,A ColPatio == 1,3023$,Yes:
                Else,3024$,Yes
3023$      ASSIGN:      Trem reservou linha de espera para carga?.NumberOut
True=
                Trem reservou linha de espera para carga?.NumberOut True
+ 1:NEXT(1083$)
3024$      ASSIGN:      Trem reservou linha de espera para carga?.NumberOut
False=
                Trem reservou linha de espera para carga?.NumberOut
False + 1:NEXT(1086$)
Model statements for module: Assign 431
1083$      ASSIGN:      Sorteio=Unif(12000,18000)+Unif(1,350):NEXT(1084$)
Model statements for module: Separate 54
1084$      DUPLICATE,   100 - 50:
                1,3027$,50:NEXT(3026$)
3026$      ASSIGN:      Separate 54.NumberOut Orig=Separate 54.NumberOut Orig +
1:NEXT(1085$)
3027$      ASSIGN:      Separate 54.NumberOut Dup=Separate 54.NumberOut Dup +
1:NEXT(1130$)
Model statements for module: Assign 432
1085$      ASSIGN:      Picture=Picture.espcarga:NEXT(1036$)
Model statements for module: Route 17
1036$      ROUTE:      0.0000000000000000,Station Anima Evento
Model statements for module: Assign 535
1130$      ASSIGN:      Temp2=A Produto * (A Produto <> 2) + 1 * (A Produto ==
2):NEXT(1095$)
Model statements for module: Decide 379
1095$      BRANCH,      1:
                If,
                V StatusPt (A PtAtual, 2) > 0 .AND. V StatusPtProd (A
PtAtual, Temp2) < E PtProd (A PtAtual, Temp2),
                3028$,Yes:
                Else,3029$,Yes
3028$      ASSIGN:      Linha de carga ficou livre ?.NumberOut True=Linha de
carga ficou livre ?.NumberOut True + 1
                :NEXT(1013$)
3029$      ASSIGN:      Linha de carga ficou livre ?.NumberOut False=Linha de
carga ficou livre ?.NumberOut False + 1
                :NEXT(1096$)
Model statements for module: Assign 242
1013$      ASSIGN:      V StatusPtProd (A PtAtual, Temp2)=V StatusPtProd (A
PtAtual, Temp2) + 1:
                V StatusPt (A PtAtual, 2)=V StatusPt (A PtAtual, 2) - 1:
                V StatusPt (A PtAtual, 1)=V StatusPt (A PtAtual, 1) +
1:
                A ColPatio=2:
                A Estado=1:
                A LinhaOcupada=0:NEXT(1090$)
Model statements for module: Signal 27
1090$      SIGNAL:      sorteio,1:NEXT(1111$)
Model statements for module: Decide 441
1111$      BRANCH,      1:
                If,TNOW > (V Warmup * 1440),3030$,Yes:
                Else,3031$,Yes
3030$      ASSIGN:      Warmup do log de eventos 7.NumberOut True=Warmup do log
de eventos 7.NumberOut True + 1:NEXT(1163$)
3031$      ASSIGN:      Warmup do log de eventos 7.NumberOut False=Warmup do log
de eventos 7.NumberOut False + 1
                :NEXT(1086$)
Model statements for module: Decide 643
1163$      BRANCH,      1:
                If,V Gravar em texto == 1,3032$,Yes:
                Else,3033$,Yes

```

```

3032$      ASSIGN:      Decide 643.NumberOut True=Decide 643.NumberOut True +
1:NEXT(1174$)
3033$      ASSIGN:      Decide 643.NumberOut False=Decide 643.NumberOut False +
1:NEXT(1027$)
    Especificação do arquivo de LOG de eventos:
    1 - Dia do evento
    2- Identificador único do trem (A IDTrem)
    3 - Número do trem (A NTrem)
    4 - Tipo do trem (A Tipo)
    5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
    6 - Pátio (A PtAtual)
    7 - Momento da saída do trem (TNOW)
    8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
    Model statements for module:  ReadWrite 171
1174$      WRITE,      Log de Eventos:
                        Dia,
                        A IDtrem,
                        A Ntrem,
                        A Tipo,
                        marca,
                        A PtAtual,
                        tnow,
                        8,
                        tnow - marca,
                        A Dist,
                        E FrotasT (A Ntrem, 1),
                        E FrotasT (A Ntrem, 2),
                        E FrotasT (A Ntrem, 3),
                        E FrotasT (A Ntrem, 4),
                        E FrotasT (A Ntrem, 5),
                        A Estado,
                        A Produto,
                        A Cliente,
                        A Tcarga,
                        A PtAtual,
                        A PtZero:NEXT(1086$)
    Model statements for module:  Assign 433
1086$      ASSIGN:      Sorteio=Unif(12000,18000)+Unif(1,350):NEXT(1034$)
    Model statements for module:  Separate 25
1034$      DUPLICATE,   100 - 50:
                        1,3036$,50:NEXT(3035$)
3035$      ASSIGN:      Separate 25.NumberOut Orig=Separate 25.NumberOut Orig +
1:NEXT(1035$)
3036$      ASSIGN:      Separate 25.NumberOut Dup=Separate 25.NumberOut Dup +
1:NEXT(1042$)
    Model statements for module:  Assign 292
1035$      ASSIGN:      Picture=Picture.carga:NEXT(1036$)
    Model statements for module:  Decide 230
1042$      BRANCH,     1:
                        If,A aTemp3 > 0,3037$,Yes:
                        Else,3038$,Yes
3037$      ASSIGN:      Trem veio de direcionamento ?.NumberOut True=Trem veio
de direcionamento ?.NumberOut True + 1
                        :NEXT(1007$)
3038$      ASSIGN:      Trem veio de direcionamento ?.NumberOut False=Trem veio
de direcionamento ?.NumberOut False + 1
                        :NEXT(1043$)
    Model statements for module:  Assign 218
1007$      ASSIGN:      marca=tnow:
                        A Estado=1:
                        V TD(A aTemp3)=V TD(A aTemp3) - (1/E Patios (A PtAtual,
2)):

```

```

A Peso=E Tabelas (A NTrem, 3) * E PesosT(A
NTrem,1):NEXT(1008$)
  Model statements for module: Delay 26
1008$      DELAY:
          E ExtraPcg (A EscolhaTMP, 3)/E PtProd(A PtAtual,(A
Produto * (A Produto <> 2) + 1 * (A Produto == 2))),,
          Other:NEXT(1049$)
  Model statements for module: Decide 299
1049$      BRANCH,      1:
          If,A Tipo == 3,3039$,Yes:
          Else,3040$,Yes
3039$      ASSIGN:      Tabela carregado nao MFE ?.NumberOut True=Tabela
carregado nao MFE ?.NumberOut True + 1:NEXT(1112$)
3040$      ASSIGN:      Tabela carregado nao MFE ?.NumberOut False=Tabela
carregado nao MFE ?.NumberOut False + 1
          :NEXT(1050$)
  Model statements for module: Decide 442
1112$      BRANCH,      1:
          If,TNOW > (V Warmup * 1440),3041$,Yes:
          Else,3042$,Yes
3041$      ASSIGN:      Warmup do log de eventos 8.NumberOut True=Warmup do log
de eventos 8.NumberOut True + 1:NEXT(1164$)
3042$      ASSIGN:      Warmup do log de eventos 8.NumberOut False=Warmup do log
de eventos 8.NumberOut False + 1
          :NEXT(1037$)
  Model statements for module: Decide 644
1164$      BRANCH,      1:
          If,V Gravar em texto == 1,3043$,Yes:
          Else,3044$,Yes
3043$      ASSIGN:      Decide 644.NumberOut True=Decide 644.NumberOut True +
1:NEXT(1175$)
3044$      ASSIGN:      Decide 644.NumberOut False=Decide 644.NumberOut False +
1:NEXT(1015$)
  Especificação do arquivo de LOG de eventos:
  1 - Dia do evento
  2- Identificador único do trem (A IDTrem)
  3 - Número do trem (A NTrem)
  4 - Tipo do trem (A Tipo)
  5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
  6 - Pátio (A PtAtual)
  7 - Momento da saída do trem (TNOW)
  8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
  Model statements for module: ReadWrite 172
1175$      WRITE,      Log de Eventos:
          Dia,
          A IDtrem,
          A Ntrem,
          A Tipo,
          marca,
          A PtAtual,
          tnow,
          1,
          A Peso,
          A Dist,
          E FrotasT (A Ntrem, 1),
          E FrotasT (A Ntrem, 2),
          E FrotasT (A Ntrem, 3),
          E FrotasT (A Ntrem, 4),
          E FrotasT (A Ntrem, 5),
          A Estado,
          A Produto,
          A Cliente,
          A Tcarga,

```

```

A PtAtual,
A PtZero:NEXT(1037$)
Model statements for module: Signal 24
1037$ SIGNAL: sorteio,1:NEXT(1056$)
Model statements for module: Decide 323
1056$ BRANCH, 1:
If,A ColPatio <> 0,3045$,Yes:
Else,3046$,Yes
3045$ ASSIGN: Decide 323.NumberOut True=Decide 323.NumberOut True +
1:NEXT(1041$)
3046$ ASSIGN: Decide 323.NumberOut False=Decide 323.NumberOut False +
1:NEXT(1057$)
Model statements for module: Assign 304
1041$ ASSIGN: Temp2=A Produto * (A Produto <> 2) + 1 * (A Produto ==
2):
V StatusPtProd (A PtAtual, Temp2)=V StatusPtProd (A
PtAtual, Temp2) - 1:
V StatusPt (A PtAtual, A ColPatio)=V StatusPt (A
PtAtual, A ColPatio) + 1:
A Etapa=2:
Entity.Picture=Figuras Carreg(A Produto):
A ColPatio=0:NEXT(1121$)
Model statements for module: Decide 470
1121$ BRANCH, 1:
If,A Grupo <= 15,3047$,Yes:
Else,3048$,Yes
3047$ ASSIGN: Grupo entre os primeiros 15 ?.NumberOut True=Grupo entre
os primeiros 15 ?.NumberOut True + 1
:NEXT(1120$)
3048$ ASSIGN: Grupo entre os primeiros 15 ?.NumberOut False=Grupo
entre os primeiros 15 ?.NumberOut False + 1
:NEXT(1122$)
Model statements for module: Assign 524
1120$ ASSIGN: A Tmp=A EscolhaTMP - ((A Grupo-1) * 12):NEXT(1060$)
Model statements for module: Assign 399
1060$ ASSIGN: A Estado=1:
A IndiRota=1:
A Peso=E Tabelas (A NTrem, 3) * E PesosT(A NTrem,1):
A Codigo=A EscolhaCDG:
MarcaCiclo=tnow:
A Etapa=2:NEXT(1061$)
Model statements for module: Decide 342
1061$ BRANCH, 1:
If,A Codigo == 1,1062$,Yes:
If,A Codigo == 2,1063$,Yes:
If,A Codigo == 3,1064$,Yes:
If,A Codigo == 4,1065$,Yes:
If,A Codigo == 5,1066$,Yes:
If,A Codigo == 6,1067$,Yes:
If,A Codigo == 7,1068$,Yes:
If,A Codigo == 8,1069$,Yes:
If,A Codigo == 9,1070$,Yes:
If,A Codigo == 10,1071$,Yes:
If,A Codigo == 11,1072$,Yes:
If,A Codigo == 12,1073$,Yes:
If,A Codigo == 13,1074$,Yes:
If,A Codigo == 14,1075$,Yes:
If,A Codigo == 15,1119$,Yes:
Else,1123$,Yes
Model statements for module: ReadWrite 94
1123$ WRITE, Log de Erros,
"%1.8g. ERR14 Erro no trem tipo %1.8g e numero %1.8g.
Escolhida rota zerada em %1.8g. Pto Zero %1.8g. Com destino a %1.8g. A Tmp %1.8g.
Trem abortado.\n":
TNOW,

```

```

A Tipo,
A Ntrem,
A PtOrigem,
A PtZero,
A PtDestino,
A Tmp:NEXT(1124$)
Model statements for module: Hold 77
1124$    QUEUE,      Hold 77.Queue:DETACH
Model statements for module: Assign 400
1062$    ASSIGN:     A Codigo=1:
          A Rota=E RotPcg ( A Tmp, 1):
          A PtDestino=44:
          A PtOrigem=A PtZero:
          A Cliente=1:NEXT(1076$)
Model statements for module: Decide 343
1076$    BRANCH,    1:
          If,A PtZero <> 151 .AND. A PtZero <> 7 .AND. A PtZero <>
148,3051$,Yes:
          Else,3052$,Yes
3051$    ASSIGN:     Trem tabela nao MFE ?.NumberOut True=Trem tabela nao MFE
          ?.NumberOut True + 1:NEXT(1131$)
3052$    ASSIGN:     Trem tabela nao MFE ?.NumberOut False=Trem tabela nao
          MFE ?.NumberOut False + 1:NEXT(1077$)
Model statements for module: Decide 473
1131$    BRANCH,    1:
          If,chechar >= 0 .AND. chechar < TNOW.OR.(A
IDTrem==Rastreio.AND.Rastreio2<TNOW),3053$,Yes:
          Else,3054$,Yes
3053$    ASSIGN:     Ponto de checagem 32.NumberOut True=Ponto de checagem
          32.NumberOut True + 1:NEXT(1132$)
3054$    ASSIGN:     Ponto de checagem 32.NumberOut False=Ponto de checagem
          32.NumberOut False + 1:NEXT(1153$)
Model statements for module: Separate 63
1132$    DUPLICATE, 100 - 50:
          1,3057$,50:NEXT(3056$)
3056$    ASSIGN:     Separate 63.NumberOut Orig=Separate 63.NumberOut Orig +
          1:NEXT(1133$)
3057$    ASSIGN:     Separate 63.NumberOut Dup=Separate 63.NumberOut Dup +
          1:NEXT(1153$)
Model statements for module: Hold 78
1133$    QUEUE,      Hold 78.Queue:DETACH
Model statements for module: Assign 585
1153$    ASSIGN:     V Trens na Malha(1)=V Trens na Malha(1) + 1:
          A Prioridade=1 + (2*(A Tipo==3)):NEXT(995$)
Model statements for module: Assign 415
1077$    ASSIGN:     A PtOrigem=A PtAtual:NEXT(1131$)
Model statements for module: Assign 401
1063$    ASSIGN:     A Codigo=2:
          A Rota=E RotPcg ( A Tmp, 2):
          A PtDestino=41:
          A PtOrigem=A PtZero:
          A Cliente=2:NEXT(1076$)
Model statements for module: Assign 402
1064$    ASSIGN:     A Codigo=3:
          A Rota=E RotPcg ( A Tmp, 3):
          A PtDestino=48:
          A PtOrigem=A PtZero:
          A Cliente=3 + 5 * (A Tipo == 3):NEXT(1076$)
Model statements for module: Assign 403
1065$    ASSIGN:     A Codigo=4:
          A Rota=E RotPcg ( A Tmp, 4):
          A PtDestino=148:
          A PtOrigem=A PtZero:
          A Cliente=4:NEXT(1076$)
Model statements for module: Assign 404

```

```

1066$      ASSIGN:      A Codigo=5:
                          A Rota=E RotPcg ( A Tmp, 5):
                          A PtDestino=148:
                          A PtOrigem=A PtZero:
                          A Cliente=5:NEXT(1076$)
      Model statements for module: Assign 405
1067$      ASSIGN:      A Codigo=6:
                          A Rota=E RotPcg ( A Tmp, 6):
                          A PtDestino=176:
                          A PtOrigem=A PtZero:
                          A Cliente=6:NEXT(1076$)
      Model statements for module: Assign 406
1068$      ASSIGN:      A Codigo=7:
                          A Rota=E RotPcg ( A Tmp, 7):
                          A PtDestino=79:
                          A PtOrigem=A PtZero:
                          A Cliente=7:NEXT(1076$)
      Model statements for module: Assign 407
1069$      ASSIGN:      A Codigo=8:
                          A Rota=E RotPcg ( A Tmp, 8):
                          A PtDestino=233:
                          A PtOrigem=A PtZero:
                          A Cliente=9:NEXT(1076$)
      Model statements for module: Assign 408
1070$      ASSIGN:      A Codigo=9:
                          A Rota=E RotPcg ( A Tmp, 9):
                          A PtDestino=351:
                          A PtOrigem=A PtZero:
                          A Cliente=13:NEXT(1076$)
      Model statements for module: Assign 409
1071$      ASSIGN:      A Codigo=10:
                          A Rota=E RotPcg ( A Tmp, 10):
                          A PtDestino=180:
                          A PtOrigem=A PtZero:
                          A Cliente=0:NEXT(1076$)
      Model statements for module: Assign 410
1072$      ASSIGN:      A Codigo=11:
                          A Rota=E RotPcg ( A Tmp, 11):
                          A PtDestino=81:
                          A PtOrigem=A PtZero:
                          A Cliente=0:NEXT(1076$)
      Model statements for module: Assign 411
1073$      ASSIGN:      A Codigo=12:
                          A Rota=E RotPcg ( A Tmp, 12):
                          A PtDestino=81:
                          A PtOrigem=A PtZero:
                          A Cliente=0:NEXT(1076$)
      Model statements for module: Assign 412
1074$      ASSIGN:      A Codigo=13:
                          A Rota=E RotPcg ( A Tmp, 12):
                          A PtDestino=173:
                          A PtOrigem=A PtZero:
                          A Cliente=0:NEXT(1076$)
      Model statements for module: Assign 413
1075$      ASSIGN:      A Codigo=14:
                          A Rota=E RotPcg ( A Tmp, 14):
                          A PtDestino=129:
                          A PtOrigem=A PtZero:
                          A Cliente=0:NEXT(1076$)
      Model statements for module: Assign 523
1119$      ASSIGN:      A Codigo=15:
                          A Rota=E RotPcg ( A Tmp, 15):
                          A PtDestino=179:
                          A PtOrigem=A PtZero:
                          A Cliente=12:NEXT(1076$)

```

```

Model statements for module: Assign 525
1122$      ASSIGN:      A Tmp=(A Grupo - 15) + 12:NEXT(1060$)
Model statements for module: Hold 57
1057$      QUEUE,      Hold 57.Queue:DETACH
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 43
1015$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeEvento):
Dia,
A IDtrem,
A Ntrem,
A Tipo,
marca,
A PtAtual,
tnow,
1,
A Peso,
A Dist,
E FrotasT (A Ntrem, 1),
E FrotasT (A Ntrem, 2),
E FrotasT (A Ntrem, 3),
E FrotasT (A Ntrem, 4),
E FrotasT (A Ntrem, 5),
A Estado,
A Produto,
A Cliente,
A Tcarga,
A PtAtual,
A PtZero:NEXT(1037$)
Model statements for module: Assign 350
1050$      ASSIGN:      A Tipo=1:NEXT(1112$)
Model statements for module: Assign 309
1043$      ASSIGN:      marca=tnow:
A Estado=1:
A Peso=E Tabelas (A NTrem, 3) * E PesosT(A
NTrem,1):NEXT(1008$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 55
1027$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeEvento):
Dia,
A IDtrem,
A Ntrem,
A Tipo,
marca,
A PtAtual,
tnow,

```



```

      8,
      tnow - marca,
      A Dist,
      E FrotasT (A Ntrem, 1),
      E FrotasT (A Ntrem, 2),
      E FrotasT (A Ntrem, 3),
      E FrotasT (A Ntrem, 4),
      E FrotasT (A Ntrem, 5),
      A Estado,
      A Produto,
      A Cliente,
      A Tcarga,
      A PtAtual,
      A PtZero:NEXT(1086$)
Model statements for module: Delay 43
1096$      DELAY:      SecondsToBaseTime(norm(15,100)),,Other:NEXT(1130$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 61
1078$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeCiclos):
      Dia,
      A IDtrem,
      A Ntrem,
      A Tipo,
      A PtOrigem,
      A PtAtual,
      tnow - MarcaCiclo,
      A Produto,
      A Estado,
      A Cliente,
      A PtZero,
      1,
      A Tcarga:NEXT(1023$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 47
1023$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeEvento):
      Dia,
      A IDtrem,
      A Ntrem,
      A Tipo,
      marca,
      A PtAtual,
      tnow,
      4,
      A Peso,
      A Dist,

```

```

E FrotasT (A Ntrem, 1),
E FrotasT (A Ntrem, 2),
E FrotasT (A Ntrem, 3),
E FrotasT (A Ntrem, 4),
E FrotasT (A Ntrem, 5),
A Estado,
A Produto,
A Cliente,
A Tcarga,
A PtAtual,
A PtZero:NEXT(1012$)
Model statements for module: Route 18
1040$ ROUTE: 0.000000000000000,Direcionamento
Model statements for module: Decide 173
1006$ BRANCH, 1:
If,A Estado == 1,3058$,Yes:
Else,3059$,Yes
3058$ ASSIGN: Este unitario esta carregado ?.NumberOut True=Este
unitario esta carregado ?.NumberOut True + 1
:NEXT(1108$)
3059$ ASSIGN: Este unitario esta carregado ?.NumberOut False=Este
unitario esta carregado ?.NumberOut False + 1
:NEXT(1021$)
Model statements for module: Decide 487
1136$ BRANCH, 1:
If,A SouHelper == 0,3060$,Yes:
Else,3061$,Yes
3060$ ASSIGN: Nao helper ?.NumberOut True=Nao helper ?.NumberOut True
+ 1:NEXT(1017$)
3061$ ASSIGN: Nao helper ?.NumberOut False=Nao helper ?.NumberOut
False + 1:NEXT(1137$)
Model statements for module: Decide 216
1017$ BRANCH, 1:
If,A Tipo == 4,3062$,Yes:
Else,3063$,Yes
3062$ ASSIGN: 2Trem grade ?.NumberOut True=2Trem grade ?.NumberOut
True + 1:NEXT(1016$)
3063$ ASSIGN: 2Trem grade ?.NumberOut False=2Trem grade ?.NumberOut
False + 1:NEXT(1018$)
Model statements for module: Assign 264
1016$ ASSIGN: V LocosNow(1)=V LocosNow(1) - E FrotasG (A Ntrem, 1):
V LocosNow(2)=V LocosNow(2) - E FrotasG (A Ntrem, 2):
V LocosNow(3)=V LocosNow(3) - E FrotasG (A Ntrem, 3):
V LocosNow(4)=V LocosNow(4) - E FrotasG (A Ntrem, 4):
V LocosNow(5)=V LocosNow(5) - E FrotasG (A Ntrem,
5):NEXT(1151$)
Model statements for module: Assign 583
1151$ ASSIGN: V Trens na Malha(4)=V Trens na Malha(4) - 1:NEXT(1116$)
Model statements for module: Decide 447
1116$ BRANCH, 1:
If,TNOW > (V Warmup * 1440),3064$,Yes:
Else,3065$,Yes
3064$ ASSIGN: Warmup do log de eventos trem grade.NumberOut True=
Warmup do log de eventos trem grade.NumberOut True +
1:NEXT(1165$)
3065$ ASSIGN: Warmup do log de eventos trem grade.NumberOut False=
Warmup do log de eventos trem grade.NumberOut False +
1:NEXT(1101$)
Model statements for module: Decide 645
1165$ BRANCH, 1:
If,V Gravar em texto == 1,3066$,Yes:
Else,3067$,Yes
3066$ ASSIGN: Decide 645.NumberOut True=Decide 645.NumberOut True +
1:NEXT(1176$)

```

```

3067$      ASSIGN:      Decide 645.NumberOut False=Decide 645.NumberOut False +
1:NEXT(1028$)
    Especificação do arquivo de LOG de eventos:
    1 - Dia do evento
    2- Identificador único do trem (A IDTrem)
    3 - Número do trem (A NTrem)
    4 - Tipo do trem (A Tipo)
    5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
    6 - Pátio (A PtAtual)
    7 - Momento da saída do trem (TNOW)
    8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
    Model statements for module:  ReadWrite 173
1176$      WRITE,      Log de Eventos:
                        Dia,
                        A IDtrem,
                        A Ntrem,
                        A Tipo,
                        marca,
                        A PtAtual,
                        tnow,
                        4,
                        A Peso,
                        A Dist,
                        E FrotasG (A Ntrem, 1),
                        E FrotasG (A Ntrem, 2),
                        E FrotasG (A Ntrem, 3),
                        E FrotasG (A Ntrem, 4),
                        E FrotasG (A Ntrem, 5),
                        A Estado,
                        A Produto,
                        A Cliente,
                        A PtOrigem,
                        A PtDestino,
                        0:NEXT(1178$)
    Especificação do arquivo de LOG de eventos:
    1 - Dia do evento
    2- Identificador único do trem (A IDTrem)
    3 - Número do trem (A NTrem)
    4 - Tipo do trem (A Tipo)
    5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
    6 - Pátio (A PtAtual)
    7 - Momento da saída do trem (TNOW)
    8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
    Model statements for module:  ReadWrite 175
1178$      WRITE,      Log de Ciclos:
                        Dia,
                        A IDtrem,
                        A Ntrem,
                        A Tipo,
                        A PtOrigem,
                        A PtAtual,
                        tnow - MarcaCiclo,
                        A Produto,
                        A Estado,
                        A Cliente,
                        0,
                        0,
                        0:NEXT(1101$)
    Model statements for module:  Decide 410
1101$      BRANCH,      1:
                        If,E Patios (A PtAtual,10)==12,3068$,Yes:

```

```

Else,3069$,Yes
3068$      ASSIGN:      Trem saindo do trecho de FVR ?.NumberOut True=Trem
saindo do trecho de FVR ?.NumberOut True + 1
: NEXT(1102$)
3069$      ASSIGN:      Trem saindo do trecho de FVR ?.NumberOut False=Trem
saindo do trecho de FVR ?.NumberOut False + 1
: NEXT(981$)
Model statements for module: Assign 459
1102$      ASSIGN:      V ContagemFVR=V ContagemFVR - 1: NEXT(981$)
Model statements for module: Dispose 33
981$      ASSIGN:      Dispose 33.NumberOut=Dispose 33.NumberOut + 1
3070$      DISPOSE:     Yes
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 57
1028$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeEvento):
Dia,
A IDtrem,
A Ntrem,
A Tipo,
marca,
A PtAtual,
tnow,
4,
A Peso,
A Dist,
E FrotasG (A Ntrem, 1),
E FrotasG (A Ntrem, 2),
E FrotasG (A Ntrem, 3),
E FrotasG (A Ntrem, 4),
E FrotasG (A Ntrem, 5),
A Estado,
A Produto,
A Cliente,
A PtOrigem,
A PtDestino,
0: NEXT(1079$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 62
1079$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeCiclos):
Dia,
A IDtrem,
A Ntrem,
A Tipo,
A PtOrigem,
A PtAtual,

```

```

        tnow - MarcaCiclo,
        A Produto,
        A Estado,
        A Cliente,
        0,
        0,
        0:NEXT(1101$)
    Model statements for module: Assign 265
1018$    ASSIGN:    V LocosNow(1)=V LocosNow(1) - E FrotasE (A Ntrem, 1):
        V LocosNow(2)=V LocosNow(2) - E FrotasE (A Ntrem, 2):
        V LocosNow(3)=V LocosNow(3) - E FrotasE (A Ntrem, 3):
        V LocosNow(4)=V LocosNow(4) - E FrotasE (A Ntrem, 4):
        V LocosNow(5)=V LocosNow(5) - E FrotasE (A Ntrem,
5):NEXT(1152$)
    Model statements for module: Assign 584
1152$    ASSIGN:    V Trens na Malha(5)=V Trens na Malha(5) - 1:NEXT(1115$)
    Model statements for module: Decide 446
1115$    BRANCH,    1:
        If, TNOW > (V Warmup * 1440), 3071$, Yes:
        Else, 3072$, Yes
3071$    ASSIGN:    Warmup do log de eventos trem expresso.NumberOut True=
        Warmup do log de eventos trem expresso.NumberOut True +
1:NEXT(1166$)
3072$    ASSIGN:    Warmup do log de eventos trem expresso.NumberOut False=
        Warmup do log de eventos trem expresso.NumberOut False +
1:NEXT(1101$)
    Model statements for module: Decide 646
1166$    BRANCH,    1:
        If, V Gravar em texto == 1, 3073$, Yes:
        Else, 3074$, Yes
3073$    ASSIGN:    Decide 646.NumberOut True=Decide 646.NumberOut True +
1:NEXT(1177$)
3074$    ASSIGN:    Decide 646.NumberOut False=Decide 646.NumberOut False +
1:NEXT(1029$)
    Especificação do arquivo de LOG de eventos:
    1 - Dia do evento
    2- Identificador único do trem (A IDTrem)
    3 - Número do trem (A NTrem)
    4 - Tipo do trem (A Tipo)
    5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
    6 - Pátio (A PtAtual)
    7 - Momento da saída do trem (TNOW)
    8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
    Model statements for module: ReadWrite 174
1177$    WRITE,    Log de Eventos:
        Dia,
        A IDtrem,
        A Ntrem,
        A Tipo,
        marca,
        A PtAtual,
        tnow,
        4,
        A Peso,
        A Dist,
        E FrotasE (A Ntrem, 1),
        E FrotasE (A Ntrem, 2),
        E FrotasE (A Ntrem, 3),
        E FrotasE (A Ntrem, 4),
        E FrotasE (A Ntrem, 5),
        A Estado,
        A Produto,
        A Cliente,

```

```

A PtOrigem,
A PtDestino,
0:NEXT(1178$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 58
1029$ WRITE, Log de MovimentacaoMDB,RECORDSET(LogdeEvento):
Dia,
A IDtrem,
A Ntrem,
A Tipo,
marca,
A PtAtual,
tnow,
4,
A Peso,
A Dist,
E FrotasE (A Ntrem, 1),
E FrotasE (A Ntrem, 2),
E FrotasE (A Ntrem, 3),
E FrotasE (A Ntrem, 4),
E FrotasE (A Ntrem, 5),
A Estado,
A Produto,
A Cliente,
A PtOrigem,
A PtDestino,
0:NEXT(1079$)
Model statements for module: Assign 568
1137$ ASSIGN: V LocosNow(5)=V LocosNow(5) - A NLocos:NEXT(1139$)
Model statements for module: Decide 488
1139$ BRANCH, 1:
If,A RetornoH == 131,1138$,Yes:
If,A RetornoH==143 .OR. A RetornoH==144,1140$,Yes:
If,A RetornoH==1,1141$,Yes:
If,A RetornoH==7,1142$,Yes:
If,A RetornoH==30 .OR. A RetornoH==37,1143$,Yes:
Else,1144$,Yes
Model statements for module: Release 23
1144$ RELEASE: Helpers7,A NLocos:NEXT(1150$)
Model statements for module: Dispose 71
1150$ ASSIGN: Dispose 71.NumberOut=Dispose 71.NumberOut + 1
3077$ DISPOSE: Yes
Model statements for module: Release 17
1138$ RELEASE: Helpers1,A NLocos:NEXT(1145$)
Model statements for module: Dispose 66
1145$ ASSIGN: Dispose 66.NumberOut=Dispose 66.NumberOut + 1
3078$ DISPOSE: Yes
Model statements for module: Release 18
1140$ RELEASE: Helpers2,A NLocos:NEXT(1146$)
Model statements for module: Dispose 67
1146$ ASSIGN: Dispose 67.NumberOut=Dispose 67.NumberOut + 1
3079$ DISPOSE: Yes
Model statements for module: Release 19
1141$ RELEASE: Helpers3,A NLocos:NEXT(1147$)
Model statements for module: Dispose 68

```

```

1147$      ASSIGN:      Dispose 68.NumberOut=Dispose 68.NumberOut + 1
3080$      DISPOSE:      Yes
      Model statements for module:  Release 21
1142$      RELEASE:      Helpers5,A NLocos:NEXT(1148$)
      Model statements for module:  Dispose 69
1148$      ASSIGN:      Dispose 69.NumberOut=Dispose 69.NumberOut + 1
3081$      DISPOSE:      Yes
      Model statements for module:  Release 22
1143$      RELEASE:      Helpers6,A NLocos:NEXT(1149$)
      Model statements for module:  Dispose 70
1149$      ASSIGN:      Dispose 70.NumberOut=Dispose 70.NumberOut + 1
3082$      DISPOSE:      Yes
      Model statements for module:  Decide 234
1046$      BRANCH,      1:
                          If,E Verifica entroncamento .OR. A Direto==1,3083$,Yes:
                          Else,3084$,Yes
3083$      ASSIGN:      Unitario vazio vai entrar em um entroncamento
?.NumberOut True=
                          Unitario vazio vai entrar em um entroncamento
?.NumberOut True + 1:NEXT(1045$)
3084$      ASSIGN:      Unitario vazio vai entrar em um entroncamento
?.NumberOut False=
                          Unitario vazio vai entrar em um entroncamento
?.NumberOut False + 1:NEXT(1044$)
      Model statements for module:  Assign 317
1045$      ASSIGN:      V Conta bypass=V Conta bypass + 1:NEXT(1019$)
      Model statements for module:  Assign 316
1044$      ASSIGN:      V Ocpt ( A PtAtual , 1+ (A Sentido == 0))=1:NEXT(1019$)
      Model statements for module:  Decide 320
1051$      BRANCH,      1:
                          If,
                          (V Sentido (A TmpTrecho) == 2 .OR. V Sentido (A
TmpTrecho) == A TmpSentido .OR. V Trecho (A TmpTrecho,5)>1),
                          3085$,Yes:
                          Else,3086$,Yes
3085$      ASSIGN:      Sentido proximo trecho liberado ou igual ao deste
trem?.NumberOut True=
                          Sentido proximo trecho liberado ou igual ao deste
trem?.NumberOut True + 1:NEXT(991$)
3086$      ASSIGN:      Sentido proximo trecho liberado ou igual ao deste
trem?.NumberOut False=
                          Sentido proximo trecho liberado ou igual ao deste
trem?.NumberOut False + 1:NEXT(992$)
      Model statements for module:  Hold 92
1179$      QUEUE,      Hold 92.Queue:DETACH
rast      DELAY:      0.0,,Other:NEXT(1181$)
1181$      VBA:      2,vba:NEXT(1183$)
      Model statements for module:  Assign 252
1183$      ASSIGN:      Rastreio2=Rast:NEXT(1182$)
      Model statements for module:  Dispose 41
1182$      ASSIGN:      Dispose 41.NumberOut=Dispose 41.NumberOut + 1
3087$      DISPOSE:      Yes
      Model statements for module:  Create 14
3088$      CREATE,      1,DaysToBaseTime(0.0),Entity
1:DaysToBaseTime(1):NEXT(3089$)
3089$      ASSIGN:      Cria entidade que conta os dias.NumberOut=Cria entidade
que conta os dias.NumberOut + 1:NEXT(1184$)
      Model statements for module:  Assign 284
1184$      ASSIGN:      Dia=Dia + 1:NEXT(1199$)
      Model statements for module:  Decide 449
1199$      BRANCH,      1:
                          If,TNOW > ((V Warmup-1) * 1440),3092$,Yes:
                          Else,3093$,Yes
3092$      ASSIGN:      Warmup de todos os dias.NumberOut True=Warmup de todos
os dias.NumberOut True + 1:NEXT(1188$)

```

```

3093$      ASSIGN:      Warmup de todos os dias.NumberOut False=Warmup de todos
os dias.NumberOut False + 1:NEXT(1262$)
      Model statements for module:  Delay 35
1188$      DELAY:      1439.5,,Other:NEXT(1191$)
      Model statements for module:  Separate 62
1191$      DUPLICATE,    100 - 50:
                        1,3096$,50:NEXT(3095$)
3095$      ASSIGN:      Separate 62.NumberOut Orig=Separate 62.NumberOut Orig +
1:NEXT(1209$)
3096$      ASSIGN:      Separate 62.NumberOut Dup=Separate 62.NumberOut Dup +
1:NEXT(1262$)
      Model statements for module:  Assign 494
1209$      ASSIGN:      Indice=0:NEXT(1211$)
      Model statements for module:  Assign 496
1211$      ASSIGN:      Indice=Indice + 1:NEXT(1278$)
      Model statements for module:  Decide 623
1278$      BRANCH,      1:
                        If,V Gravar em texto == 1,3097$,Yes:
                        Else,3098$,Yes
3097$      ASSIGN:      Decide 623.NumberOut True=Decide 623.NumberOut True +
1:NEXT(1281$)
3098$      ASSIGN:      Decide 623.NumberOut False=Decide 623.NumberOut False +
1:NEXT(1187$)
      Especificação do arquivo de LOG de eventos:
      1 - Dia do evento
      2- Identificador único do trem (A IDTrem)
      3 - Número do trem (A NTrem)
      4 - Tipo do trem (A Tipo)
      5 - Momento de início da atividade (unitários) ou Horário do trem (A Horario -
grade/expresso)
      6 - Pátio (A PtAtual)
      7 - Momento da saída do trem (TNOW)
      8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
      Model statements for module:  ReadWrite 176
1281$      WRITE,      Log de Utilizacao:
                        Dia,
                        1,
                        Indice,
                        (V LocosAcm2(indice) / 1440) / V LocosFrota(indice),
                        0,
                        V LocosFrota(indice),
                        0:NEXT(1228$)
      Model statements for module:  Assign 506
1228$      ASSIGN:      V LocosAcm2(indice)=0:NEXT(1210$)
      Model statements for module:  Decide 455
1210$      BRANCH,      1:
                        If,Indice>=5,3099$,Yes:
                        Else,3100$,Yes
3099$      ASSIGN:      Verifica se acabaram as frotas.NumberOut True=Verifica
se acabaram as frotas.NumberOut True + 1
                        :NEXT(1213$)
3100$      ASSIGN:      Verifica se acabaram as frotas.NumberOut False=Verifica
se acabaram as frotas.NumberOut False + 1
                        :NEXT(1211$)
      Model statements for module:  Assign 497
1213$      ASSIGN:      Indice=0:NEXT(1215$)
      Model statements for module:  Assign 498
1215$      ASSIGN:      Indice=Indice + 1:NEXT(1277$)
      Model statements for module:  Decide 622
1277$      BRANCH,      1:
                        If,V Gravar em texto == 1,3101$,Yes:
                        Else,3102$,Yes
3101$      ASSIGN:      Decide 622.NumberOut True=Decide 622.NumberOut True +
1:NEXT(1282$)

```



```

3102$      ASSIGN:      Decide 622.NumberOut False=Decide 622.NumberOut False +
1:NEXT(1212$)
      Model statements for module:  ReadWrite 177
1282$      WRITE,      Log de Utilizacao:
                          Dia,
                          2,
                          V IDTC(Indice),
                          AcumulaPC(indice) / 1440,
                          0,
                          E Patios (V IDTC(indice), 2) * (E Patios (V
IDTC(indice), 2) > 0) + (E Patios (V IDTC(indice), 2) == 0),
                          0:NEXT(1225$)
      Model statements for module:  Assign 503
1225$      ASSIGN:      AcumulaPC(indice)=0:NEXT(1214$)
      Model statements for module:  Decide 456
1214$      BRANCH,      1:
                          If,Indice>=19,3103$,Yes:
                          Else,3104$,Yes
3103$      ASSIGN:      Verifica se acabaram os terminais de carga.NumberOut
True=
                          Verifica se acabaram os terminais de carga.NumberOut
True + 1:NEXT(1217$)
3104$      ASSIGN:      Verifica se acabaram os terminais de carga.NumberOut
False=
                          Verifica se acabaram os terminais de carga.NumberOut
False + 1:NEXT(1215$)
      Model statements for module:  Assign 499
1217$      ASSIGN:      Indice=0:NEXT(1219$)
      Model statements for module:  Assign 500
1219$      ASSIGN:      Indice=Indice + 1:NEXT(1263$)
      Model statements for module:  Decide 521
1263$      BRANCH,      1:
                          If,Indice == 10 .OR. Indice == 11,3105$,Yes:
                          Else,3106$,Yes
3105$      ASSIGN:      Desvia dos clientes 10 e 11 Grade e Expresso.NumberOut
True=
                          Desvia dos clientes 10 e 11 Grade e Expresso.NumberOut
True + 1:NEXT(1226$)
3106$      ASSIGN:      Desvia dos clientes 10 e 11 Grade e Expresso.NumberOut
False=
                          Desvia dos clientes 10 e 11 Grade e Expresso.NumberOut
False + 1:NEXT(1276$)
      Model statements for module:  Assign 504
1226$      ASSIGN:      AcumulaPD(indice)=0:NEXT(1218$)
      Model statements for module:  Decide 457
1218$      BRANCH,      1:
                          If,Indice>=13,3107$,Yes:
                          Else,3108$,Yes
3107$      ASSIGN:      Verifica se acabaram os terminais de Descarga.NumberOut
True=
                          Verifica se acabaram os terminais de Descarga.NumberOut
True + 1:NEXT(1258$)
3108$      ASSIGN:      Verifica se acabaram os terminais de Descarga.NumberOut
False=
                          Verifica se acabaram os terminais de Descarga.NumberOut
False + 1:NEXT(1219$)
      Model statements for module:  Assign 579
1258$      ASSIGN:      Indice=0:NEXT(1260$)
      Model statements for module:  Assign 580
1260$      ASSIGN:      Indice=Indice + 1:NEXT(1279$)
      Model statements for module:  Decide 624
1279$      BRANCH,      1:
                          If,V Gravar em texto == 1,3109$,Yes:
                          Else,3110$,Yes

```

```

3109$      ASSIGN:      Decide 624.NumberOut True=Decide 624.NumberOut True +
1:NEXT(1284$)
3110$      ASSIGN:      Decide 624.NumberOut False=Decide 624.NumberOut False +
1:NEXT(1257$)
      Model statements for module:  ReadWrite 179
1284$      WRITE,      Log de Utilizacao:
                          Dia,
                          6,
                          Indice,
                          V FilasH ( Indice,2 ),
                          0,
                          0,
                          0:NEXT(1261$)
      Model statements for module:  Assign 581
1261$      ASSIGN:      V FilasH(Indice,1)=0:
                          V FilasH(Indice,2)=0:NEXT(1259$)
      Model statements for module:  Decide 491
1259$      BRANCH,      1:
                          If,Indice>=8,3111$,Yes:
                          Else,3112$,Yes
3111$      ASSIGN:      Verifica se acabaram os patios de helper.NumberOut True=
                          Verifica se acabaram os patios de helper.NumberOut True
+ 1:NEXT(1264$)
3112$      ASSIGN:      Verifica se acabaram os patios de helper.NumberOut
False=
                          Verifica se acabaram os patios de helper.NumberOut False
+ 1:NEXT(1260$)
      Model statements for module:  Decide 575
1264$      BRANCH,      1:
                          If,Desliga Utilizacao Patios==0,3113$,Yes:
                          Else,3114$,Yes
3113$      ASSIGN:      Quer Estatistica?.NumberOut True=Quer
Estatistica?.NumberOut True + 1:NEXT(1221$)
3114$      ASSIGN:      Quer Estatistica?.NumberOut False=Quer
Estatistica?.NumberOut False + 1:NEXT(1298$)
      Model statements for module:  Assign 501
1221$      ASSIGN:      Indice=0:NEXT(1223$)
      Model statements for module:  Assign 502
1223$      ASSIGN:      Indice=Indice + 1:NEXT(1275$)
      Model statements for module:  Decide 620
1275$      BRANCH,      1:
                          If,V Gravar em texto == 1,3115$,Yes:
                          Else,3116$,Yes
3115$      ASSIGN:      Decide 620.NumberOut True=Decide 620.NumberOut True +
1:NEXT(1285$)
3116$      ASSIGN:      Decide 620.NumberOut False=Decide 620.NumberOut False +
1:NEXT(1220$)
      Model statements for module:  ReadWrite 180
1285$      WRITE,      Log de Utilizacao:
                          Dia,
                          4,
                          Indice,
                          AcumulaPP(Indice,1) / 1440,
                          1,
                          1,
                          1:NEXT(1286$)
      Model statements for module:  ReadWrite 181
1286$      WRITE,      Log de Utilizacao:
                          Dia,
                          4,
                          Indice,
                          AcumulaPP(Indice,4) / 1440,
                          2,
                          1,
                          1:NEXT(1287$)

```

```

Model statements for module: ReadWrite 182
1287$      WRITE,      Log de Utilizacao:
                          Dia,
                          4,
                          Indice,
                          AcumulaPP(Indice,2) / 1440,
                          1,
                          1,
                          2:NEXT(1288$)
Model statements for module: ReadWrite 183
1288$      WRITE,      Log de Utilizacao:
                          Dia,
                          4,
                          Indice,
                          AcumulaPP(Indice,5) / 1440,
                          2,
                          1,
                          2:NEXT(1289$)
Model statements for module: ReadWrite 184
1289$      WRITE,      Log de Utilizacao:
                          Dia,
                          4,
                          Indice,
                          AcumulaPP(Indice,3) / 1440,
                          1,
                          1,
                          3:NEXT(1290$)
Model statements for module: ReadWrite 185
1290$      WRITE,      Log de Utilizacao:
                          Dia,
                          4,
                          Indice,
                          AcumulaPP(Indice,6) / 1440,
                          2,
                          1,
                          3:NEXT(1291$)
Model statements for module: ReadWrite 186
1291$      WRITE,      Log de Utilizacao:
                          Dia,
                          5,
                          Indice,
                          AcumulaUPP(Indice,1),
                          1,
                          1,
                          1:NEXT(1292$)
Model statements for module: ReadWrite 187
1292$      WRITE,      Log de Utilizacao:
                          Dia,
                          5,
                          Indice,
                          AcumulaUPP(Indice,4),
                          2,
                          1,
                          1:NEXT(1293$)
Model statements for module: ReadWrite 188
1293$      WRITE,      Log de Utilizacao:
                          Dia,
                          5,
                          Indice,
                          AcumulaUPP(Indice,2),
                          1,
                          1,
                          2:NEXT(1294$)
Model statements for module: ReadWrite 189
1294$      WRITE,      Log de Utilizacao:

```

```

Dia,
5,
Indice,
AcumulaUPP (Indice,5),
2,
1,
2:NEXT(1295$)
Model statements for module: ReadWrite 190
1295$ WRITE, Log de Utilizacao:
Dia,
5,
Indice,
AcumulaUPP (Indice,3),
1,
1,
3:NEXT(1296$)
Model statements for module: ReadWrite 191
1296$ WRITE, Log de Utilizacao:
Dia,
5,
Indice,
AcumulaUPP (Indice,6),
2,
1,
3:NEXT(1247$)
Model statements for module: Assign 517
1247$ ASSIGN: AcumulaUPP (Indice,1)=0:
AcumulaUPP (Indice,2)=0:
AcumulaUPP (Indice,3)=0:
AcumulaUPP (Indice,4)=0:
AcumulaUPP (Indice,5)=0:
AcumulaUPP (Indice,6)=0:NEXT(1227$)
Model statements for module: Assign 505
1227$ ASSIGN: AcumulaPP (Indice,1)=0:
AcumulaPP (Indice,2)=0:
AcumulaPP (Indice,3)=0:
AcumulaPP (Indice,4)=0:
AcumulaPP (Indice,5)=0:
AcumulaPP (Indice,6)=0:NEXT(1222$)
Model statements for module: Decide 458
1222$ BRANCH, 1:
If,Indice>=361,3117$,Yes:
Else,3118$,Yes
3117$ ASSIGN: Verifica se acabaram os terminais de
cruzamento.NumberOut True=
Verifica se acabaram os terminais de
cruzamento.NumberOut True + 1:NEXT(1302$)
3118$ ASSIGN: Verifica se acabaram os terminais de
cruzamento.NumberOut False=
Verifica se acabaram os terminais de
cruzamento.NumberOut False + 1:NEXT(1223$)
Model statements for module: Dispose 85
1302$ ASSIGN: Dispose 85.NumberOut=Dispose 85.NumberOut + 1
3119$ DISPOSE: Yes
Model statements for module: ReadWrite 81
1220$ WRITE, Log de MovimentacaoMDB,RECORDSET (LogdeUtilizacao):
Dia,
4,
Indice,
AcumulaPP (Indice,1) / 1440,
1,
1,
1:NEXT(1224$)
Model statements for module: ReadWrite 82
1224$ WRITE, Log de MovimentacaoMDB,RECORDSET (LogdeUtilizacao):

```

```

Dia,
4,
Indice,
AcumulaPP (Indice,4) / 1440,
2,
1,
1:NEXT(1229$)
Model statements for module: ReadWrite 83
1229$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeUtilizacao):
Dia,
4,
Indice,
AcumulaPP (Indice,2) / 1440,
1,
1,
2:NEXT(1230$)
Model statements for module: ReadWrite 84
1230$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeUtilizacao):
Dia,
4,
Indice,
AcumulaPP (Indice,5) / 1440,
2,
1,
2:NEXT(1231$)
Model statements for module: ReadWrite 85
1231$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeUtilizacao):
Dia,
4,
Indice,
AcumulaPP (Indice,3) / 1440,
1,
1,
3:NEXT(1232$)
Model statements for module: ReadWrite 86
1232$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeUtilizacao):
Dia,
4,
Indice,
AcumulaPP (Indice,6) / 1440,
2,
1,
3:NEXT(1248$)
Model statements for module: ReadWrite 88
1248$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeUtilizacao):
Dia,
5,
Indice,
AcumulaUPP (Indice,1),
1,
1,
1:NEXT(1249$)
Model statements for module: ReadWrite 89
1249$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeUtilizacao):
Dia,
5,
Indice,
AcumulaUPP (Indice,4),
2,
1,
1:NEXT(1250$)
Model statements for module: ReadWrite 90
1250$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeUtilizacao):
Dia,
5,

```

```

        Indice,
        AcumulaUPP (Indice,2),
        1,
        1,
        2:NEXT (1251$)
    Model statements for module:  ReadWrite 91
1251$      WRITE,      Log de MovimentacaoMDB,RECORDSET (LogdeUtilizacao) :
        Dia,
        5,
        Indice,
        AcumulaUPP (Indice,5),
        2,
        1,
        2:NEXT (1252$)
    Model statements for module:  ReadWrite 92
1252$      WRITE,      Log de MovimentacaoMDB,RECORDSET (LogdeUtilizacao) :
        Dia,
        5,
        Indice,
        AcumulaUPP (Indice,3),
        1,
        1,
        3:NEXT (1253$)
    Model statements for module:  ReadWrite 93
1253$      WRITE,      Log de MovimentacaoMDB,RECORDSET (LogdeUtilizacao) :
        Dia,
        5,
        Indice,
        AcumulaUPP (Indice,6),
        2,
        1,
        3:NEXT (1247$)
    Model statements for module:  Assign 725
1298$      ASSIGN:      Indice=1:NEXT (1304$)
    Model statements for module:  Decide 658
1304$      BRANCH,      1:
        If,V CronoPPT (Indice,1) ==0,3120$,Yes:
        Else,3121$,Yes
3120$      ASSIGN:      Trecho com tempo zerado sentido 1 ?.NumberOut True=
        Trecho com tempo zerado sentido 1 ?.NumberOut True +
1:NEXT (1306$)
3121$      ASSIGN:      Trecho com tempo zerado sentido 1 ?.NumberOut False=
        Trecho com tempo zerado sentido 1 ?.NumberOut False +
1:NEXT (1303$)
    Model statements for module:  Decide 659
1306$      BRANCH,      1:
        If,V CronoPPT (Indice,2) ==0,3122$,Yes:
        Else,3123$,Yes
3122$      ASSIGN:      Trecho com tempo zerado sentido 2 ?.NumberOut True=
        Trecho com tempo zerado sentido 2 ?.NumberOut True +
1:NEXT (1301$)
3123$      ASSIGN:      Trecho com tempo zerado sentido 2 ?.NumberOut False=
        Trecho com tempo zerado sentido 2 ?.NumberOut False +
1:NEXT (1305$)
    Model statements for module:  ReadWrite 193
1301$      WRITE,      Log de Pares por Trecho:
        Dia,
        Indice,
        V NewPPT (indice,1),
        V NewPPT (indice,2),
        V NewPPT (indice,3),
        V NewPPT (indice,4):NEXT (1299$)
    Model statements for module:  Assign 726
1299$      ASSIGN:      V NewPPT (indice,1)=0:
        V NewPPT (indice,2)=0:

```

```

V NewPPT(indice,3)=0:
V NewPPT(indice,4)=0:
Indice=indice + 1:NEXT(1300$)
Model statements for module: Decide 654
1300$      BRANCH,      1:
                        If,indice > 402,3124$,Yes:
                        Else,3125$,Yes
3124$      ASSIGN:      Acabaram os trechos ?.NumberOut True=Acabaram os trechos
?.NumberOut True + 1:NEXT(1185$)
3125$      ASSIGN:      Acabaram os trechos ?.NumberOut False=Acabaram os
trechos ?.NumberOut False + 1:NEXT(1304$)
Model statements for module: Dispose 49
1185$      ASSIGN:      Dispose 49.NumberOut=Dispose 49.NumberOut + 1
3126$      DISPOSE:     Yes
Model statements for module: Assign 730
1305$      ASSIGN:      V NewPPT(Indice,4)=TNOW - V CronoPPT(Indice,2):
                        V CronoPPT(Indice,2)=tnow:NEXT(1301$)
Model statements for module: Assign 729
1303$      ASSIGN:      V NewPPT(Indice,2)=TNOW - V CronoPPT(Indice,1):
                        V CronoPPT(Indice,1)=tnow:NEXT(1306$)
Model statements for module: ReadWrite 95
1257$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeUtilizacao):
                        Dia,
                        6,
                        Indice,
                        V FilasH ( Indice,2 ),
                        0,
                        0,
                        0:NEXT(1261$)
Model statements for module: Decide 621
1276$      BRANCH,      1:
                        If,V Gravar em texto == 1,3127$,Yes:
                        Else,3128$,Yes
3127$      ASSIGN:      Decide 621.NumberOut True=Decide 621.NumberOut True +
1:NEXT(1283$)
3128$      ASSIGN:      Decide 621.NumberOut False=Decide 621.NumberOut False +
1:NEXT(1216$)
Model statements for module: ReadWrite 178
1283$      WRITE,      Log de Utilizacao:
                        Dia,
                        3,
                        Indice,
                        AcumulaPD(indice) / 1440,
                        0,
                        E Patios (V IDTD(indice), 3) * (E Patios (V
IDTD(indice), 3) > 0) + (E Patios (V IDTD(indice), 3) == 0),
                        0:NEXT(1226$)
Model statements for module: ReadWrite 80
1216$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeUtilizacao):
                        Dia,
                        3,
                        Indice,
                        AcumulaPD(indice) / 1440,
                        0,
                        E Patios (V IDTD(indice), 3) * (E Patios (V
IDTD(indice), 3) > 0) + (E Patios (V IDTD(indice), 3) == 0),
                        0:NEXT(1226$)
Model statements for module: ReadWrite 79
1212$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeUtilizacao):
                        Dia,
                        2,
                        V IDTC(Indice),
                        AcumulaPC(indice) / 1440,
                        0,

```

```

      E Patios (V IDTC(indice), 2) * (E Patios (V
IDTC(indice), 2) > 0) + (E Patios (V IDTC(indice), 2) == 0),
      0:NEXT(1225$)
Especificação do arquivo de LOG de eventos:
1 - Dia do evento
2- Identificador único do trem (A IDTrem)
3 - Número do trem (A NTrem)
4 - Tipo do trem (A Tipo)
5 - Momento de início da atividade (unitários) ou Horário do trem (A Horário -
grade/expresso)
6 - Pátio (A PtAtual)
7 - Momento da saída do trem (TNOW)
8 - Atividade (1=carga, 2=descarga, 3=contigência, 4=Saida de grade ou expresso
5=outras)
Model statements for module: ReadWrite 56
1187$      WRITE,      Log de MovimentacaoMDB,RECORDSET(LogdeUtilizacao):
      Dia,
      1,
      Indice,
      (V LocosAcm2(indice) / 1440) / V LocosFrota(indice),
      0,
      V LocosFrota(indice),
      0:NEXT(1228$)
Model statements for module: Assign 591
1262$      ASSIGN:      V TU de ontem=V TUDia:NEXT(1265$)
Model statements for module: Separate 76
1265$      DUPLICATE,      100 - 50:
      1,3131$,50:NEXT(3130$)
3130$      ASSIGN:      Separate 76.NumberOut Orig=Separate 76.NumberOut Orig +
1:NEXT(1273$)
3131$      ASSIGN:      Separate 76.NumberOut Dup=Separate 76.NumberOut Dup +
1:NEXT(1192$)
Model statements for module: ReadWrite 150
1273$      WRITE,      Teste P2," %1.8g %1.8g %1.8g %1.8g\n":
      Dia,
      V TUDia,
      NC(Conta entrando) - NC(Conta saindo),
      Nument:NEXT(1307$)
Model statements for module: ReadWrite 196
1307$      WRITE,      Teste P3," %1.8g %1.8g\n":
      Dia,
      V TUDia:NEXT(1274$)
Model statements for module: Dispose 83
1274$      ASSIGN:      Dispose 83.NumberOut=Dispose 83.NumberOut + 1
3132$      DISPOSE:      Yes
Model statements for module: Decide 423
1192$      BRANCH,      1:
      If,V TUDia < 95000,3133$,Yes:
      Else,3134$,Yes
3133$      ASSIGN:      TU foi baixa hoje ?.NumberOut True=TU foi baixa hoje
?.NumberOut True + 1:NEXT(1193$)
3134$      ASSIGN:      TU foi baixa hoje ?.NumberOut False=TU foi baixa hoje
?.NumberOut False + 1:NEXT(1195$)
Model statements for module: Assign 476
1193$      ASSIGN:      V TUBaixa=V TUBaixa + 1:NEXT(1194$)
Model statements for module: Decide 424
1194$      BRANCH,      1:
      If,V TUBaixa > 3 .AND. (TNOW > ((V Warmup-1) * 1440))
.and. 0,3135$,Yes:
      Else,3136$,Yes
3135$      ASSIGN:      Detector de travamento.NumberOut True=Detector de
travamento.NumberOut True + 1:NEXT(1198$)
3136$      ASSIGN:      Detector de travamento.NumberOut False=Detector de
travamento.NumberOut False + 1:NEXT(1196$)
1198$      VBA:      3,vba:NEXT(1197$)

```



```

Model statements for module: Dispose 60
1197$    ASSIGN:      Dispose 60.NumberOut=Dispose 60.NumberOut + 1
3137$    DISPOSE:     Yes
Model statements for module: Assign 478
1196$    ASSIGN:      V TUDia=0:NEXT(1197$)
Model statements for module: Assign 477
1195$    ASSIGN:      V TUBaixa=0:NEXT(1196$)
Model statements for module: Create 15
3138$    CREATE,      1,MinutesToBaseTime(1),Entity
1:MinutesToBaseTime(1),1:NEXT(3139$)
3139$    ASSIGN:      Cria entidade registradora das utilizacoes.NumberOut=
Cria entidade registradora das utilizacoes.NumberOut +
1:NEXT(1235$)
Model statements for module: Hold 76
1235$    QUEUE,      Aguarda warmup.Queue
SCAN:    TNOW > ((V Warmup) * 1440):NEXT(1186$)
Model statements for module: Assign 286
1186$    ASSIGN:      V LocosAcm(1)=
V LocosAcm(1) * (V LocosNow(1) < V LocosAcm(1)) + V
LocosNow(1) * (V LocosNow(1) >= V LocosAcm(1)):
V LocosAcm(2)=
V LocosAcm(2) * (V LocosNow(2) < V LocosAcm(2)) + V
LocosNow(2) * (V LocosNow(2) >= V LocosAcm(2)):
V LocosAcm(3)=
V LocosAcm(3) * (V LocosNow(3) < V LocosAcm(3)) + V
LocosNow(3) * (V LocosNow(3) >= V LocosAcm(3)):
V LocosAcm(4)=
V LocosAcm(4) * (V LocosNow(4) < V LocosAcm(4)) + V
LocosNow(4) * (V LocosNow(4) >= V LocosAcm(4)):
V LocosAcm(5)=
V LocosAcm(5) * (V LocosNow(5) < V LocosAcm(5)) + V
LocosNow(5) * (V LocosNow(5) >= V LocosAcm(5))
:NEXT(1203$)
Model statements for module: Assign 488
1203$    ASSIGN:      V LocosAcm2(1)=V LocosAcm2(1) + V LocosNow(1):
V LocosAcm2(2)=V LocosAcm2(2) + V LocosNow(2):
V LocosAcm2(3)=V LocosAcm2(3) + V LocosNow(3):
V LocosAcm2(4)=V LocosAcm2(4) + V LocosNow(4):
V LocosAcm2(5)=V LocosAcm2(5) + V
LocosNow(5):NEXT(1204$)
Model statements for module: Assign 489
1204$    ASSIGN:      Indice=1:NEXT(1254$)
Model statements for module: Decide 472
1254$    BRANCH,     1:
If,Desliga Utilizacao Patios == 1,3142$,Yes:
Else,3143$,Yes
3142$    ASSIGN:      Verifica se deve Desligar Utilizacao Patios.NumberOut
True=
Verifica se deve Desligar Utilizacao Patios.NumberOut
True + 1:NEXT(1207$)
3143$    ASSIGN:      Verifica se deve Desligar Utilizacao Patios.NumberOut
False=
Verifica se deve Desligar Utilizacao Patios.NumberOut
False + 1:NEXT(1233$)
Model statements for module: Assign 492
1207$    ASSIGN:      AcumulaPC(1)=AcumulaPC(1) + (E Patios (145, 2) - V
StatusPt (145, 2)):
AcumulaPC(2)=AcumulaPC(2) + (E Patios (132, 2) - V
StatusPt (132, 2)):
AcumulaPC(3)=AcumulaPC(3) + (E Patios (133, 2) - V
StatusPt (133, 2)):
AcumulaPC(4)=AcumulaPC(4) + (E Patios (4, 2) - V
StatusPt (4, 2)):
AcumulaPC(5)=AcumulaPC(5) + (E Patios (1, 2) - V
StatusPt (1, 2)):

```

```

AcumulaPC(6)=AcumulaPC(6) + (E Patios (1, 2) - V
StatusPt (1, 2)):
AcumulaPC(7)=AcumulaPC(7) + (E Patios (129, 2) - V
StatusPt (129, 2)):
AcumulaPC(8)=AcumulaPC(8) + (E Patios (127, 2) - V
StatusPt (127, 2)):
AcumulaPC(9)=AcumulaPC(9) + (E Patios (157, 2) - V
StatusPt (157, 2)):
AcumulaPC(10)=AcumulaPC(10) + (E Patios (41, 2) - V
StatusPt (41, 2)):
AcumulaPC(11)=AcumulaPC(11) + (E Patios (89, 2) - V
StatusPt (89, 2)):
AcumulaPC(12)=AcumulaPC(12) + (E Patios (69, 2) - V
StatusPt (69, 2)):
AcumulaPC(13)=AcumulaPC(13) + (E Patios (72, 2) - V
StatusPt (72, 2)):
AcumulaPC(14)=AcumulaPC(14) + (E Patios (187, 2) - V
StatusPt (187, 2)):
AcumulaPC(15)=AcumulaPC(15) + (E Patios (130, 2) - V
StatusPt (130, 2)):
AcumulaPC(16)=AcumulaPC(16) + (E Patios (211, 2) - V
StatusPt (211, 2)):
AcumulaPC(17)=AcumulaPC(17) + (E Patios (350, 2) - V
StatusPt (350, 2)):
AcumulaPC(18)=AcumulaPC(18) + (E Patios (14, 2) - V
StatusPt (14, 2)):
AcumulaPC(19)=AcumulaPC(19) + (E Patios (20, 2) - V
StatusPt (20, 2)):NEXT(1208$)
  Model statements for module: Assign 493
1208$   ASSIGN:   AcumulaPD(1)=AcumulaPD(1) + V StatusPtD(1):
          AcumulaPD(2)=AcumulaPD(2) +V StatusPtD(2):
          AcumulaPD(3)=AcumulaPD(3) + V StatusPtD(3):
          AcumulaPD(4)=AcumulaPD(4) + V StatusPtD(4):
          AcumulaPD(5)=AcumulaPD(5) + V StatusPtD(5):
          AcumulaPD(6)=AcumulaPD(6) + V StatusPtD(6):
          AcumulaPD(7)=AcumulaPD(7) + V StatusPtD(7):
          AcumulaPD(8)=AcumulaPD(8) + V StatusPtD(8):
          AcumulaPD(9)=AcumulaPD(9) + V StatusPtD(9):
          AcumulaPD(12)=AcumulaPD(12) + V StatusPtD(12):
          AcumulaPD(13)=AcumulaPD(13) + V
StatusPtD(13):NEXT(1237$)
  Model statements for module: Delay 45
1237$   DELAY:   1,,Other:NEXT(1186$)
  Model statements for module: Decide 463
1233$   BRANCH,  1:
          If,NumPP(Indice,1 )==0,3144$,Yes:
          Else,3145$,Yes
3144$   ASSIGN:   Sentido crescente esta vazio?.NumberOut True=Sentido
crescente esta vazio?.NumberOut True + 1
          :NEXT(1241$)
3145$   ASSIGN:   Sentido crescente esta vazio?.NumberOut False=Sentido
crescente esta vazio?.NumberOut False + 1
          :NEXT(1240$)
  Model statements for module: Assign 514
1241$   ASSIGN:   RegPP(Indice,1)=0:NEXT(1239$)
  Model statements for module: Decide 464
1239$   BRANCH,  1:
          If,NumPP(Indice,2 )==0,3146$,Yes:
          Else,3147$,Yes
3146$   ASSIGN:   Sentido decrescente esta vazio?.NumberOut True=Sentido
decrescente esta vazio?.NumberOut True + 1
          :NEXT(1242$)
3147$   ASSIGN:   Sentido decrescente esta vazio?.NumberOut False=Sentido
decrescente esta vazio?.NumberOut False + 1
          :NEXT(1245$)

```

```

Model statements for module: Assign 515
1242$      ASSIGN:      RegPP (Indice,2)=0:NEXT (1205$)
Model statements for module: Assign 490
1205$      ASSIGN:      AcumulaPP (Indice,1)=AcumulaPP (Indice,1)
+StatusPP (Indice,1) :
                        AcumulaPP (Indice,2)=AcumulaPP (Indice,2)
+StatusPP (Indice,2) :
                        AcumulaPP (Indice,3)=AcumulaPP (Indice,3)
+StatusPP (Indice,3) :
                        AcumulaPP (Indice,4)=AcumulaPP (Indice,4)
+StatusPP (Indice,4) :
                        AcumulaPP (Indice,5)=AcumulaPP (Indice,5)
+StatusPP (Indice,5) :
                        AcumulaPP (Indice,6)=AcumulaPP (Indice,6)
+StatusPP (Indice,6) :
                        Indice=indice + 1:NEXT(1206$)
Model statements for module: Decide 453
1206$      BRANCH,      1:
                        If,Indice>=275,3148$,Yes:
                        Else,3149$,Yes
3148$      ASSIGN:      Verifica se acabaram os patios.NumberOut True=Verifica
se acabaram os patios.NumberOut True + 1
                        :NEXT (1238$)
3149$      ASSIGN:      Verifica se acabaram os patios.NumberOut False=Verifica
se acabaram os patios.NumberOut False + 1
                        :NEXT (1233$)
Model statements for module: Assign 513
1238$      ASSIGN:      aTemp3=dia:NEXT (1207$)
Model statements for module: Decide 467
1245$      BRANCH,      1:
                        If,aTemp3 <> dia,3150$,Yes:
                        Else,3151$,Yes
3150$      ASSIGN:      Mudou o dia para sentido decrescente ?.NumberOut True=
Mudou o dia para sentido decrescente ?.NumberOut True +
1:NEXT (1244$)
3151$      ASSIGN:      Mudou o dia para sentido decrescente ?.NumberOut False=
Mudou o dia para sentido decrescente ?.NumberOut False +
1:NEXT (1246$)
Model statements for module: Assign 516
1244$      ASSIGN:      AcumulaUPP (Indice,4)=AcumulaUPP (Indice,4)
+StatusPP (Indice,4) :
                        AcumulaUPP (Indice,5)=AcumulaUPP (Indice,5)
+StatusPP (Indice,5) :
                        AcumulaUPP (Indice,6)=AcumulaUPP (Indice,6)
+StatusPP (Indice,6) :
                        RegPP (Indice,2)=NumPP (Indice,2 ) :NEXT (1205$)
Model statements for module: Decide 468
1246$      BRANCH,      1:
                        If,RegPP (Indice,2) <> NumPP (Indice,2 ),3152$,Yes:
                        Else,3153$,Yes
3152$      ASSIGN:      Mudou o trem para sentido decrescente ?.NumberOut True=
Mudou o trem para sentido decrescente ?.NumberOut True +
1:NEXT (1244$)
3153$      ASSIGN:      Mudou o trem para sentido decrescente ?.NumberOut False=
Mudou o trem para sentido decrescente ?.NumberOut False
+ 1:NEXT (1205$)
Model statements for module: Decide 465
1240$      BRANCH,      1:
                        If,aTemp3 <> dia,3154$,Yes:
                        Else,3155$,Yes
3154$      ASSIGN:      Mudou o dia para sentido crescente ?.NumberOut True=
Mudou o dia para sentido crescente ?.NumberOut True +
1:NEXT (1234$)
3155$      ASSIGN:      Mudou o dia para sentido crescente ?.NumberOut False=

```

```

Mudou o dia para sentido crescente ?.NumberOut False +
1:NEXT(1243$)
  Model statements for module: Assign 512
1234$   ASSIGN:   AcumulaUPP (Indice,1)=AcumulaUPP (Indice,1)
+StatusPP (Indice,1) :
        AcumulaUPP (Indice,2)=AcumulaUPP (Indice,2)
+StatusPP (Indice,2) :
        AcumulaUPP (Indice,3)=AcumulaUPP (Indice,3)
+StatusPP (Indice,3) :
        RegPP (Indice,1)=NumPP (Indice,1 ):NEXT (1239$)
  Model statements for module: Decide 466
1243$   BRANCH,   1:
        If,RegPP (Indice,1) <> NumPP (Indice,1 ),3156$,Yes:
        Else,3157$,Yes
3156$   ASSIGN:   Mudou o trem para sentido crescente ?.NumberOut True=
        Mudou o trem para sentido crescente ?.NumberOut True +
1:NEXT (1234$)
3157$   ASSIGN:   Mudou o trem para sentido crescente ?.NumberOut False=
        Mudou o trem para sentido crescente ?.NumberOut False +
1:NEXT(1239$)
  Model statements for module: Create 16
3158$   CREATE,   1,MinutesToBaseTime(1),Entity
1:MinutesToBaseTime(1):NEXT(3159$)
3159$   ASSIGN:   Cria entidade gravadora do status das froτας.NumberOut=
        Cria entidade gravadora do status das froτας.NumberOut +
1:NEXT(1200$)
  Model statements for module: Decide 451
1200$   BRANCH,   1:
        If,TNOW > ((V Warmup) * 1440),3162$,Yes:
        Else,3163$,Yes
3162$   ASSIGN:   Warmup do log de locos.NumberOut True=Warmup do log de
locos.NumberOut True + 1:NEXT(1280$)
3163$   ASSIGN:   Warmup do log de locos.NumberOut False=Warmup do log de
locos.NumberOut False + 1:NEXT(1201$)
  Model statements for module: Decide 625
1280$   BRANCH,   1:
        If,V Gravar em texto == 1,3164$,Yes:
        Else,3165$,Yes
3164$   ASSIGN:   Decide 625.NumberOut True=Decide 625.NumberOut True +
1:NEXT(1297$)
3165$   ASSIGN:   Decide 625.NumberOut False=Decide 625.NumberOut False +
1:NEXT(1190$)
  Model statements for module: ReadWrite 192
1297$   WRITE,   Log de Locos:
        tnow,
        V LocosAcm(1),
        V LocosAcm(2),
        V LocosAcm(3),
        V LocosAcm(4),
        V LocosAcm(5):NEXT(1202$)
  Model statements for module: Assign 485
1202$   ASSIGN:   V LocosAcm(1)=0:
        V LocosAcm(2)=0:
        V LocosAcm(3)=0:
        V LocosAcm(4)=0:
        V LocosAcm(5)=0:NEXT(1189$)
  Model statements for module: Dispose 55
1189$   ASSIGN:   Dispose 55.NumberOut=Dispose 55.NumberOut + 1
3166$   DISPOSE:   Yes
  Model statements for module: ReadWrite 59
1190$   WRITE,   Log de MovimentacaoMDB,RECORDSET (LogdeLocos) :
        tnow,
        V LocosAcm(1),
        V LocosAcm(2),
        V LocosAcm(3),

```

```

                V LocosAcm(4),
                V LocosAcm(5):NEXT(1202$)
    Model statements for module: Dispose 62
1201$      ASSIGN:      Dispose 62.NumberOut=Dispose 62.NumberOut + 1
3167$      DISPOSE:      Yes
    Model statements for module: Create 17
3168$      CREATE,      1,HoursToBaseTime(0.0),Entity
1:HoursToBaseTime(EXPO(1)),1:NEXT(3169$)
3169$      ASSIGN:      Cria entidade configuradora.NumberOut=Cria entidade
configuradora.NumberOut + 1:NEXT(1255$)
    Model statements for module: Assign 546
1255$      ASSIGN:      MR(Helpers1)=E Helpers(5,1):
                        MR(Helpers2)=E Helpers(6,1):
                        MR(Helpers3)=E Helpers(7,1):
                        MR(Helpers5)=E Helpers(8,1):
                        MR(Helpers6)=E Helpers(9,1):
                        MR(Helpers7)=E Helpers(10,1):NEXT(1256$)
    Model statements for module: Dispose 63
1256$      ASSIGN:      Dispose 63.NumberOut=Dispose 63.NumberOut + 1
3172$      DISPOSE:      Yes
    Model statements for module: Assign 710
1266$      ASSIGN:      Sh1=1:NEXT(1267$)
    Model statements for module: Separate 77
1267$      DUPLICATE,   100 - 50:
                        23,3175$,50:NEXT(3174$)
3174$      ASSIGN:      Separate 77.NumberOut Orig=Separate 77.NumberOut Orig +
1:NEXT(1268$)
3175$      ASSIGN:      Separate 77.NumberOut Dup=Separate 77.NumberOut Dup +
1:NEXT(1268$)
    Model statements for module: Assign 711
1268$      ASSIGN:      A aTemp3=sh1:
                        Sh1=sh1 + 1:
                        Sh2=V TUDia:NEXT(1269$)
    Model statements for module: Delay 56
1269$      DELAY:      HoursToBaseTime(A aTemp3 - (1/60)),,Other:NEXT(1270$)
    Model statements for module: Assign 712
1270$      ASSIGN:      Temp1=V TUDia - Sh2:
                        Sh2=V TUDia:NEXT(1273$)
    Model statements for module: Create 18
3176$      CREATE,      1,HoursToBaseTime(0.0),Entity
1:HoursToBaseTime(EXPO(1)),1:NEXT(3177$)
3177$      ASSIGN:      Create 18.NumberOut=Create 18.NumberOut + 1:NEXT(1271$)
    Model statements for module: ReadWrite 149
1271$      WRITE,      Teste P2,"dia TUDia Retidos Nument\n"::NEXT(1272$)
    Model statements for module: Dispose 82
1272$      ASSIGN:      Dispose 82.NumberOut=Dispose 82.NumberOut + 1
3180$      DISPOSE:      Yes

```