Tecnologias SoC e CAN em sistemas de controle distribuído

Jean Mimar Santa Cruz Yabarrena

Maio 2006

Agenda

- Objetivo e Motivação
- Introdução a Sistemas de tempo real, sistema embarcado, SoC, barramentos de campo
- Implementação
- Resultados
- Conclusões e Provocações

Objetivo

- O Objetivo deste trabalho é implementar um SoC para um sistema de controle.
- Desafio do design.
- Estratégias de projeto e validação de IP cores (CAN, quadrature, PWM)
- A motivação na area aeroespacial e mudança de paradigmas.

Introdução

- Sistemas de tempo real

Hard Real-Time

(Cíclico)

Soft Real-Time

(event-driven)

Fonte: EPFL

Introdução

● Sistemas embarcados

CRISE DE IDENTIDAD

Sistema de Tempo-Real e Sistema Embarcado

Suposição popular respeito a Sistema Embarcado e de Tempo-Real

Fonte: Montavista Linux™

Introdução

Visão real dos sistemas

Firm real-time e Weakly real-time (Guillen Bernat)

Fonte: Montavista Linux™

Introdução

● Protocolos de Campo

Redes para aplicações *Safety-critical*

- MILSTD 1553
- ARINC629 Boeing 777
- SAFEBus ARINC659 (Honeywell) Boeing 777 4X
- TTA. Time Triggered Architecture(U. Vienna) Audi Volkswagen.
- SPIDER Scalable Processor-Independent Design for Electromagnetic Resilience (NASA Langley RC) DO-254
- FlexRay BMW, DaimlerChrysler, Motorola e Philips.
- TTP Airbus A380

Implementação

Implementação

Software

Hardware

Morphware

Por que FPGAs

- Plataforma pre-engineered
- Tecnologia 90 nm.
- Software e metodologia
- Ferramentas avançadas de depuração
- Suporte de RTOS, vendedores de arquiteturas
- Co-design

1958

A device having multiple electrical components and their interconnects manufactured on a single substrate

Noções de Espaço Tempo

Computer Architecture, Computer Organization and Design, Patterson & Hennessy, ©2005

Implementação

Implementação

Implementação

● Hardware

The eZdsp F2812

- TMS320F2812 Digital Signal Processor
- 150 MIPS operating speed
- 18K words on-chip RAM
- 128K words on-chip Flash memory
- 64K words off-chip SRAM memory
- 30 MHz. clock
- 2 Expansion Connectors (analog, I/O)
- Onboard IEEE 1149.1 JTAG Controller
- 5-volt only operation with supplied AC adapter
- TI F28xx Code Composer Studio tools driver
- On board IEEE 1149.1 JTAG emulation connector

Implementação

● Morphware

Virtex Pro II XC2VP30

- Virtex-2 Pro XC2VP30 FPGA with 30,816 Logic Cells, 136 18-bit multipliers, 2,448Kb of block RAM, and two PowerPC Processors
- DDR SDRAM DIMM that can accept up to 2Gbytes of RAM
- 10/100 Ethernet port
- USB2 port
- Compact Flash card slot
- XSGA Video port
- Audio Codec
- SATA, and PS/2, RS-232 ports
- High and Low Speed expansion connectors with a large collection of available expansion boards

Implementação

● Morphware

- System gates 200K
- Logic cells 5.292
- BRAM 56Kbits
- DLL 4
- I/O 289

Spartan 2e XC2S200E

Arquitetura Virtex II Pro

□ Configurable logic block (CLB)

Arquitetura Virtex II Pro

Arquitetura Virtex II Pro

Arquitetura Virtex II Pro

Arquitetura Virtex II Pro

Arquitetura Virtex II Pro

Arquitetura Virtex II Pro

Arquitectura Virtex II Pro

- Configurable logic block (CLB)
- Input Output Block (IOB)
- 18 Kb Block RAM
- 18x18 bit multiplier
- 2.5 Gbps Rocket I/O transceiver
- Digital Clock Manager (DCM)
- Processor block
- Global and local routing

Arquitectura Virtex II Pro

- Configurable logic block (CLB)
- Input Output Block (IOB)
- 18 Kb Block RAM
- 18x18 bit multiplier
- 2.5 Gbps Rocket I/O transceiver
- Digital Clock Manager (DCM)
- Processor block
- Global and local routing
- CANBus Controller

SoC

● Alvo: Controle do motor DC

IP quadrature

PWM

CAN controller

SoC

● Integração Bus-on-Chip

PWM

Encoder

CAN

Microblaze

Arquitetura

System on PC baseado no padrão
CoreConnect

Resultados

IP PWM

```
entity pwm_bsp is
generic
(
    C_WIDTH: integer := 4
);
Port ( clk_i: in STD_LOGIC;
      rst_i: in STD_LOGIC;
      cs_i: in STD_LOGIC;
      wr_i: in STD_LOGIC;
      addr_i: in STD_LOGIC;
      data_io: inout STD_LOGIC_VECTOR(C_WIDTH-1 downto 0);
      tx_o: out STD_LOGIC;
end pwm_bsp;
```

Resultados

IP Encoder

```
entity quadrature_bsp is
  Generic
  (
    C_WIDTH: integer := 32
  )
  Port (clk_i: in STD_LOGIC;
        rst_i: in STD_LOGIC;
        cs_i: in STD_LOGIC;
        rx_a_i: in STD_LOGIC;
        rx_b_i: in STD_LOGIC;
        rx_idx_i: in STD_LOGIC;
        addr_i: in STD_LOGIC_VECTOR (1 downto 0);
        data_io: out STD_LOGIC_VECTOR (C_WIDTH-1 downto 0));
end quadrature_bsp;
```

addr_i(1:0)	data_io(31:0)
clk_i	
cs_i	
rst_i	
rx_a_i	
rx_b_i	
rx_idx_i

Resultados

Controlador CAN

ale_i	bus_off_on
clk_i	clkout_o
cs_can_i	
rd_i	irq_on
rst_i	tx_o
rx_i	
wr_i	port_0_io(7:0)

```
void CANInitialize(void)
{
  // Set bus timing registers
  can_write_data(0x06, 0x03) /* B
  90kHz
  can_write_data(0x07, 0x28) /* BT
  // Set clock divider
  can_write_data(0x1f, 0x05) /* 0
  // 05 > 8.333
  // Set acceptance mode
  can_write_data(0x04, 0xe8) /* AC
  // Set acceptance mask
  can_write_data(0x05, 0xff) /* AMF
  // Switch off reset mode
  can_write_data(0x00, 0x00) /* OF
  sleep(2);
  // Begin message transmission
}
```

```
can_write_data(unsigned char address, unsigned char data)
{
  unsigned short int address_16 = 0;
  unsigned short int data_16 = 0;
```

```
address_16 = 0x0C00 | address; // Set CS = 1 ALE =1
data_16 = 0x0500 | data; // Set CS = 1 WR =1
```

```
CAN_CONTROLLER_mWriteSlaveRegX
XPAR_CAN_CONTROLLER_0_BASEADDR, address_16);
```

```
CAN_CONTROLLER_mWriteSlaveRegX \
BaseAddress, Value) XIo_Out8(BaseAddress) + \
(CAN_CONTROLLER_SLAVE_REG_OFFSET), (XuintB)(Value))
```

```
CAN_CONTROLLER_mWriteSlaveRegX
XPAR_CAN_CONTROLLER_0_BASEADDR, data_16);
```

```
CAN_CONTROLLER_mWriteSlaveRegX
XPAR_CAN_CONTROLLER_0_BASEADDR, data_16);
```

```
*(int *) BaseAddress + offset = Value;
/* mov 0xA00C000, Value */ // Chamada ao arbitrador OPB
```

Resultados

Arquitetura de desenvolvimento

- Integração do sistema

Footprint

90nm.

180nm.

Recursos utilizados no teste do controlador CAN (8Kgates)

SYNC_FSM_TASK Project Status				
Project File:	sync_fsm_task.isc	Current State:	Placed and Routed	
Module Name:	send_frame_basic	• Errors:	No Errors	
Target Device:	xc2vp30-7ff896	• Warnings:	155 Warnings	
Product Version:	ISE, 8.1.02	• Updated:	seg 22. mai 03:34:53 2006	

Device Utilization Summary				
Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	359	27,392	1%	
Number of 4 input LUTs	780	27,392	2%	
Logic Distribution				
Number of occupied Slices	516	13,696	3%	
Number of Slices containing only related logic	516	516	100%	
Number of Slices containing unrelated logic	0	516	0%	
Total Number 4 input LUTs	783	27,392	2%	
Number used as logic	780			
Number used as a route-thru	3			
Number of bonded IOBs	5	556	1%	
IOB Flip Flops	1			
Number of PPC405s	0	2	0%	
Number of GCLKs	2	16	12%	
Number of GTs	0	8	0%	
Number of GT10s	0	0	0%	
Total equivalent gate count for design	8,043			
Additional JTAG gate count for IOBs	240			

Recursos utilizados no teste do SoC

Virtex-II Pro

191	Device Utilization Summary:		
192			
193	Number of BSCANs	1 out of 1	100%
194	Number of BUFMUXs	2 out of 16	12%
195	Number of DCMs	1 out of 8	12%
196	Number of External IOBs	18 out of 556	3%
197	Number of LOCed IOBs	11 out of 18	61%
198			
199	Number of MULT18X18s	7 out of 136	5%
200	Number of RAMB16s	19 out of 136	13%
201	Number of SLICES	3255 out of 13696	23%
202	Number of TBUFs	1 out of 6848	1%

Spartan 2e

Device Utilization Summary:			
Number of BLOCKRAMs	11 out of 14	78%	
Number of BSCANs	1 out of 1	100%	
Number of GCLKs	2 out of 4	50%	
Number of External GCLKIOBs	1 out of 4	25%	
Number of LOCed GCLKIOBs	1 out of 1	100%	
Number of External IOBs	36 out of 142	25%	
Number of LOCed IOBs	36 out of 36	100%	
Number of SLICES	2101 out of 2352	89%	

Estudo de caso

- Controle do motor MAXON 148008
- PID, SoC, Loop de controle via Rede

Visão projetada do Sistema

Visão global do sistema

Resposta Degrau

- $K_p = 0,5$
- $K_i = 0,033$
- $K_d = 0,1095$
- CAN 1Mbps
- Set_point = 70rad/s
- 668.5 rpm.

Conclusões

- Ferramentas e conhecimento é possível redirecionar domínio absoluto de software para hardware.
- Co-design usando Microblaze
- Os IP podem ser reutilizados em outras pesquisas
- O processo de escolha da arquitetura abrangente.

Cinco formas de implementar un SoC

Trabalhos futuros

- Implementação TTCAN
- Estudo comparativo DSP FPGA
- Estudo RTOS em SoC
- Impacto dos tempos nos sistemas de controle baseados em rede
- Injeção de erro no barramento (70 controladores CAN)

Provocações

Provocações

Provocações

Provocações

Obrigado.