Projeto de um controlador PID para controle de ganho de uma câmera com sensor CMOS utilizando computação reconfigurável

Drausio Linardi Rossi

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito: 10/01/2012

Assinatura:_____

Projeto de um controlador PID para controle de ganho de uma câmera com sensor CMOS utilizando computaçãoo reconfigurável

Dráusio Linardi Rossi

Orientador: **Prof. Dr. Eduardo Marques**

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

USP – São Carlos
Janeiro de 2012

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados fornecidos pelo(a) autor(a)

R831p

Rossi, Drausio Linardi
Projeto de um controlador PID para controle de ganho de uma câmera com sensor CMOS utilizando computação reconfigurável / Drausio Linardi Rossi; orientador Eduardo Marques -- São Carlos, 2011.
71 p.

Dissertação (Mestrado - Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional) -- Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2011.

1. CIÊNCIA DA COMPUTAÇÃO. 2. COMPUTAÇÃO RECONFIGURÁVEL. 3. CIRCUITOS FPGA. I. Marques, Eduardo, orient. II. Título.

Dedico este trabalho

À Deus

Ao meu querido filho Hiago, e a minha esposa Márcia, pelo incentivo e compreenção.

À minha família

A sabedoria

Primeiro ela o conduzirá por caminhos tortuosos, causando-lhe medo e tremor, e o atormentará com sua disciplina, até que o homem confie nela e até que ela o tenha provado com suas exigências. Depois ela o reconduzirá pelo caminho reto, o alegrará e lhe manifestrá os seus segredos.
Eclesiástico 17-18

Agradecimentos

Aos colegas do LCR, em especial ao Leandro Martinez, que se fez presente em horas difíceis.

À Whirlpool do Brasil, pelas horas cedidas, para que este trabalho fosse viabilizado.

Aos amigos Miguel Martins, Mario Luis Botêga Jr, Bruno Franciscon Mazzotti e Luiz Henrique Kiehn.

Ao orientador e amigo Professor Doutor Eduardo Marques, pela orientação, amizade e sobretudo pela confiança.

Ao Professor Doutor João Miguel Gago Pontes de Brito Lima da Universidade do Algarve, Portugal, pelo apoio na área de controladores PID, essencial na conclusão deste trabalho.

Sumário

1 Introdução **1**

1.1 Descrição do Sistema Proposto 1

1.2 Objetivos 3

1.3 Organização do Trabalho 4

2 Fundamentação Teórica **5**

2.1 Sistemas de Controle 5

2.1.1 Controladores Analógicos 6

2.1.2 Controladores Digitais 6

2.2 Controlador PID 7

2.2.1 Histórico 7

2.2.2 Conceito 8

2.2.3 Termo Proporcional 9

2.2.4 Termo Integral 9

2.2.5 Termo Derivativo 11

2.2.6 Controlador PID Digital 12

2.2.7 Ajuste do controlador PID 14

2.3 Histograma 14

2.3.1 Conceito 14

2.3.2 Especificação direta de histograma 20

2.4 Tecnologia de Sensores de Imagem Digital 22

2.4.1 Introdução 22

2.4.2 Sensor de Imagem CCD 26

2.4.3 Convertendo Fótons em carga elétrica - Efeito Fotoelétrico 29

2.4.4 Processo de Leitura de Carga 30

2.4.5 Arquitetura da Matriz de *Pixels* CCD 31

2.4.6 Sensor de Imagem CMOS 33

2.4.7 Processo de Leitura de Carga 36

2.4.8 Fatores que Comprometem a Qualidade da Imagem 36

2.4.9 Comparativo entre CCD e CMOS 37
2.5 Computação Reconfigurável 41
2.5.1 FPGAs 44
2.5.2 Estruturas das FPGAs 46

3 Plataforma de Desenvolvimento 49
3.1 Kit de Desenvolvimento DE2 70 49
3.1.1 Placa de Desenvolvimento DE2 70 50
3.1.2 Ferramentas de Desenvolvimento de Software Embarcado 51
3.1.3 Processador RISC Customizável 52
3.1.4 Ferramentas GNUpro para Software Embarcado 55
3.2 Módulo da Câmera CMOS 55
3.3 Conexão da Câmera com a Placa de Desenvolvimento D2 - 70 58
3.4 Monitoramento da Imagem no Sistema 58

4 Implementação do Sistema Proposto 59
4.0.1 Cálculo do histograma e cálculo do erro 60

5 Resultados do Sistema Proposto 64

6 Conclusão 68

7 Trabalhos Futuros 70

Referências Bibliográficas 72

Lista de Figuras

1.1 Fluxo de Dados na Detecção de Pedestres (de Holanda, 2010) 2
1.2 Caminho do sinal analógico-digital e realimentação do ganho analógico, adaptada de (www.terasic.com, 2008) 3

2.1 Sistema de Controle Digital, adaptada de (Ahmed, 1997). 7
2.2 Controlador PID contínuo, paralelo de malha fechada (Åström e Hägglund, 2006) 8
2.3 Resposta de um controlador PID para variações de ganho proporcional Kp 10
2.4 Resposta de um controlador PID para variações de ganho Integral Ki . . . 11
2.5 Resposta de um controlador PID para variações de ganho Derivativo Kd . 12
2.6 Exemplo de valores de conversão A/D de 8 bits para imagem com cinco níveis de cinza 15
2.7 Exemplo de histograma para imagem com oito níveis de cinza (Filho e Neto, 1999) 17
2.8 Exemplo de histogramas (Filho e Neto, 1999) 18
2.9 Imagens correspondentes aos histogramas da figura 2.8 (Filho e Neto, 1999) 19
2.10 Gráfico: Diâmetro da Pupila em Função da Luminância (Moon e Spencer, 1944) 22
2.11 Sistema digital de imagem (A. El Gamal, 2005) 25
2.12 Sensores CCD movem a carga gerada pelos fótons de pixel a pixel até convertê-la em voltagem em uma saída. Sensores CMOS convertem a carga em voltagem dentro de cada pixel (Litwiller, 2005) 25
2.13 Anatomia de um Sensor CCD (www.circuitstoday.com, 2010) 28
2.14 Conjunto Sensor CCD mais circuito de câmera (http://sensorcleaning.com, 2010) 28
2.15 Interação Fótons, Comprimento de Onda e Silício (Company, 2008b) . . . 30
2.16 Técnica *three-phases* de transferência de carga (Company, 2008b) 31
2.17 Arquitetura de dispositivos CCD (Taylor, 1998) 32

2.18 Conceito básico dos sensores CMOS inteligentes. (a) Processamento a nível de *Chip* , (b) Processamento a nível de coluna, e (c) processamento a nível de píxel Ohta (2008). 33
2.19 Conjunto Sensor CMOS mais circuito de câmera (http://sensorcleaning.com, 2010). 35
2.20 Flexibilidade versus Performance de classes de processadores (Bobda, 2007). 44
2.21 Estrutura de uma FPGA (Bobda, 2007) 45
2.22 Quatro tipos básicos de estruturas de FPGAs (Bobda, 2007) 47

3.1 Placa de desenvolvimento DE2 70 (www.terasic.com, 2008) 52
3.2 Componentes do sistema com processador Nios II 54
3.3 Sensor CMOS Visto ao Microscópio 56
3.4 Descrição do Vetor de Pixels (www.terasic.com, 2008) 56
3.5 Câmera CMOS D5M conectada à placa de desenvolvimento DE2-70 (www.terasic.com, 2008). 57
3.6 Temporização Default de Saída do Frame (www.terasic.com, 2008). 58

4.1 Implementação do sistema utilizando FPGA 60
4.2 Cálculo do Erro Através da Análise do Histograma 61
4.3 Arquitetura simplificada do controlador PID 62
4.4 Arquitetura simplificada do cálculo do histograma e cálculo do erro. 63

5.1 Resposta do Sistema com variação do Ganho Proporcional 65
5.2 Resposta do Sistema com variação do Ganho Integral 66
5.3 Resposta do Sistema com variação do Ganho Derivativo 67

Lista de Tabelas

2.1 Exemplo de histograma (Filho e Neto, 1999). 16
2.2 CCD vs CMOS (Litwiller, 2005). 37
2.3 Performance CCD vs CMOS (Litwiller, 2005). 37
2.4 O Caminho Tortuoso do Desenvolvimento dos Sensores CMOS (Litwiller, 2005) 38

3.1 Características do dispositivo EP2C70F896C6N 50

Resumo

Este trabalho propõe um controlador PID (Proporcional, Integrador, Derivativo), implementado em hardware reconfigurável, para controle de ganho de uma câmera com sensor CMOS. O conceito utilizado é o de sistemas SoC (System-on-a-Chip). As principais funções realizadas pelo sistema são: Aquisição da imagem, montagem do histograma, análise do histograma, controle de ganho baseado na análise do histograma.

O sistema proposto tem como objetivo conter algumas funções básicas de controle de ganho que possam servir de base para construção de sistemas de visão computacional que possibilitem a otimização do tempo gasto na construção de novos sistemas, deixando o projetista concentrado na parte mais específica do sistema. O algoritmo de controle de ganho através da análise de histograma demonstrou ser além de funcional, altamente flexível, pois pode ser aplicado a qualquer câmera, independente do tipo do sensor. Este algoritmo pode ser aplicado a tipos diferentes de sensores, com diferentes taxas de aquisição e transmissão de imagens.

Este ambiente baseado em computação reconfigurável proporciona alta performance e flexibilidade no modo de implementação, possibilitando que o hardware seja configurado para satisfazer situações que exigem alto desempenho, que pode ser obtido através do paralelismo de operações. Esta arquitetura ainda possibilita a configuração de processadores que executam operações em software em conjunto com operações executadas em hardware.

O sistema final controla a câmera CMOS de maneira adequada às aplicações robóticas de tempo real.

Abstract

This paper proposes a PID controller (Proportional, Integrator, Derivative), implemented in reconfigurable hardware to control a CMOS sensor camera gain. The concept used is the system SoC (System-on-a-Chip). The main functions performed by the system are: image acquisition, assembly of the histogram, histogram analysis, gain control based analysis of the histogram.

The proposed system aims to contain some basic gain control functions. These functions may serve as a basis for future construction of computer vision systems. This work will optimize the time spent in building new systems, leaving the designer free to concentrate on more specific development. The gain control algorithm through the analysis of histogram proved be functional, highly flexible, and it can be applied to any camera, regardless of the type of sensor. This algorithm can be applied to different types of image sensors with different acquisition and transmission rates.

This environment-based reconfigurable computing provides high performance and flexibility in implementation, enabling the hardware to be configured to meet situations that require high performance, which can be obtained through parallelism of operations. This architecture also enables the configuration of processors that perform software operations in conjunction with hardware operations.

The final system controls the CMOS camera accordingly to real-time robotic applications.

Capítulo

1

Introdução

Implementar um controlador de ganho geral para uma câmera CMOS demonstrou ser uma tarefa desafiadora, visto que a maioria das propostas e implementações apresentam abordagens diferentes como controladores de ganho a nível de pixel. Outro desafio se encontra na escassa literatura a respeito de controladores de ganho de câmeras com sensores CMOS, uma vez que a maioria dos métodos são de e de propriedade dos fabricantes, e que na sua maioria, já os oferece como funções automáticas para as câmeras.

1.1 Descrição do Sistema Proposto

O projeto propôs a implementação e a análise de um controlador PID *(Proporcional, Integral, Derivativo)* para controlar o ganho analógico de uma câmera CMOS de baixo custo. (R. Drausio, 2011)

O sinal de erro do controlador foi baseado na análise do histograma da imagem aquisitada pelo sensor.

Foi utilizada a placa de desenvolvimento DE2 70 da Terasic e uma câmera TRDBD5M, também da Terasic, que possui um sensor CMOS de 5 mega pixels. A arquitetura utilizada é reconfigurável, o que torna o seu desenvolvimento flexível.

O controlador baseado em arquitetura SoC para controle de ganho será integrado a um ambiente para projeto e implementação de controle evolucionário embarcado de detecção de pedestres, como é mostrado na figura 1.1. Este ambiente vem sendo desenvolvido pelo Laboratório de Computação Reconfigurável - LCR do Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo. A base da pirâmide representa a percepção do ambiente e é constituída por um conjunto de sensores que são responsáveis pela percepção das condições ambientais. Esta etapa é de suma importância, pois é a base para o processamento futuro dos dados e a aplicação dos modelos propostos. O sistema de visão necessita lidar com diferentes condições como chuva, sol, neblina, faróis de automóveis, além de mudanças bruscas nestas condições, como situações de sobra como entrada e saída de túneis, sombras, etc.

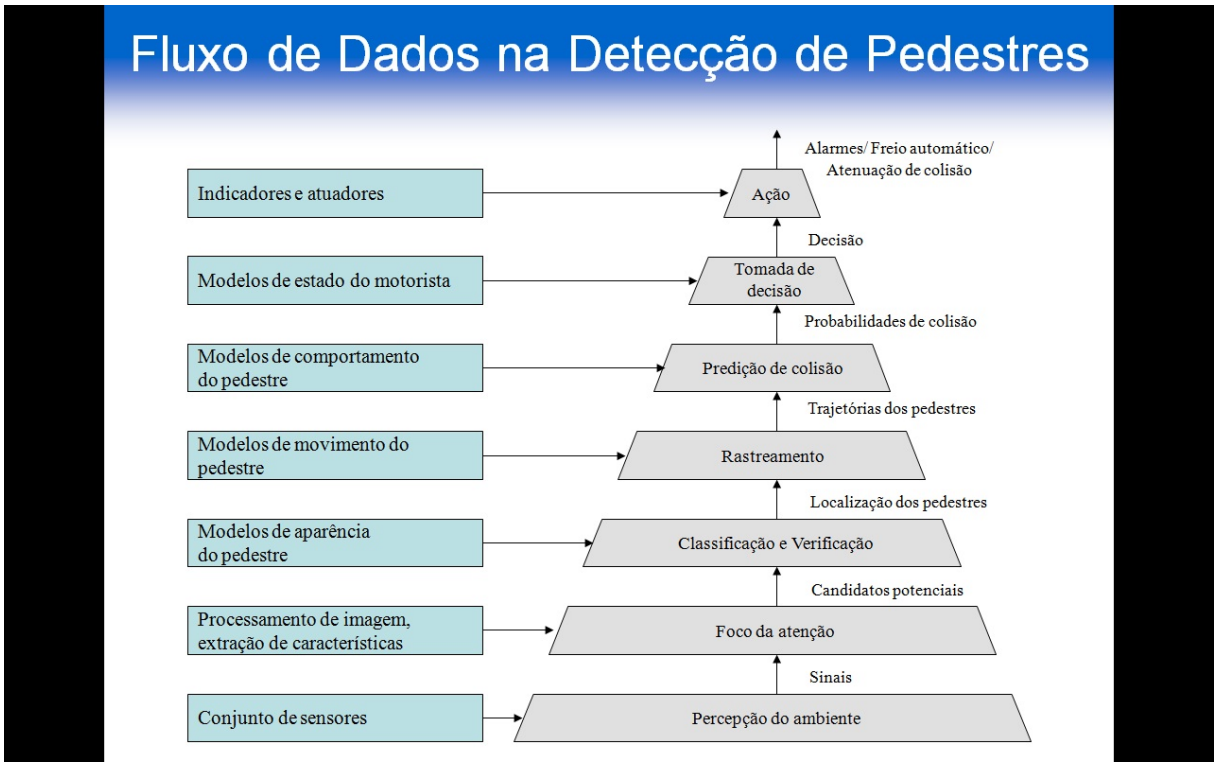

Figura 1.1: Fluxo de Dados na Detecção de Pedestres (de Holanda, 2010)

O sistema proposto será divido em 4 etapas: a primeira realiza a montagem do histograma da imagem, a segunda analisa o histograma, a terceira é realizada pelo controlador PID que calcula o ganho necessário para a câmera e a quarta realiza a realimentação do ganho no sensor CMOS. Para cada função descrita acima, módulos específicos foram implementados em computação reconfigurável, sendo um módulo para cada função do sistema.

O caminho do sinal e do controle de ganho é mostrado na figura 1.2 Cada pixel do sensor realiza a leitura da luz ambiente, e o seu valor é transferido para um conversor analógico digital, passando por um multiplicador de ganho analógico, que é o nosso foco neste trabalho. Todas as outras etapas de controle de ganho, como calibração de nível de negro, ganho digital e correção de offset digital, foram desabilitadas.

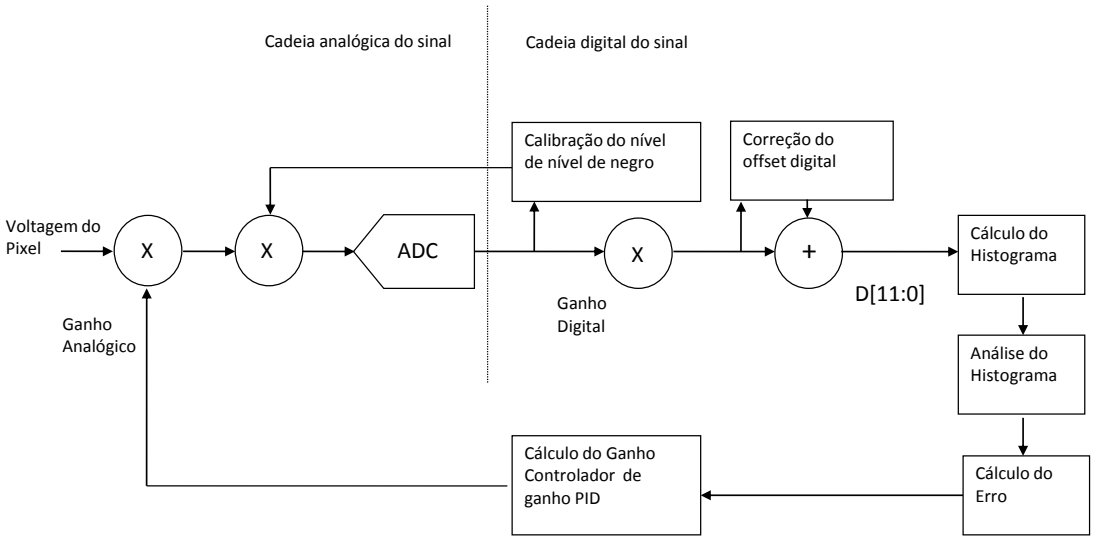

Figura 1.2: Caminho do sinal analógico-digital e realimentação do ganho analógico, adaptada de (www.terasic.com, 2008)

1.2 Objetivos

Os Objetivos principais deste trabalho são:

- Desenvolver um controlador PID em dispositivo FPGA, aplicado ao controle de ganho de uma câmera dotada de um sensor CMOS com a finalidade de servir de base para o desenvolvimento de módulos de visão computacional mais específicos aplicados a robôs móveis.

- Dominar a tecnologia da câmera CMOS (Complementary Metal Oxide Semiconductor), e dominar a tecnologia, linguagens e técnicas de programação de dispositivos reconfiguráveis;

- Contribuir com a evolução do ambiente para projeto e implementação de controle evolucionário embarcado de robôs móveis reconfiguráveis dinamicamente, que vem sendo desenvolvido pelo Laboratório de Computação Reconfigurável - LCR do Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo;

1.3 Organização do Trabalho

O capítulo 2 apresenta a fundamentação teórica necessária para o desenvolvimento deste trabalho. Os assuntos abordados são referentes a controladores PID, histogramas, tecnologia de câmeras e *hardware* reconfigurável. O capítulo 3 descreve os recursos de *hardware* e *software* que são utilizados para a implementação e validação do sistema proposto. O capítulo 4 apresenta os módulos de análise de histograma e controle de ganho projetados para o trabalho e o capítulo 5 apresenta os resultados obtidos. No final são apresentadas as conclusões e sugestões para trabalhos futuros.

Capítulo

2

Fundamentação Teórica

Este capítulo apresenta, de maneira resumida, a fundamentação teórica dos temas predominantes deste projeto.

Os temas abordados são: controlador PID (Proporcional, Integrador, Derivativo), que é utilizado para o controle de ganho do sensor; histograma; tecnologia de sensores CMOS *(Complementary Metal Oxide Semiconductor)*; tecnologia de *hardware* reconfigurável, com ênfase em dispositivos FPGA *(Field Programmable Gate Array)*, sendo esta a tecnologia de *hardware* adotada para a implementação deste trabalho.

2.1 Sistemas de Controle

Um sistema de controle é um conjunto de componentes que atuam conjuntamente e realizam comandos ou regulam um processo de forma a atingir uma saída desejada. Um sistema de controle é formado por três componentes principais: sensores, atuadores e um controlador. Os sensores medem as variváeis do processo ou do sistema e proveem informações de realimentação. Em controle digital, os sensores são qualquer componente capaz de converter uma grandeza física em uma tensão, que é então transformada em um valor digital e realimentada ao controlador. O controlador gera comandos que atuam em

resposta às informações recebidas e consiste em elementos computacionais que processam informações afim de atingir uma determinada resposta de todo o sistema. A função do controlador é assegurar que o sistema responda o mais rápido possível e ao mesmo tempo, permaneça estável sob todas as condições de operação. Os atuadores são responsáveis em transformar as respostas do controlador em ações de controle. Podem ser mecânicos, elétricos, eletro-mecânicos, químicos, etc.(Irfan, 1997).

2.1.1 Controladores Analógicos

Sistemas de controle foram inicialmente implementados usando componentes analógicos como amplificadores operacionais, resistore e capacitores. A combinação destes elementos implementam estruturas semelhantes a filtros que modificam a frequência de resposta do sistema. Embora elementos de processamento analógico mais poderosos estejam disponíveis, como multiplicadores, eles não são geralmente usados devido ao seu elevado custo. Apesar de possuir elementos mais simples, controladores analógicos podem ser usados em sistemas de alta performance.

2.1.2 Controladores Digitais

Com a performance e a confiabilidade inerente aos componentes digitais utilizados, como microcontroladores, microprocessadores, DSPs e FPGAs, controladores digitais estão cada vez mais presentes em nosso meio. Em um sistema de controle digital, o controlador é implementado em um circuito integrado que é responsável pelo processamento de sinal do sistema. No entanto os comandos do controlador são digitais e necessitam ser convertidos em sinais analógicos por um conversor Digital-Analógico. Da mesma maneira, as medidas dos sensores necessitam ser convertidas para um valor digital por um conversor Analógico-Digital. A figura 2.1 mostra o diagrama de blocos de um sistema de controle digital. O conversor D/A converte a saída digital do Controlador Digital, u(n), em um sinal analógico, u(t). A saída u(t), do conversor D/A necessita na maioria das vezes de amplificação de potência para levar a planta até o ponto de referêcia, r(n). A saída da Planta, y(t), é medida através de um sensor e convertida em um sinal digital, y(n), pelo conversor A/D. O sinal de realimentação é então subtraído do sinal de referência r(n) para

criar o sinal de erro e(n). O sinal de erro, e(n), é usado pelo controlador para calcular a correspondente ação de controle u(n).

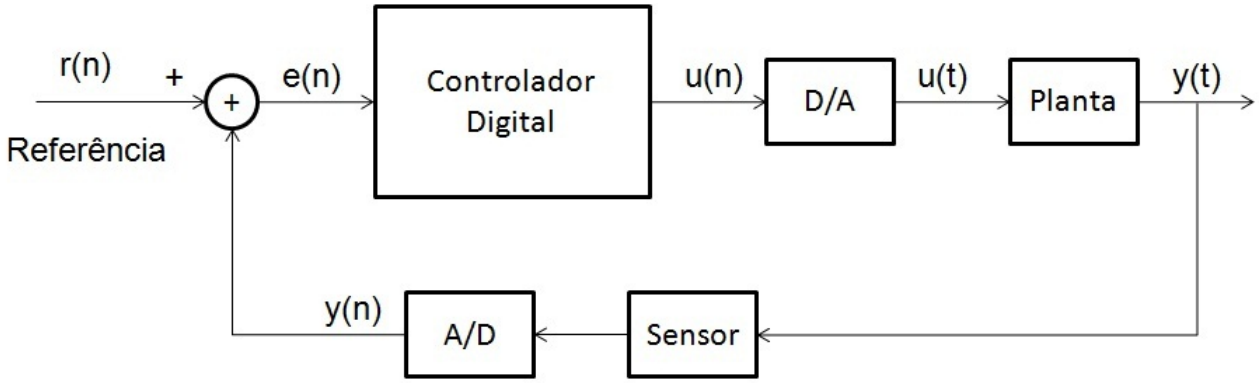

Figura 2.1: Sistema de Controle Digital, adaptada de (Ahmed, 1997).

2.2 Controlador PID

2.2.1 Histórico

Nos últimos 70 anos controladores PID tem sido amplamente utilizados. Isto esta documentado no guia de referencia Muhidin e Zoran (L. Muhidin, 2000). A pesquisa e desenvolvimento de aplicações usando este tipo de controlador mostra uma dinâmica muito vasta. É estimado que na década passada foram publicados mais de 300 papers em jornais sobre teoria e aplicações de controladores PID.

A literatura sobre controladores PID e sua regulação é muito ampla. Uma breve referência aos desenvolvimentos feitos nas décadas recentes pode ser o trabalho de Morari e Zafíriou (M. Morari, 1989), onde os autores mostram que as tecnologias que usam Controle de Modelo Interno (IMC) levam ao controlador PID para a maioria dos modelos dos processos. Um controlador IMC PID equivalente pode ser conseguido através do uso da resposta em freqüência e do algoritmo dos últimos quadrados; isto foi provado por Wang ET AL (Q. G. Wang, 2000). Em 2001, Katebi e Moradi (M. R. Katebi, 2001) desenvolveram um controlador PID preditivo para processos SISO baseado no método de controle preditivo generalizado (GPC), a generalizacao para processos MIMO com representacoes polinomiais, foi feita em 2002 (M. H. Moradi, 2002). O desenvolvimentos desta tecnica foi realizada por Moradi e Johnson (M. H. Moradi, 2010). O uso de computação recon-

figurável no auto ajuste de controladores PID tem sido investigado com sucesso (Lima, 2004). Especificamente, podemos apontar a habilidade de redes neurais para modelar plantas, juntamente com algoritmos genéticos para a otimização dos critérios de controle.

Apesar de novos métodos específicos de regulação do controlador PID terem sido apresentados, para problemas específicos, técnicas clássicas como Ziegler and Nichols, 1942(J. G. Ziegler, 1942), ainda são amplamente utilizadas em uma grande gama de aplicações. Este método também é apontado como referência, por exemplo, pelo método desenvolvido em 1952 por Cohen e Coon, ou no método de feedback de Aström e Hägglund. Os algoritmos desenvolvidos usam tanto respostas temporais quanto resposta em freqüência do processo a ser controlado. Estas técnicas são freqüentemente usadas como referência para comparação de metodologias recentes.

2.2.2 Conceito

Um controlador típico PID paralelo, contínuo e de malha fechada é mostrado na figura 2.2

Os sinais r, e, u, d e y representam a referência r, o erro e, o sinal de controle u, o sinal de distúrbio d, e a saída do sistema y, respectivamente.

O bloco G(s) representa o sistema que queremos controlar.

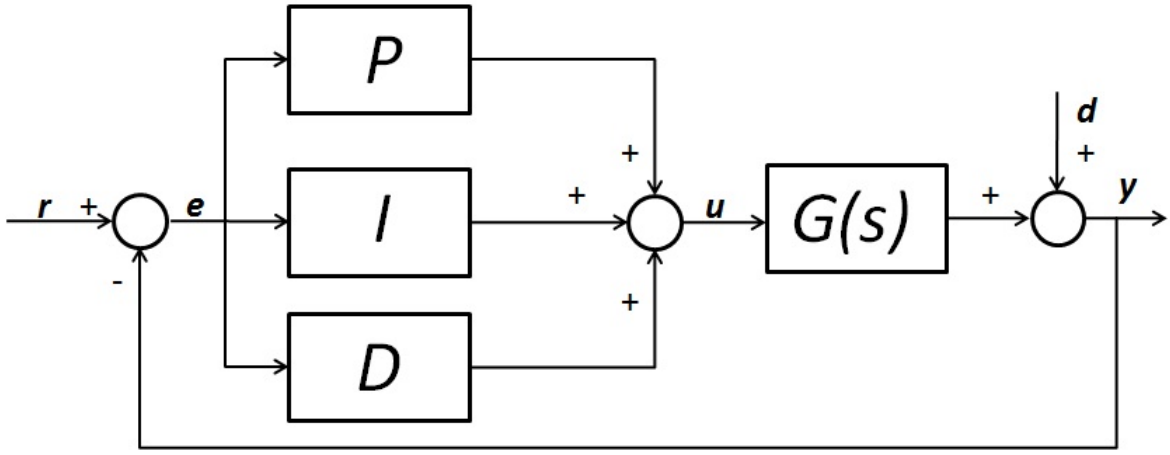

Figura 2.2: Controlador PID contínuo, paralelo de malha fechada (Åström e Hägglund, 2006)

A lei de controle do controlador PID é definida em termos de 3 parâmetros, o ganho proporcional k_c o tempo integral t_i e o tempo derivativo t_d. No domínio do tempo, o

controlador PID pode ser descrito como a relação entre o sinal de controle e os sinais de erro, como mostra a equação 2.1

$$u(t) = k_c e(t) + \frac{1}{t_i}\int_0^t e(t)d(t) + t_d \frac{d}{dt} e\,(t) \tag{2.1}$$

2.2.3 Termo Proporcional

O termo proporcional faz uma alteração na saída que é proporcional ao valor de erro atual. A resposta proporcional é ajustada, multiplicando-se o erro $e(t)$ pela constante k_c, chamada de ganho proporcional.

O termo proporcional é dado por 2.2

$$k_c.e\,(t) \tag{2.2}$$

A alta do ganho proporcional resulta em uma grande mudança na saída para uma dada mudança no erro. Se o ganho proporcional é muito alto, o sistema pode se tornar instável. Em contraste, se o ganho proporcional é muito baixo, a ação de controle pode ser pequena ao responder a perturbações do sistema. Na maioria dos controladores existentes, o termo proporcional contribui com a maior parte da mudança de saída.

A figura 2.3 mostra a resposta de um sistema controlado por um controlador PID onde somente o ganho proporcional é variado. A referência representa o ponto de ajuste desejado para o sistema. As outras três curvas do gráfico representam a saída do sistema. Pode ser observado, que a medida em que o ganho aumenta, há uma diminuição do tempo de resposta e uma redução do erro de regime estacionário. Nota-se também, que dependendo do aumento no ganho, pode haver a formação de sobre-sinal.

2.2.4 Termo Integral

A contribuição do termo integral é proporcional à magnitude do erro e da duração do erro. A integral em um controlador PID é a soma do erro instantânea ao longo do tempo e da compensação acumulada que deveria ter sido corrigido anteriormente. O erro acumulado é então multiplicado pelo ganho integral $1/t_i$ e acrescentada à saída do

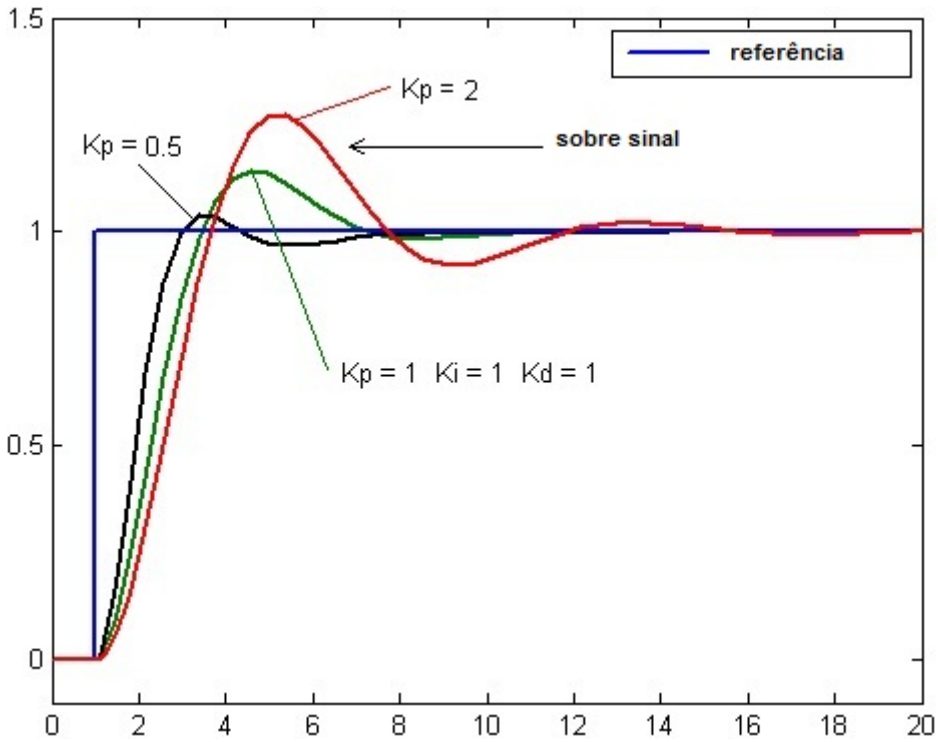

Figura 2.3: Resposta de um controlador PID para variações de ganho proporcional Kp

controlador.

O termo integral é dada por 2.3

$$\frac{1}{t_i}\int_0^t e(t)d(t) \tag{2.3}$$

O termo integral acelera o movimento do processo no sentido de setpoint e elimina o erro de estado estacionário residual que ocorre com um controlador proporcional puro. No entanto, uma vez que o termo integral responde a erros acumulados do passado, pode levar o valor presente a ultrapassar o valor de referência.

A figura 2.4 mostra a resposta de um sistema controlado por um controlador PID onde somente o ganho integral é variado. O *setpoint* representa o ponto de ajuste desejado para o sistema. As outras três curvas do gráfico representam a saída do sistema. Pode ser observado, que a medida em que o ganho integral aumenta, há uma diminuição do tempo de resposta e uma redução do erro de regime estacionário.Quando o sensor de um sistema é susceptível a ruídos ou à interferência elétrica, ações de controle derivativas podem causar grandes flutuações de saída do sistema. Nestas circunstâncias, a melhor escolha de controlador seria o do tipo PI (proporcional mais integral) (Giorgini, 2001).

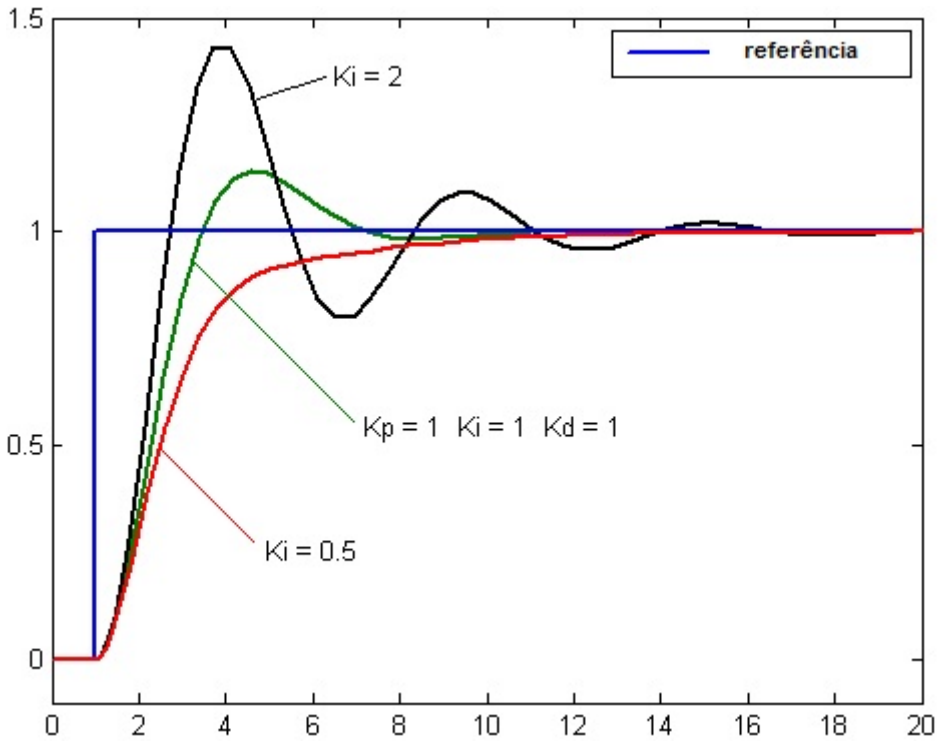

Figura 2.4: Resposta de um controlador PID para variações de ganho Integral Ki

2.2.5 Termo Derivativo

O termo derivativo do controlador é calculado através da multiplicação do ganho derivativo t_d pela taxa de declividade do erro ao longo do tempo.

O termo derivativo é dado por 2.4

$$t_d \frac{d}{dt} e\left(t\right) \tag{2.4}$$

O termo derivado diminui a taxa de variação da saída do controlador. Controle derivativo é usado para reduzir a magnitude do sobre sinal produzida pelo componente integral e melhorar a estabilidade do controlador de processo combinado. No entanto, como podemos observar na figura 2.5 o termo derivado retarda a resposta transitória do controlador. Além disso, a diferenciação de um sinal amplifica o ruído e, portanto, este termo no controlador é altamente sensível ao ruído no sinal de erro, e pode causar um processo para tornar-se instável se o ruído e o ganho derivativo forem suficientemente grandes. Por isso uma aproximação para um diferencial com uma largura de banda limitada é mais comumente usado.

Os problemas de estabilidade e de sobre-sinal apresentados por controladores propor-

cionais com alto ganho, podem ser minimizados com a adição de um termo derivativo.

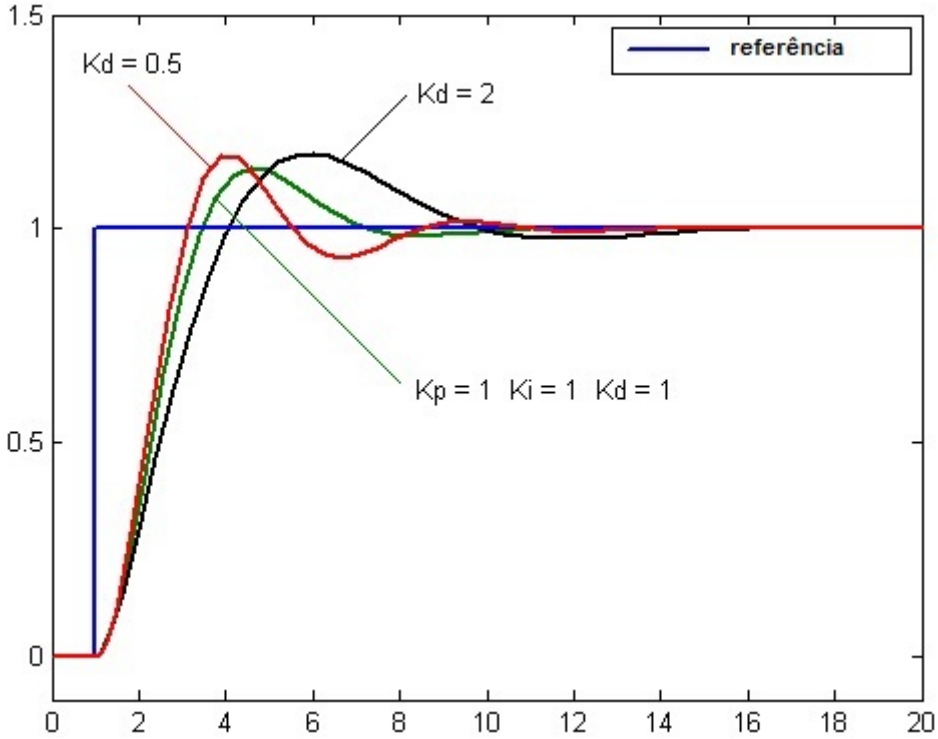

Figura 2.5: Resposta de um controlador PID para variações de ganho Derivativo Kd

2.2.6 Controlador PID Digital

Como ja mostrado, o controlador PID pode ser descrito como a relação entre o sinal de controle e os sinais de erro 2.5

$$u(t) = k_c e(t) + \frac{1}{t_i}\int_0^t e(t)d(t) + t_d \frac{d}{dt} e\,(t) \tag{2.5}$$

Para convertermos o controlador para sua forma discreta, afim de ser implementado em um sistema digital, o termo integral é aproximado por 2.6

$$\int edt = \sum e_i T \tag{2.6}$$

Fazendo o intervalo de amostragem pequeno suficiente podemos aproximar o termo diferencial por 2.7 onde e(n) e e(n-1) são os valores do sinal de erro para os intervalos de tempo n e n-1.

$$\frac{de}{dt} = \frac{e(n) - e(n-1))}{T} \tag{2.7}$$

Para tempos de amostragem pequenos *t0* a equação 2.5 pode ser expressa de forma discreta por equações diferenciais. O termo derivativo é simplesmente substituído pela equação diferença de primeira ordem e o termo integral é substituído por uma soma. A integração contínua pode ser discretizada tanto por aproximação retangular quanto por aproximação trapezoidal.

Integrando-se por aproximação retangular tem-se a equação 2.8:

$$u(n) = k_c[e(n) + \frac{t_0}{t_i}\sum_{i=0}^{n-1} e(i) + \frac{t_d}{t_0}(e(n) - e(n-1))] \tag{2.8}$$

Esta equação representa o algoritmo não recursivo de controle. Para a programação do controlador PID um algoritmo recursivo seria mais apropriado. O algoritmo recursivo se caracteriza por calcular a variável atual, *u(n)*, baseada na variável calculada no tempo anterior, *u(n-1)* mais os termos de correção (ISERMANN, 1989).

Para se derivar a equação recursiva deve-se subtrair 2.9 de 2.8 representada a seguir.

$$u(n-1) = k_c[e(n-1) + \frac{t_0}{t_i}\sum_{i=0}^{n-2} e(i) + \frac{t_d}{t_0}(e(n-1) - e(n-2))] \tag{2.9}$$

Como resultado obtém-se 2.10 para o algoritmo recursivo do controlador PID.

$$u(n) = u(n-1) + a_0 e(n) + a_1 e(n-1) + a_2 e(n-2) \tag{2.10}$$

onde

$$a_0 = K_c(1 + \frac{t_d}{t_s}) \tag{2.11}$$

$$a_1 = -K_c(1 + 2\frac{t_d}{t_s} - \frac{t_s}{t_i}) \tag{2.12}$$

$$a_2 = -K_c\frac{t_d}{t_s} \tag{2.13}$$

Os parêmetros *Kc*, *ti* e *td* são os parâmetros para ajuste do controlador PID, e *t0* é o período de amostragem em segundos (K. Åström, 1995).

2.2.7 Ajuste do controlador PID

Os métodos de avaliação dos parâmetros de ganho proporcional kc, do tempo integral ti e do tempo derivativo td são chamados métodos de ajuste do controlador PID

A sintonia do controlador PID digital pode ser feita aplicando técnicas para sistemas contínuos e discretizando os blocos, tendo em conta um determinado período de amostragem ou o tempo discreto de processamento do controlador.

A implementação de controladores digitais em FPGA permite atingir alguns requisitos como desempenho, baixo custo e facilidade de reconfiguração, que ainda não foram observados em DSPs (Processador Digital de Sinais). A abordagem endereça a solução de um tipo de SoC (Sistema em chip)(Åström e Hägglund, 2006).

Existem várias classes de ajustes de controladores PID, como Ziegler-Nichols, Cohen-Coon, etc. No desenvolvimento deste trabalho o controlador foi ajustado empíricamente, (tentativa e erro). O ajuste do controlador através de algum método já consagrado, pode ser alvo de um trabalho futuro.

2.3 Histograma

2.3.1 Conceito

Compreender o que são, como funcionam e se familiarizar com os histogramas são provavelmente os passos mais importantes para trabalhar com imagens de uma câmera digital. Um histograma pode dizer se a imagem foi exposta corretamente, se o tipo de luz era dura ou suave e quais ajustes funcionam melhor em sua câmera. Esse conhecimento é essencial para se trabalhar com imagens digitais.

O histograma é possivelmente a ferramenta mais importante disponível para trabalhar com imagem digital.

A figura 2.6 mostra uma classificação de níveis de brilho de uma imagem. Esta é a mesma classificação de um histograma onde os valores de conversão analógico digital são de 8 bits para uma imagem com cinco níveis de cinza. O valor máximo que um pixel pode atingir é 255, que indica um valor de alto brilho, e este é classificado no lado mais

a direita do histograma. Valores de baixo brilho (escuros) são colocados à esquerda do histograma, indicando baixos valores do conversor analógico digital para aquele pixel. O número de divisões é determinado pelo uso do histograma, e pelo número de bits do conversor analógico digital do sensor. Um histograma de um sensor que utiliza 8 bits pode ter ao máximo 255 faixas de valores, onde cada faixa indica um valor do conversor analógico digital para um dado pixel.

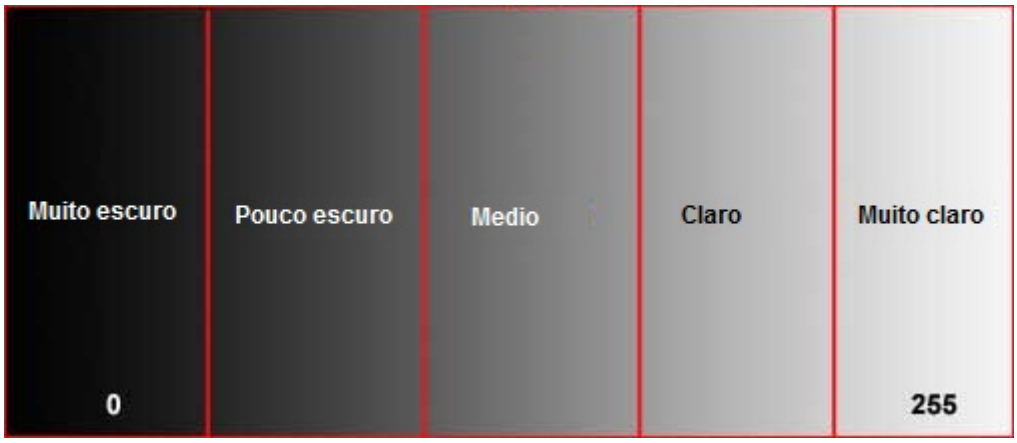

Figura 2.6: Exemplo de valores de conversão A/D de 8 bits para imagem com cinco níveis de cinza

O histograma de uma imagem é simplesmente um conjunto de números indicando o percentual de pixels naquela imagem que apresentam um determinado nível de cinza. Estes valores são normalmente representados por um gráfico de barras que fornece para cada nível de cinza o número (ou o percentual) de pixels correspondentes na imagem. Através da visualização do histograma de uma imagem obtemos uma indicação de sua qualidade quanto ao nível de contraste e quanto ao seu brilho médio (se a imagem é predominantemente clara ou escura). Cada elemento deste conjunto é calculado como na equação 2.14:

$$p_r(r_k) = \frac{n_k}{n} \tag{2.14}$$

onde:

$0 \leq rk \leq 1$

$k = 0, 1, ..., L\text{-}1,$ onde L é o número de níveis de cinza da imagem digitalizada;

n = número total de pixels na imagem;

$pr(rk)$ = probabilidade do k-ésimo nível de cinza;

nk = número de pixels cujo nível de cinza corresponde a k.

Os dados da tabela 2.1 correspondem a uma imagem de 128 x 128 pixels, com 8 níveis de cinza. O número de pixels correspondentes a um certo tom de cinza está indicado na segunda coluna, enquanto as respectivas probabilidades pr(rk) aparecem na terceira coluna. A representação gráfica equivalente deste histograma é mostrada na figura2.7. Um histograma apresenta várias características importantes. A primeira delas é que cada pr(rk) fornece, como sugere a notação, a probabilidade de um pixel da imagem apresentar nível de cinza rk. Portanto, um histograma nada mais é que uma função de distribuição de probabilidades e como tal deve obedecer aos axiomas e teoremas da teoria de probabilidade. Por exemplo, é possível verificar que na tabela 2.1 a soma dos valores de pr(rk) é 1, o que já era esperado (Filho e Neto, 1999).

Nível de cinza *(rk)*	nk	pr(rk)
0	1120	0,068
1/7	3214	0,196
2/7	4850	0,296
3/7	3425	0,209
4/7	1995	0,122
5/7	784	0,048
6/7	541	0,033
1	455	0,028
Total	16384	1

Tabela 2.1: Exemplo de histograma (Filho e Neto, 1999).

A figura 2.8 apresenta cinco exemplos de tipos de histogramas freqüentemente encontrados em imagens. O histograma (a) da figura 2.8 apresenta grande concentração de pixels nos valores mais baixos de cinza, correspondendo a uma imagem predominantemente escura. No histograma (b) da figura 2.8 os pixels estão concentrados em valores próximos ao limite superior da escala de cinza, caracterizando uma imagem clara. Na parte (c) da figura 2.8, os pixels estão agrupados em torno de valores intermediários de cinza, correspondendo a uma imagem de brilho médio. Na figura 2.8 nos histogramas (a), (b) e (c) a maioria dos pixels está concentrada em uma estreita faixa da escala de cinza, significando que as imagens correspondentes apresentam baixo contraste. A figura 2.8 (d) corresponde a uma imagem com pixels distribuídos ao longo de toda a escala de cinza. É comum dizer que uma imagem com estas características apresenta um bom contraste.

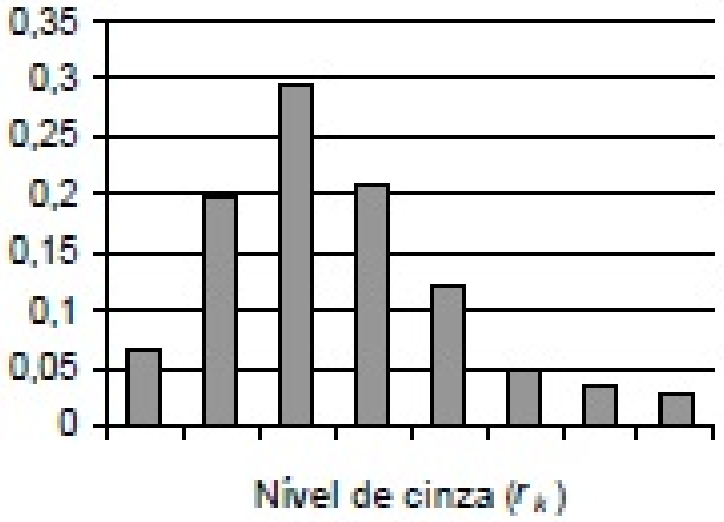

Figura 2.7: Exemplo de histograma para imagem com oito níveis de cinza (Filho e Neto, 1999)

A figura 2.8 (e) mostra um histograma tipicamente bimodal, isto é, apresentando duas concentrações de pixels, uma delas em torno de valores escuros e outra na região clara do histograma. Pode-se dizer que a imagem correspondente apresenta alto contraste entre as duas concentrações, uma vez que elas se encontram razoavelmente espaçadas. Para verificar a relação entre imagens e respectivos histogramas, a figura 2.9 mostra cinco imagens monocromáticas cujos histogramas são aqueles da figura 2.8.

O conceito de histograma também é aplicável a imagens coloridas. Neste caso, a imagem é decomposta de alguma forma (por exemplo, em seus componentes R, G e B) e para cada componente é calculado o histograma correspondente.

Para computar o histograma de uma imagem monocromática, inicializa-se com zero todos os elementos de um vetor de L elementos, onde L é o número de tons de cinza possíveis. Em seguida, percorre-se a imagem, pixel a pixel, e incrementa-se a posição do vetor cujo índice corresponde ao tom de cinza do pixel visitado. Após toda a imagem ter sido percorrida, cada elemento do vetor conterá o número de pixels cujo tom de cinza equivale ao índice do elemento. Estes valores poderão ser normalizados, dividindo cada um deles pelo total de pixels na imagem.

Convém enfatizar que, embora o histograma de uma imagem forneça diversas informações qualitativas e quantitativas sobre ela (e.g. nível de cinza mínimo, médio e

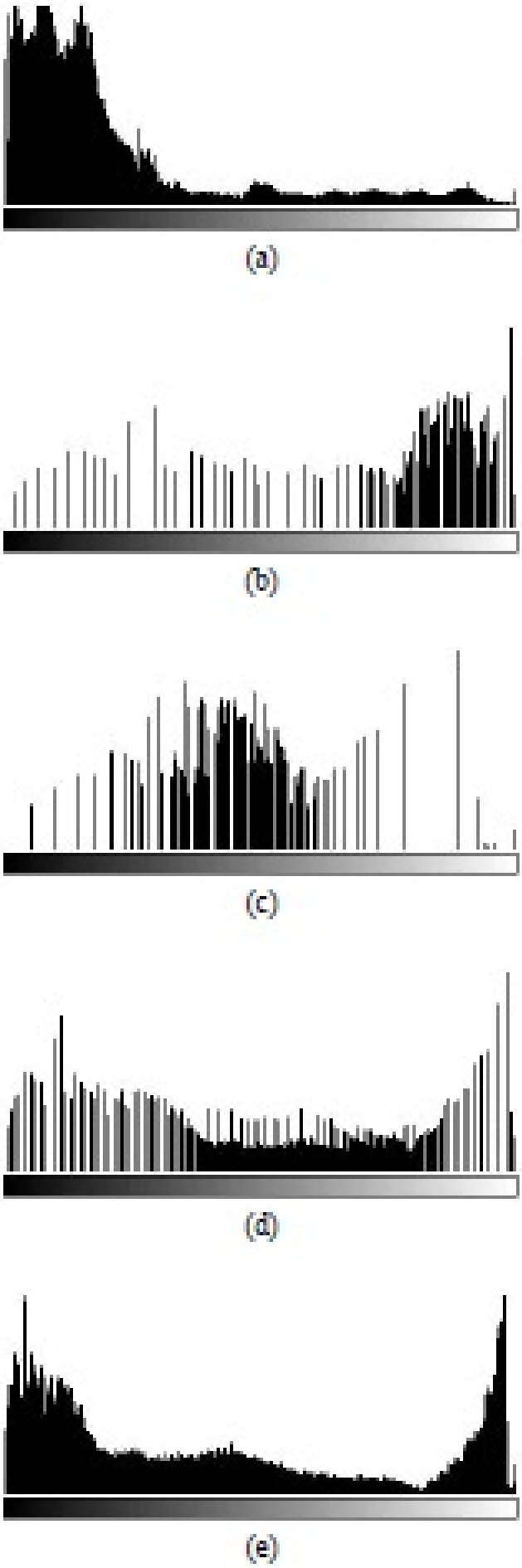

Figura 2.8: Exemplo de histogramas (Filho e Neto, 1999)

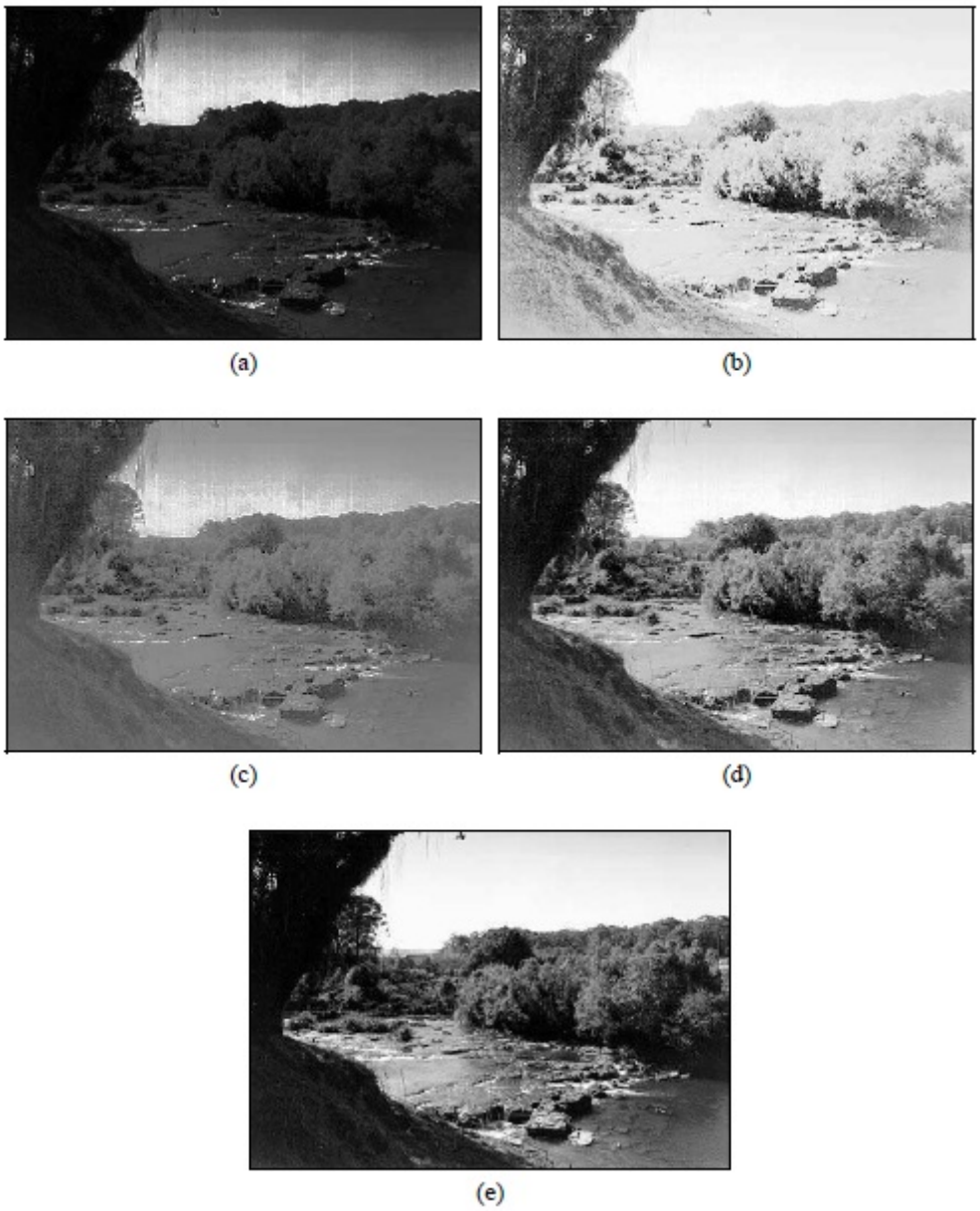

Figura 2.9: Imagens correspondentes aos histogramas da figura 2.8 (Filho e Neto, 1999)

máximo, predominância de pixels claros ou escuros etc.), outras conclusões de caráter qualitativo (e.g. qualidade subjetiva global da imagem, presença ou não de ruído etc.) somente podem ser extraídas dispondo-se da imagem propriamente dita. (Filho e Neto, 1999)

2.3.2 Especificação direta de histograma

Vamos entrar mais detalhadamente neste tópico, pois o sistema proposto baseia-se na análise direta do histograma da imagem, para determinar o erro do controlador e consequentemente o ganho do sensor CMOS. A teoria deste tópico é ligada ao trabalho e maiores detalhes serão fornecidos neste capítulo. O resultado final de avaliação do controle de ganho realizado, tem como base parâmetros do histograma.

Apesar de sua grande utilização em situações de aprimoramento de contraste de imagens, a equalização de histograma apresenta como principal limitação o fato de não permitir a especificação de nenhum parâmetro, a não ser a função de transformação, que, como vimos na Seção anterior, costuma ser a cdf da distribuição de probabilidade original. Existem situações, entretanto, em que seria desejável poder especificar que tipo de mudança se deseja sobre o histograma. Nestes casos, uma das possíveis técnicas é a especificação direta de histograma. Dada uma imagem (e seu histograma original) e o novo histograma desejado, o procedimento da especificação direta de histograma consiste em:

1) equalizar os níveis da imagem original usando a *cdf* discreta:

$$S_k = T(r_k) = \sum_{j=0}^{k} \frac{n_j}{n} = \sum_{j=0}^{k} p_r(r_j) \tag{2.15}$$

2) equalizar a função densidade de probabilidade discreta (isto é, o histograma) desejada(o):

$$v_k = G(z_k) = \sum_{j=0}^{k} p_z(z_j) \tag{2.16}$$

3) aplicar a função de transformação inversa aos níveis obtidos no passo 1:

$$Z = G^{-1}(s) \tag{2.17}$$

No trabalho desenvolvido, o objetivo do controlador é obter um histograma cuja distribuição seja equalizada.

2.4 Tecnologia de Sensores de Imagem Digital

2.4.1 Introdução

Em visão biológica, a retina é o sensor de imagem. A retina não apenas detecta sinais de imagem, mas também processa tarefas limitadas de baixo nível, com o intuito de melhorar a performance e o processamento de tarefas de outros estágios. Baseado em sistemas biológicos, a integração de processamento e sensoriamento para aumentar a performance do sistema é factível. Na figura 2.10 podemos ver a curva característica do diâmetro da pupila humana, em mm,pela luminância, dada em log de cândela por metro quadrado. Esta pode ser considerada a curva de controle do sistema de controle do nossso sensor de visão. Tradicionalmente, um sistema de processamento de visão é composto por módulos discretos; um sensor de imagem (normalmente uma âmera com sensor CCD ou CMOS), um sistema de conversão Analógico/Digital e um sistema de processamento digital.

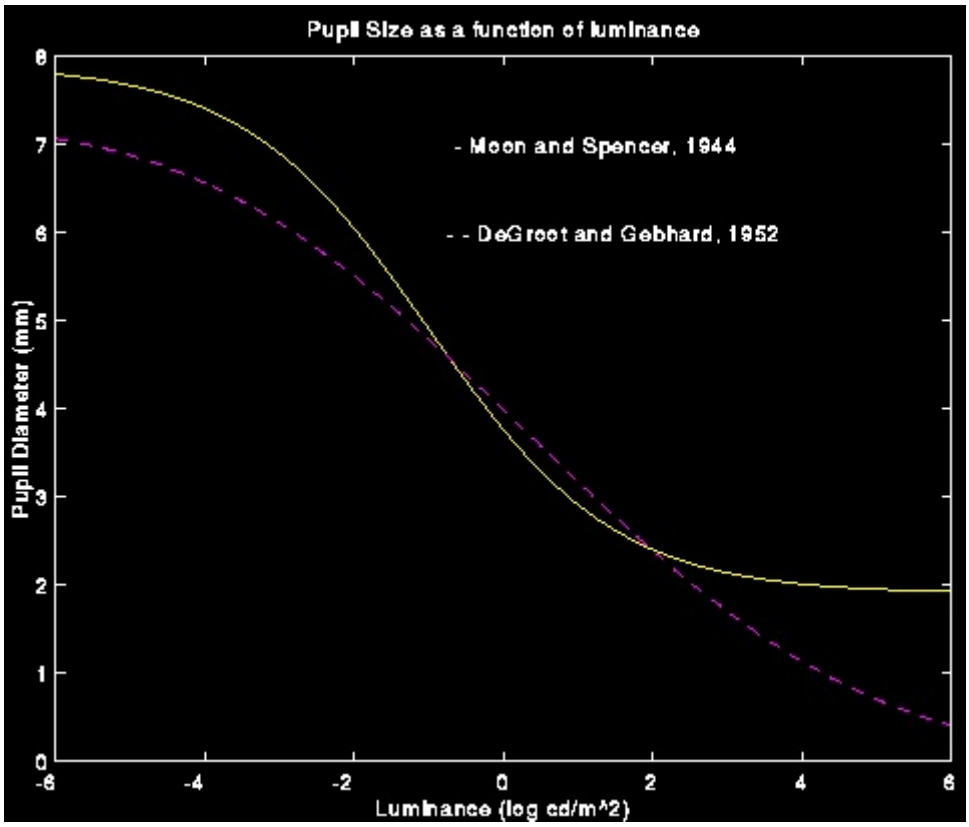

Figura 2.10: Gráfico: Diâmetro da Pupila em Função da Luminância (Moon e Spencer, 1944)

Sensores computacionais (sensores de visão em outras palavras) são sistemas integrados muito pequenos, nos quais sensoriamento e processamento são unificados em um único circuito integrado *VLSI* (Integração de larga escala). Este circuito é projetado para uma aplicação específica, e muitas pesquisas tem sido realizadas sobre estes sensores. Existem várias propostas e implementações de sensores computacionais. Primeiramente, suas qualidades são sumarizadas a partir de vários pontos de vista, como vantagens vs. desvantagens, neurais vs. funcionais, tipo de arquitetura, analógico vs. digital, processamento global vs local, processamento vs. tamanho, e alguns novos paradigmas. Existem vários exemplos, entre eles processadores espaciais, temporais, conversores Analógico-Digital e sensores computacionais programáveis (KlYOHARU AlZAWA, 2004).

O desenvolvimento de tecnologias *VLSI* (Integração de larga escala) tem levado a produção de uma nova geração de sensores de imagem que integram processamento e sensoriamento em um único chip, e são chamados de sensores computacionais, sensores inteligentes ou circuitos de visão. Sensores computacionais são projetados para aplicações específicas.

Sensoriamento e processamento possuem interações de baixo nível. A maior vantagem dos sensores computacionais são seu tamanho e sua velocidade. Diferentemente do sistemas convencionais de processamento de imagens que são compostos por sistemas modulares e seguem o paradgima sensoriamento-processamento, a integração sensoriamento-processamento realiza uma integração em um *system on a chip* (Sistema dentro de um Circuito Integrado). A velocidade não é mais limitada pela taxa de video padrão, o que traz um impacto significante em aplicações de imagem em tempo real. Por exemplo, vamos citar as aplicações de estimação de movimento, que requerem um processamento pesado e que são limitadas a taxa convencional de transmissão de 30Hz, que é o gargalo da comunicação entre o sensor e o processador. Por causa da complexidade da estimação de movimento, que é determinada pelas dimensões do espaço de busca, uma taxa mais rápida de *frames* pode diminuir a dificuldade da tarefa. Se esta taxa pode ser muito mais alta que a taxa de transmissão de vídeo, a área de procura pode ser muito menor e a estimação de movimento pode se tornar muito mais fácil.

A tecnologia dos sensores de imagem também esta mudando. Até agora sensores

CCDs tem predominado, mas os sensores CMOS com pixels ativos estão emergindo como uma tecnologia competitiva. Embora os sensores CMOS passivos já tenham existido, os senores ativos são relativamente novos (T.Kobayashi e S.Ozawa, 1990) (K.Nohsoh e S.Ozawa, 1993).

A tecnologia dos sensores de imagem CCD e CMOS foi inventada no final dos anos 60, e início dos anos 70. Na época, a performance dos sensores CMOS era limitada pela tecnologia de litografia disponível, o que permitiu que os sensores CCDs dominassem os primeiros 25 anos seguintes. O argumento original para a renovação dos sensores CMOS como competidores dos sensores CCDs foi baseada em algumas idéias:

1 - Os processos de litografia e o controle dos processos de fabricação atingiram níveis que logo permitirão aos sensores CMOS fornecerem uma qualidade de imagem próxima a dos sensores CCD.

2 - A capacidade de integração de funções em um único componente criando uma *camera-on-a-chip* ou um sistema em um único chip.

3 - Baixo consumo de energia.

4 - Sistema de aquisição de imagem com tamanho reduzido, como resultado da integração de componentes e consumo de energia.

5 - A possibilidade de utilizar a mesma linha de produção CMOS para fabricar as funções lógicas principais e a memória de maneira integrada, o que gera uma grande economia de escala para os fabricantes de sensores CMOS.

6 - Outros argumentos que favorecem a tecnologia CMOS incluem a operação com uma única fonte de energia, e a possiblidade de fabricação das regiões dos elementos de leitura juntamente a outros componentes do sistema.

Um diagrama de blocos de um sistema de imagem digital é mostrado na figura 2.11, e um esquemático com as principais características dos senores CCD e CMOS são mostrados na figura 2.12.

Primeiro a cena é focada no sensor através das lentes do sensor. O sensor composto por um vetor bi-dimensional de pixels converte a luz incidente na sua superfície em um vetor de sinais eletrônicos. Estes sinais eletrônicos são lidos do sensor e digitalizados por um conversor analógico-digital (ADC). Um processamento de sinais considerável é

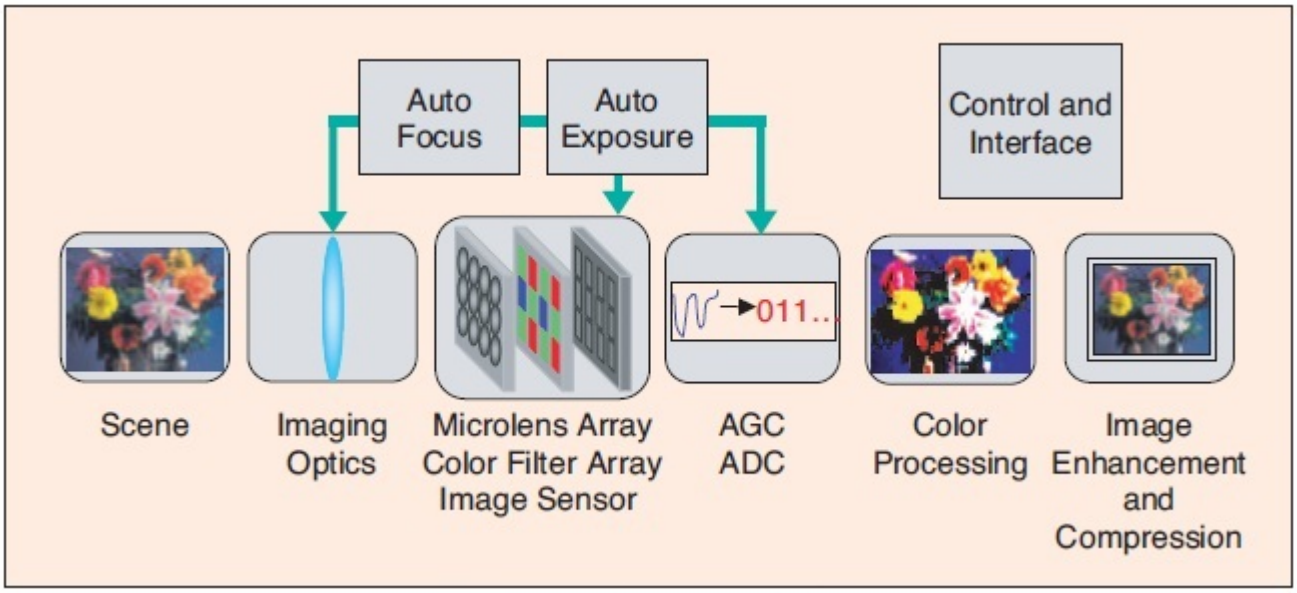

Figura 2.11: Sistema digital de imagem (A. El Gamal, 2005)

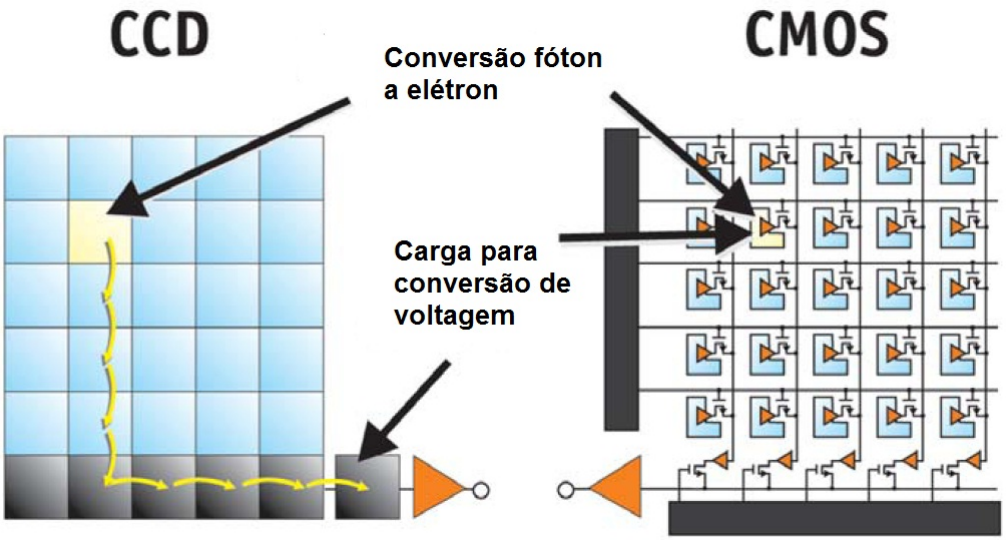

Figura 2.12: Sensores CCD movem a carga gerada pelos fótons de pixel a pixel até convertê-la em voltagem em uma saída. Sensores CMOS convertem a carga em voltagem dentro de cada pixel (Litwiller, 2005)

empregado para processar a cor, a qualidade, e a compressão da imagem. Outros tipos de processamento e controle são empregados para realizar o auto-foco, auto-exposição e controles gerais da câmera. Apesar de cada componente mostrado na figura ter um papel importante, o sensor é o componente chave para determinar o limite de performance final(A. El Gamal, 2005).

Atualmente existem diversos tipos de sensores disponíveis no mercado, sendo que cada qual é caracterizado pela tecnologia que utiliza, consumo de energia, qualidade da imagem, velocidade de captura de *frames*, tamanho, custo e etc.. Dentre os sensores de imagem mais utilizados atualmente estão os de tecnologia CCD e CMOS (Company, 2008a) que são apresentados nas seções seguintes.

2.4.2 Sensor de Imagem CCD

CCD é a abreviação para *charge-coupled device* (dispositivo de carga acoplado). Sensores de imagem CCD são circuitos integrados feitos de silício, e consistem de uma densa matriz de fotodiodos, que operam convertendo energia da luz, na forma de fótons, em carga elétrica (Flora M. Li, 2004).

Como muitas tecnologias, os sensores CCD (Dispositivo de Carga Acoplada) começaram como um tipo de desenvolvimento e terminaram em algo completamente diferente do proposto inicialmente. Inventado no final dos anos 60, por pesquisadores dos Laboratórios Bell, era inicialmente um novo tipo de circuito de memória de computador, e foi apresentado em 1970 para essa finalidade. Logo ficou claro que os dispositivos CCD tinham várias outras aplicações potenciais, incluindo processamento de sinais e imagem. Quanto ao campo de imagens, a sensibilidade à luz do silício, que responde a comprimentos de onda menores que 1.1um (a luz visível fica entre 0.4um e 0.7um). A promessa que o circuito CCD serviria para a construção de memórias desapareceu, mas sua habilidade de detectar a luz tornou a tecnologia CCD a primeira a ser usada em sensores de imagem. De maneira similar ao circuitos integrados (IC), CCDs começaram como finas camadas de silício que são processadas em uma série de processos elaborados, que definem várias funções dentro do circuito. Cada fatia contém vários circuitos integrados (IC) idênticos, cada qual pode ser montado em um dispositivo funcional. Circuitos selecionados, basea-

dos em uma variedade de testes iniciais, são cortados a partir desta fatia e acoplados a outros circuitos ou sistemas (Company, 2008a).

Quando um fóton ultravioleta, visível ou infravermelho chega a um átomo do silício na foto célula (fotodiodo ou pixel) do CCD, produzirá normalmente um elétron livre e um vazio criado pela ausência momentânea do elétron na malha cristalina do silício. O elétron livre mantêm-se então num poço potencial (situado dentro da área do silício conhecida como a camada do depleção), enquanto que o vazio é transferido e eventualmente deslocado para o substrato do silício. Os pixéis são isolados electronicamente dos seus vizinhos por canais de paragem, que se formam por disseminação de ións de Boro através de uma máscara num substrato de silício do tipo p.

Esta ação permite a transmissão dos elétrons gerados através dos registradores até que encontrem o circuito de leitura. O sinal de saida é então transferido para um computador para a geração ou processamento da imagem.

A figura 2.13 mostra a construção típica de um sensor CCD, que consiste em camadas de semi-condutores que são conectadas por uma rede de controle de transferência de sinal, que realiza a transferência de cargas dos pixels para o circuito de saída.

A tecnologia de sensores de imagem CCD foi inventada em 1970 por Willard Boyle e George Smith no laboratório da Bell, nos Estados Unidos (Boyle e Smith, 1970). Deste então, vem sendo utilizada em grande escala em produtos, incluindo máquina de fax, fotocópias, câmeras, scanners e celulares. A figura 2.14 mostra um conjunto sensor CCD mais *hardware* genérico de uma câmera.

O CCD é composto por milhares (ou milhões) de *pixels* ou células sensitivas à luz que são capazes de produzir uma carga elétrica proporcional ao montante da luz que elas recebem. Normalmente, os *pixels* são organizados em uma simples linha ou numa matriz bi-dimensional composta por linhas e colunas. Em scanners, por exemplo, os *pixels* são organizados numa linha, neste caso, é necessário mover o CCD sobre a imagem (ou vice versa). Já as câmeras digitais são formadas por uma matriz de *pixels* bi-dimensional, dessa forma, uma imagem bi-dimensional pode ser capturada numa única revelação.

Um dos principais parâmetros de um CCD é a resolução, que é igual ao número total de *pixels* existentes na área sensitiva à luz do dispositivo. Um dos primeiros dispositivos

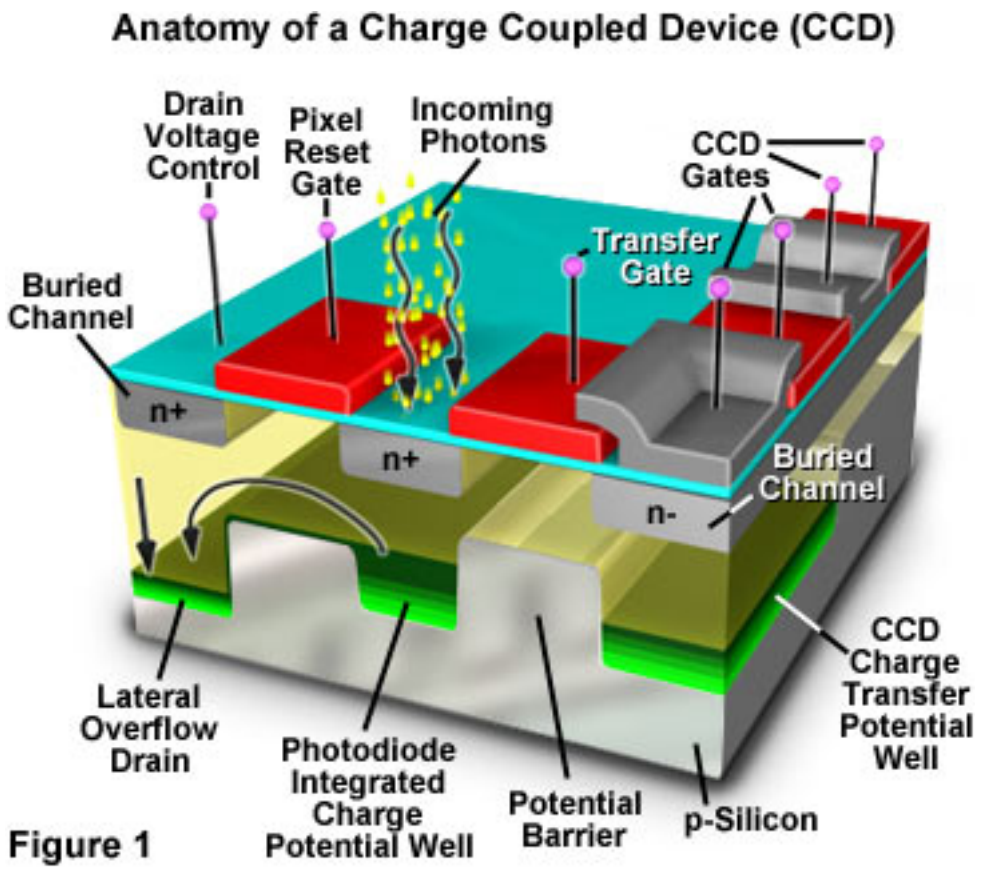

Figura 2.13: Anatomia de um Sensor CCD (www.circuitstoday.com, 2010)

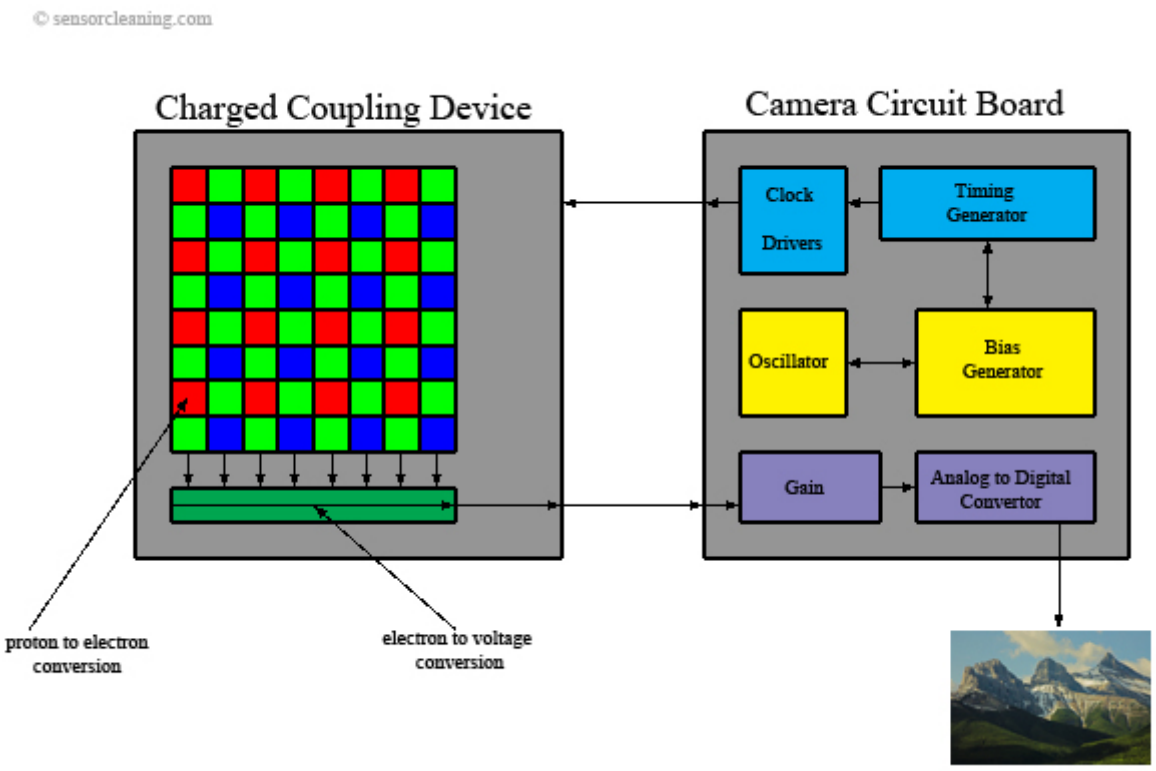

Figura 2.14: Conjunto Sensor CCD mais circuito de câmera (http://sensorcleaning.com, 2010)

CCD de matriz, desenvolvido por Fairchild em 1974, possuía resolução de 100x100 *pixels* (de Oregon, 2003). Atualmente, encontram-se dispositivos com resolução de aproximadamente 9000x7000, ou seja, 63 milhões de *pixels*.

Dispositivos CCD são circuitos integrados que possuem uma janela de vidro que permite a luz chegar até o silício do chip. Os CCD são fabricados utilizando a tecnologia MOS (Metal-Oxide-Semiconductor), onde cada *pixel* pode ser pensado como sendo um capacitor MOS que converte fótons (luz) em carga elétrica (elétron), sendo esta carga armazenada para uma leitura posterior.

2.4.3 Convertendo Fótons em carga elétrica - Efeito Fotoelétrico

Uma imagem é aquisitada quando luz, em forma de fótons, atinge o vetor de pixels. A energia associada a cada fóton é absorvida pelo silício e provoca uma reação. Esta reação leva a criação de um vazio elétrico O funcionamento dos sensores CCD baseia-se no efeito fotoeléctrico. Algumas substâncias têm a propriedade de absorver fótons e libertar no processo um elétron Quando um elétron recebe uma energia luminosa ele se excita e emite um fóton luminoso Na verdade, a oscilação do nível de energia do elétron quando recebe a partícula de luz é que origina a diferença de potencial que gera corrente elétrica nos eletródos do CCD O número de elétrons coletados por cada pixel é linearmente dependente do nível de luz e do tempo de exposição e é não-linearmente dependente do comprimento de onda da luz. Muitos fatores podem afetar a habilidade de detectar um fóton. Filmes finos de materiais depositados intencionalmente no silício durante a fabricação podem criar uma maior tendência a absorvição ou reflexão da luz. Fótons são absorvidos em diferentes profundidades dentro do silício, dependendo de seu comprimento de onda. Existem situações nas quais os elétrons excitados pelos fótons não podem ser detectados por causa da localização em que foram criados dentro do silício. A figura 2.15 mostra a interação dos fótons e seus respectivos comprimentos de onda com o substrato de silício.

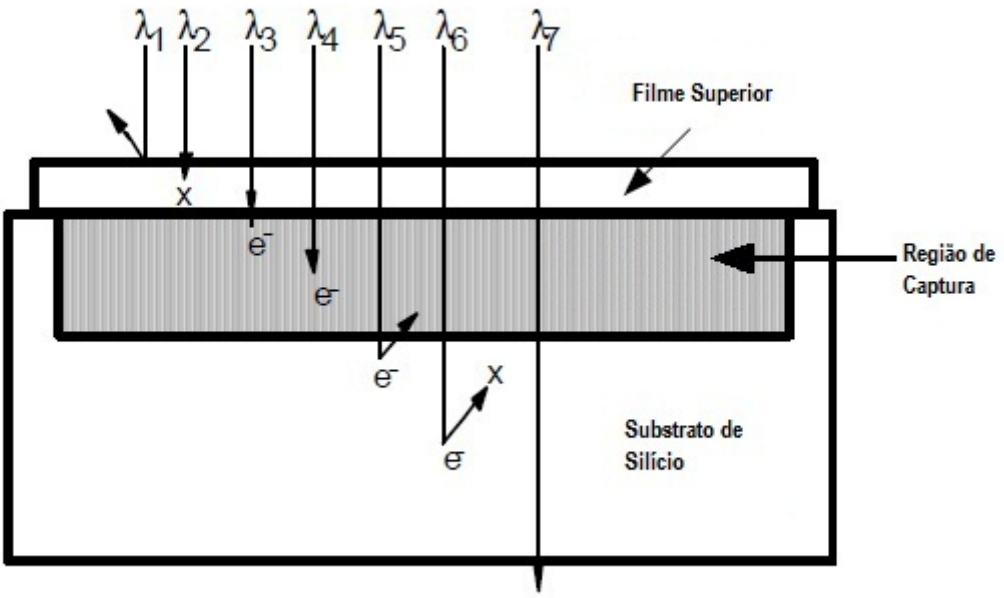

Figura 2.15: Interação Fótons, Comprimento de Onda e Silício (Company, 2008b)

2.4.4 Processo de Leitura de Carga

O processo de leitura de carga é realizado em dois estágios, primeiro a carga do *pixel* é transferida através da matriz de *pixels* e depois a carga do *pixel* é armazenada em um registrador antes de ser digitalizada (Taylor, 1998).

O primeiro processo, a transferência de carga depende do número de *gates* de que cada *pixel* é composto. Assim, as técnicas de transferência são chamadas de *two-phases*(dois *gates*), *three-phases* (três *gates*) ou *four-phases* (quatro *gates*).

Tomando como exemplo o processo *three-phases* (Company, 2008a), visto na figura 2.16, a carga está armazenada no *gate* $\emptyset 1$ enquanto que os *gates* $\emptyset 2$ e $\emptyset 3$ estão no estado bloqueado. O *gate* $\emptyset 2$ é trazido para o nível alto e logo em seguida o *gate* $\emptyset 1$ vai para o nível baixo. Agora, a carga está armazenada no *gate* $\emptyset 2$ e é transferida para o *gate* $\emptyset 3$, manipulando os *gates* $\emptyset 2$ e $\emptyset 3$ da mesma maneira que foi realizada anteriormente. O ciclo de transferência é completado quando a carga é transportada para o *gate* $\emptyset 1$ do próximo *pixel*.

Já o segundo estágio do processo de leitura de carga ocorre após cada linha de *pixels* ter sua carga transferida para a próxima linha. Além das linhas de *pixels*, existe no topo e/ou na base da matriz de *pixels* uma linha adicional chamada de registrador de leitura, que

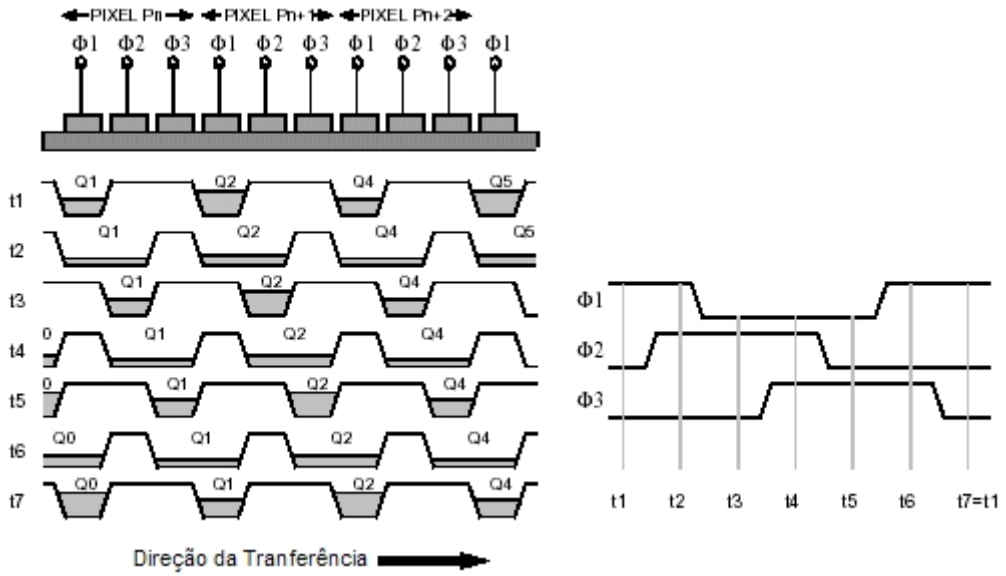

Figura 2.16: Técnica *three-phases* de transferência de carga (Company, 2008b)

armazena as cargas dos *pixels* localizados no topo da matriz quando uma transferência entre linhas é realizada. A carga armazenada no registrador de leitura é enviada para um conversor ADC (Analog to Digital Converter) e posteriormente armazenada numa memória.

2.4.5 *Arquitetura da Matriz de* Pixels *CCD*

Geralmente as arquiteturas dos dispositivos CCD pertencem a uma das quatro categorias: *full frame*, *frame transfer*, *split frame transfer* e *interline transfer* (Taylor, 1998). A seguir estas quatro categorias são descritas e apresentadas através da figura 2.17.

- *Full Frame:* A imagem é transferida diretamente da seção da imagem do sensor ao registrador de leitura. No entanto, somente uma simples linha de *pixels* por vez pode ser transferida para o registrador de leitura, enquanto que o restante dos *pixels* da imagem deve aguardar. Durante este período, os *pixels* que ainda não foram lidos da matriz continuam armazenando informações da imagem, isso pode distorcer e borrar a imagem. Uma solução para este problema é proteger da luz a região de captura da imagem mecanicamente através de uma cobertura (Company, 2008a).

- *Frame Transfer:* O dispositivo com esta arquitetura utiliza uma seção de armazenamento protegida da luz do mesmo tamanho da matriz de *pixels*. Após o tempo de

integração da imagem, a imagem capturada é transferida para a seção adjacente de armazenamento. Enquanto a próxima cena está sendo capturada, a cena capturada anteriormente é transferida para o registrador de saída (paralelismo de operações). Durante a transferência da imagem capturada para a seção de armazenamento, é necessário proteger a matriz de *pixels* da luz para evitar os mesmo problemas citados na *full frame*.

- *Split Frame Transfer:* Este tipo de dispositivo é semelhante ao *frame transfer*, exceto pelo fato de que a seção de armazenamento é dividida em duas partes, sendo que cada parte contém um registrador de saída. Esta arquitetura é vantajosa, pois permite transferir a imagem da seção de armazenamento ao registrador de saída com a metade do tempo gasto na arquitetura *frame transfer*.

- *Interline Transfer:* Nesta arquitetura o dispositivo é composto por colunas de *pixels* e por colunas de registradores protegidos da luz de forma intercalada. No final do tempo de integração, a imagem capturada é transferida rapidamente para as colunas de registradores. Enquanto um novo *frame* é capturado, a carga armazenada nas colunas de registradores são transferidas para os registradores de saída. Este tipo de arquitetura não requer o uso de mecanismos para proteção da luz, visto que a imagem capturada é transferida diretamente às colunas de registradores.

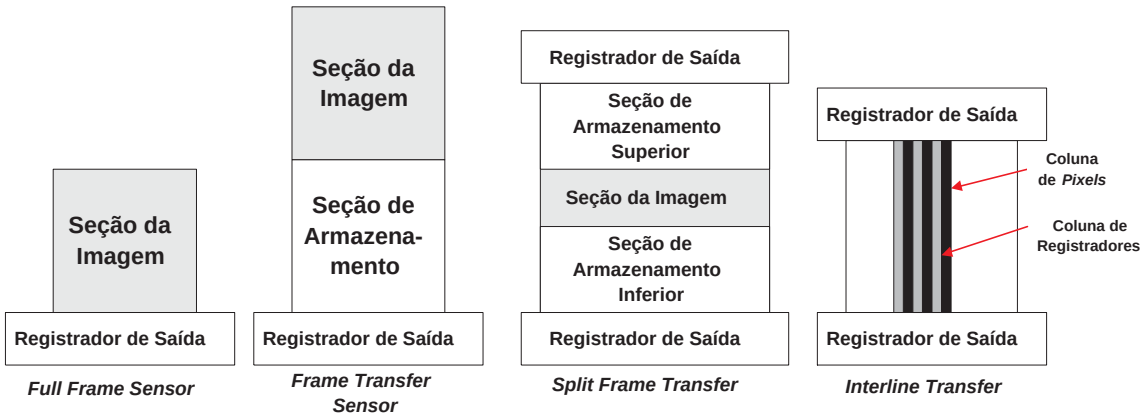

Figura 2.17: Arquitetura de dispositivos CCD (Taylor, 1998)

2.4.6 Sensor de Imagem CMOS

Existem 3 categorias principais de sensores CMOS inteligentes, como mostrado na Figura 2.18: processamento a nível de pixel, processamento a nível de *Chip* e processamento a nível de coluna. A primeira categoria é a dos circuitos de visão integrados ou de processamento paralelo de pixels. Nos anos 1980 foi apresentada e demosntrada uma retina de silício, ou circuito de visão (Ohta, 2008). A retina de silício imita o processamento da visão humana com uma capacidade massiva de processamento paralelo. Desde 1980, um trabalho considerável tem sido feito no desenvolvimento de cirtuitos integrados de visão e equipamentos similares. Um alto nível de processamento paralelo no plano focal é muito atrativo e tem sido assunto de muita pesquisas dentro do campo de retinas artificiais.

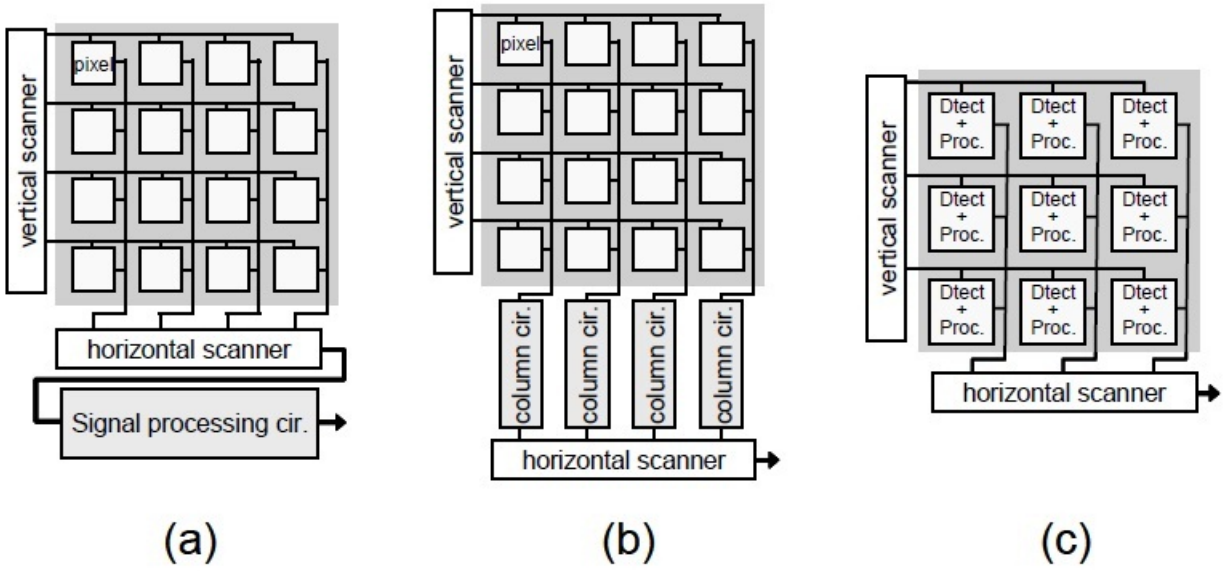

Figura 2.18: Conceito básico dos sensores CMOS inteligentes. (a) Processamento a nível de *Chip* , (b) Processamento a nível de coluna, e (c) processamento a nível de píxel Ohta (2008).

Sensores de imagem CMOS (Semicondutor Metal-Óxido Complementar) tem sido assunto de um intensivo desenvolvimento a atualmente divide mercado com os sensores CCD (Dispositivo de Carga Acoplada) que domiram o mercado por um longo período. Sensores de imagem CMOS são atualmente usados não apenas em eletrônica de consumo, como câmeras, telefones celulares, filmadoras, mas também em câmeras de automóveis, circuitos de segurança, visão robótica, etc. Recentemente, aplicações de sensores CMOS em biotecnologia e medicina tem emergido. Muitas das aplicações que requerem performance,

alta faixa dinâmica, alta velocidade, e alta sensibilidade, enquanto outras necessitam de funções dedicadas como processamento em tempo real e análise tri-dimencional. É difícil executar estas tarefas com sensores de imagem convencionais. Além disso, alguns processamentos de sinais sáo insuficientes para estes propósitos. Sensores de imagem CMOS inteligentes, que possuem funções integradas na sua construção, podem atender a tais requisitos e realizar tais tarefas.

Sensores CMOS são fabricados baseados no padrão de fabricação de integração de larga escala (LSI), enquanto os sensores CCD são baseados em um processo especial de fabricação. Esta característica dos sensores CMOS torna possível a integração de circuitos funcionais para desenvolver sensores de imagem inteligentes e obter uma performance mais alta do que a alcançada pelos sensores CCD e sensores CMOS convencionais.

Senores CMOS inteligentes estão separados em dois campos principais. Um é o de melhoramento das características dos sensores CMOS, como faixa dinâmica, velocidade e sensibilidade. O outro é o dedicado a novas funções como análise tridimensional, rastreamento de alvos e detecção modular de luz. Em ambos os campos, muitas arquiteturas e estruturas, assim como materiais, tem sido propostos. Os termos seguintes também são associados aos sensores CMOS: Sensores computacionais, sensores CMOS com funcionalidades integradas, circuitos de visão, processadores de imagem focal plana, assim como vários outros. Além de circuitos de visão, estes termos sugerem um sensor de imagem que possui outras funções além da aquisição de imagens (Company, 2003).

O primeiro sensor de imagem utilizando a tecnologia CMOS foi produzido pelo laboratório Jet Propulsion Laboratory (JPL) da NASA em 1990 (Technology, 2003).

Os sensores CMOS, bem como CCD, são formados por elementos sensitivos a luz, cada qual capaz de produzir um sinal elétrico ou carga proporcional a luz que incide nestes elementos. No entanto, o processo para realizar isto é bem diferente em cada tecnologia. No CCD o *pixel* é baseado num capacitor MOS, já no CMOS o *pixel* pode ser construído utilizando fotodiodo, transistores, capacitores, ou seja, estes componentes podem ser integrados formando um circuito específico para cada *pixel* (Company, 2003).

Sensores CMOS com pixels ativos possuem varios transistores que funcionam como um amplificador ativo em cada pixel.Por causa da compatibilidade do processo de construção

deste tipo de sensor, ele é facilmente integrado a circuitos adicionais, e pode atingir funcionalidades com baixo consumo de potência. O sensor CMOS foi então integrado a conversores Analógico/Digital, controladores de tempo, etc., o que resultou em uma *on-chip-digitalcamera* (câmera-circuito-integrado). A maior vantagem do sensor CMOS é sua funcionalidade, a maioria dos sensores computacionais tem sido construidos a partir de um processo CMOS comum, e isto pode ser visto como um sensor CMOS funcional. Ate agora, sensores CMOS tem melhorado sua qualidade de imagem e vários produtos tem sido introduzidos no mercado utilizando sua tecnologia. (Por exemplo, câmeras digitais, telefones celulares, etc.). Em um futuro próximo, sua funcionalidade vai ser mais e mais important. Por isso, o interesse neste tipo de sensor e em sua tecnologia de processamento vão convergir. Tanto a tecnologia do sensor e as aplicações de seu processamento interno compartilham um campo de pesquisa comum. (KlYOHARU AlZAWA, 2004)

A figura 2.19 mostra um conjunto sensor CMOS mais *hardware* genérico de uma câmera.

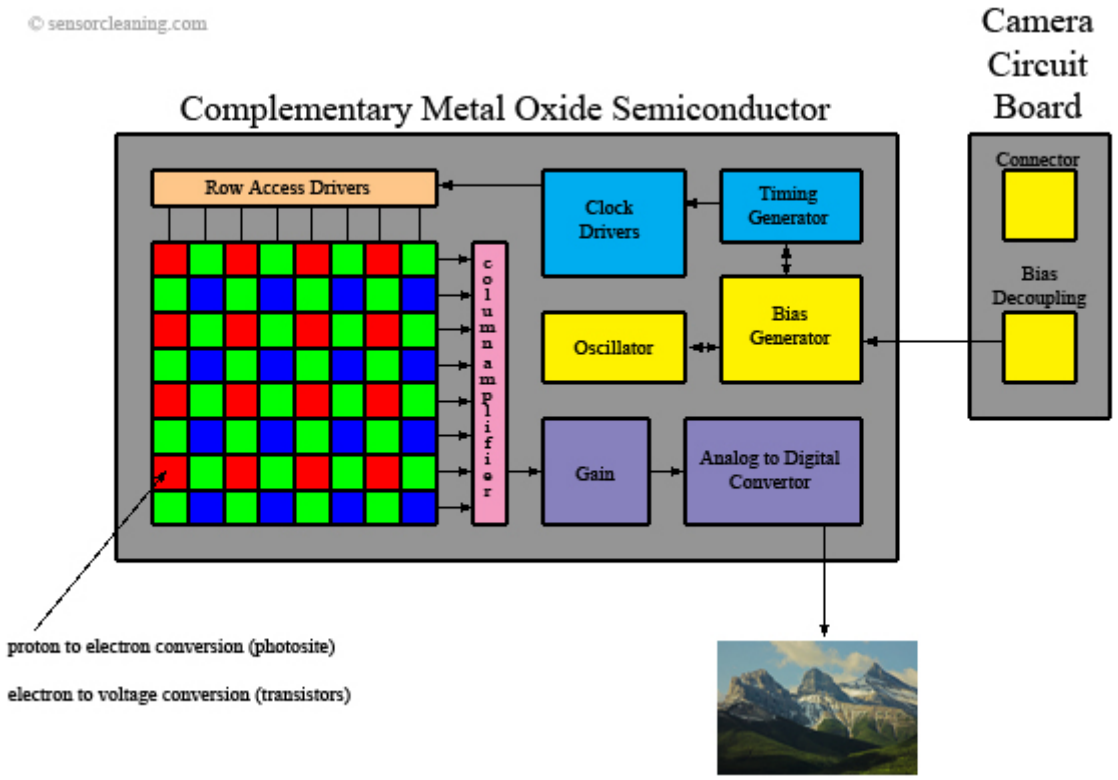

Figura 2.19: Conjunto Sensor CMOS mais circuito de câmera (http://sensorcleaning.com, 2010).

2.4.7 Processo de Leitura de Carga

Existem basicamente dois tipos de sensores de imagem CMOS: *passive pixel* e *active pixel*. O dispositivo baseado em *passive pixel* é composto por um capacitor e um transistor em cada *pixel* e por um amplificador em cada coluna da matriz de *pixels*. O capacitor armazena a carga de elétrons obtida durante a captura da imagem e o transistor faz o chaveamento desta carga para o amplificador da coluna correspondente ao *pixel*. Já no dispositivo baseado em *active pixel* o amplificador é introduzido junto ao *pixel*, isso permite que haja um controle na amplificação da carga para cada *pixel*, mas por outro lado, aumenta a área necessária para construção do mesmo.

Ambos dispositivos *passive* e *active pixel* usam as mesmas técnicas para leitura dos *pixels*. As linhas e colunas da matriz são selecionadas por um circuito seletor, a linha selecionada ativa os transistores e a carga de cada *pixel* é transferida para todas as colunas da matriz. Após, o circuito seleciona a coluna do *pixel* que deseja ler (Blanc, 2001). Esta carga lida do *pixel* é convertida para um valor digital através de um ADC e posteriormente armazenado em memória.

A possibilidade de selecionar as linhas e colunas permite acessar *pixels* individuais ou somente região de interesse dentro da imagem, assim, é possível aumentar o número de *frames* lidos por segundo.

2.4.8 Fatores que Comprometem a Qualidade da Imagem

Os dispositivos, tanto CCD como CMOS, apresentam alguns problemas de conversão de cenas em imagem digital (Taylor, 1998), sendo alguns deles apresentados a seguir:

- *Fill Factor:* É o percentual que cada *pixel* é sensitivo a luz. O ideal seria 100%, mas na realidade este valor não é obtido devido ao fato do *pixel* possuir internamente áreas insensíveis à luz.

- *Dark current noise:* É definido por uma carga não desejável que se acumula no *pixel* devido variação de temperatura que ocorre no ambiente onde o dispositivo está operando.

- *Quantum efficiency (QE):* É a proporção do número de elétrons detectados, dividido pelo produto do número de fótons incidentes num determinado período e multiplicado pelo número dos elétrons que cada fóton pode gerar.

- *Blooming:* É o efeito que ocorre quando, durante o período de integração da imagem, o potencial de carga de elétrons é excedido. Normalmente é causado pela presença de um objeto com brilho excessivo. Quando ocorre este excesso, os elétrons fluem para as regiões vizinhas criando uma área da imagem saturada.

2.4.9 Comparativo entre CCD e CMOS

Característica	CCD	CMOS
Sinal de saída do pixel	Elétron	Voltagem
Sinal de saída do circuito	Analógico	Digital
Sinal de saída da câmera	Digital	Digital
Sensibilidade	Alta	Moderada
Erro de Amplificação	N/A	Moderado
Ruído do Sistema	Baixo	Moderado
Complexidade do Sistema	Alta	Baixa
Complexidade do Sensor	Baixa	Alta
Componentes da Câmera	Sensor + Circuitos de Suporte + Lentes	Sensor + Lente, + Circuitos de Suporte
Custo Relativo de Desenvolvimento	Baixo	Alto
Custo Relativo do Sistema	Dependente da Aplicação	Dependente da Aplicação

Tabela 2.2: CCD vs CMOS (Litwiller, 2005).

Performance	CCD	CMOS
Ganho de Resposta	Moderado	Moderado/Alto
Faixa Dinâmica	Alto	Moderado
Uniformidade	Alta	Baixa a Moderada
Uniformidade de Exposição	Rápido, Comum	Pobre
Uniformidade de Sensibilidade	Alta	Baixa a Moderada
Velociade	Moderada a Alta	Muito Alta
Janelamento	Limitado	Extensivo
Anti-embaçamento	Alta a Baixa	Alta
Ajustabilidade e Temporização	Múltipla, Alta Voltagem	Simples, Baixa Voltagem

Tabela 2.3: Performance CCD vs CMOS (Litwiller, 2005).

CCD vs. CMOS

Previsão Inicial	Mudança	Resultado
Equivalência em performance comparado ao CCD	Necessitou de um processo de adaptação muito maior e de uma Litografia profunda de submicron que inicialmente pensado	Alta performance disponível, mas com um custo de desenvolvimento maior do que o CCD
Integração dentro do Circuito	Ciclos de desenvolvimento mais longos, custo elevado, problemas com ruído e flexibilidade de operação.	Maior integração com CMOS mas circuitos auxiliares ainda são necessários tanto para CCD quanto CMOS
Consumo de energia reduzido	Melhoria constante nos CCDs	Vantagem para o CMOS, mas com pouca margem.
Sistemas de imagem reduzidos	Componentes ópticos, circuitos adicionais e componentes de montagem são geralmente os fatores dominantes no sub-sistema de imagem.	CCDs e CMOS empatados
Economia de escala baseada no uso do mesmo processo de fabricação de circuitos integrados comuns	Processo de desenvolvimento extensivo e otimização necessária.	Sensores CMOS usam linhas de produçao com processos altamente adaptados, semelhante aos sensores CCD

Tabela 2.4: O Caminho Tortuoso do Desenvolvimento dos Sensores CMOS (Litwiller, 2005)

As tecnologias e os mercados continuam a amadurecer, mas a comparação ainda é muito comentada.

Sensores de imagam CCD (Dispositivo de Carga Acoplada) e CMOS (Semicondutor de Óxido Metálico Complementar) são duas tecnologias diferentes para capturar imagens digitalmente. Cada qual possui vantagens e desvantagens em aplicações diferentes. Nem uma categoria é superior à outra, embora alguns fabricantes venderem que apenas uma tecnologia é vantajosa. Nos últimos cinco anos muito mudou em ambas as tecnologias, e muitas projeções de ascenção e queda de ambas foram provadas falsas. A situação atual e a projeção futura para ambos é promissora, mas uma nova estrutura já existe para considerar as relativas vantagens e oportunidades para cada tecnologia.

Ambos os sensores convertem luz em carga elétrica e processam em sinais eletrônicos. Em um sensor CCD, a carga de cada pixel é transferida através de um número limitado de saídas (na maioria das vezes apenas uma) para ser convertida em voltagem, armazenada e enviada para fora do cirtuito como um sinal analógico. Todo o pixel pode ser dedicado à captura de luz, e a uniformidade da saída (um fator chave para a qualidade da imagem)

é alta. Em um sensor CMOS, cada pixel contém seu próprio conversor, e o sensor comumente possui amplificadores incluídos, corretores de ruído e circuitos digitalizadores, portanto a saída do circuito é digital. Estas outras funções aumentao a complexidade do projeto e reduzem a área disponível para a captura da luz. Com cada pixel realizando a sua própria conversão, a uniformidade é baixa. Mas o circuito pode ser construído para requerer menos circuitos secundários para operações básicas.

Existem algumas diferenças entre as tecnologias CCD e CMOS, as quais são apresentadas a seguir:

- O sensor de image CMOS pode conter vários circuitos no mesmo chip (System-on-a-Chip), eliminando os vários chips necessários para o CCD. Isso permite redução no tamanho, no consumo de energia e no custo. A redução do custo não ocorre somente pela integração de circuitos no mesmo chip, é também obtida porque a tecnologia CMOS não é desenvolvida especificamente para desenvolvimento de sensores CMOS, como é o caso da tecnologia utilizada no CCD. Esta tecnologia, é aplicada também no desenvolvimento da maioria dos circuitos integrados existentes no mercado atual, como microprocessadores, microcontroladores e DSP (Digital Signal Processing).

- Enquanto sensores de imagem CMOS se sobressaem na captura de imagem com alta intensidade luminosa, sofrem com a baixa quantidade de luz. A sensibilidade com a luz é reduzida porque parte do *pixel* é preenchido com circuitos que foram integrados ao chip. O percentual do *pixel* que é dedicado a coletar a luz é chamado de *fill factor*.

- Atualmente, os sensores de imagens CMOS adquirem imagem no modo contínuo (imagem completa) em torno de 1 *Gigapixel* por segundo. Isso ocorre graças à possibilidade de arquiteturas paralelas e à integração de funções em um único chip, reduzindo capacitâncias, indutâncias e atrasos de propagação (Blanc, 2001).

- Os sensores CCD são aplicados em produtos como filmadoras e câmeras digitais de alta qualidade, os quais normalmente são utilizados na ciência e astrologia. Sensores CMOS, normalmente são aplicados em sistemas portáteis, tais como biométricos, automotivos e robóticos, onde não se requer alta definição. Baixo consumo de

energia e tamanho reduzido também são características dos sensores CMOS.

2.5 Computação Reconfigurável

Em um nível fundamental, computação reconfigurável é o processo de melhor explorar o potencial de um *hardware* reconfigurável. Embora o sistema completo deva incluir a compilação de um *software* e aplicações de alta performance, a melhor maneira para começar a entender a computação reconfigurável é a nível de circuito, uma vez que as habilidades e limitações dos circuitos integrados tem influência crucial em todos os passos da construção do sistema. No entanto, o contrário é verdadeiro, como sistemas reconfiguráveis são projetados inicialmente para uma determinada aplicação que será desenvolvida, e um circuito que não necessariamente supre as necessidades de uma determindada aplicação, ou não suporta uma função importante, ou não pode ser utilizado de maneira eficiente em um projeto, este não obterá sucesso (Hauck, 2008).

A computação reconfigurável tem se destacado devido o desenvolvimento das FPGAs. De um lado, as FPGAs comuns são cruciais para este campo. Elas representam uma fonte de *commodities*, oferecendo um componente de silício barato e rápido, fabricado através dos mais modernos processos de fabricação existentes.

Por outro lado, a simples razão que a vasta maioria dos clientes das FPGAs as usarem como simples e baratos ASICs, com um rápido tempo de desenvolvimento, faz com que as FPGAs não sejam otimizadas para a computação reconfigurável. Por isso estes dispositivos não são totalmente o que um usuário de computação reconfigurável possa querer, mas são muito próximas disto.

Máquinas reconfiguráveis tem emergido como uma tecnologia capaz de prover alta performance computacional em diversas aplicações, incluindo processamento de sinais, processamento de imagem, simulações aceleradas e computação científica. Em muitos casos, aplicações que utilizam computação reconfigurável superam o equivalente de dezenas ou centenas de microprocessadores contemporâneos ou processadores digitais de sinais (DSPs). Alta performance é atingida atráves da construção de operadores customizados dinamicamente, dutos de dados e transmissores projetados especificamente para a tarefa a ser realizada. Com esta abordagem, caracteristicas de uma aplicação particular, como paralelismo, localidade e resolução de dados podem ser exploradas. Por isso,

máquinas reconfiguráveis provém benefícios de performance computacional de circuitos de aplicações específicas (ASICs), e ainda possuem a flexibilidade e a reconfigurabilidade de microprocessadores de propósito geral.

Pesquisas em arquiteturas de computadores tem sido sempre a preocupação central de comunidades de engenheiros de computação. Os objetivos das pesquisas variam conforme as aplicações e o ambiente em que os processadores irão ser empregados. Para processadores que irão ser empregados em máquinas paralelas para alta performance, como é o caso de aplicações de simulação climática, o foco esta concentrado em altas frequências de *clock*, paralelismo e altas taxas de comunicação, tudo isso utilizando bastante potência. No entanto, em sistemas embarcados, o preço final do equipamento é o fator principal durante o desenvolvimento. Um pequeno microcontrolador é usualmente usado para controlar a aquisição de dados de sensores e promover dados aos atuadores, em uma frequência mais baixa. Em alguns outros sistemas embarcados, potência e custo são os objetivos centrais. Nesses tipos de sistemas, o aumento da demanda por poder computacional, que contradiz custo e potência, exerce muita pressão sobre os engeinheiros que necessitam encontrar um balanceamento entre todos os requisitos de projeto. Para um veículo autônomo usado para exploração de um dado ambiente, a unidade de processamento deve ser capaz de capturar, compactar e enviar imagens a uma estação de controle. Paralelamente a isso, o sistema deve realizar outras tarefas como detectar obstáculos e evitá-los. Em tal sistema, o consumo de energia deve ser otimizado para permitir que ele tenha a maior autonomia possível. Por outro lado, o processador deve processar frames de imagens o mais rápido possível, a fim de evitar perdas de reconhecimento. Detecção e desvio de obstáculos também necessitam de velocidade, pois também devem ser feitos da forma mais rápida possível para evitar danos ao veículo, ou seja, deve ser realizado em tempo real.

A multiplicidade de objetivos têm levado ao desenvolvimento de várias arquiteturas de processamento , cada qual otimizada de acordo com um objetivo específico. Estas arquiteturas podem ser categorizadas em três grupos dominantes, de acordo com o seu grau de flexibilidade: grupo de computação geral, que é baseado no paradigma de Von Neumann (VN); processadores de domínio específico, feitos para uma classe de aplicações que possui caracteristicas bem variadas; e processadores para aplicações específicas, projetados para

apenas uma aplicação.

Considerando estes três grupos de arquiteturas, podemos identificar dois pontos principais para caracterização dos processadores: flexibilidade e performance. Os computadores que utilizam a arquitetura Von Neumann, chamado de grupo geral, são muito flexíveis, pois são aptos a realizar qualquer tipo de tarefa. Esta é a razão da terminologia GPP (Processadores de Propósito Geral) ser usada para a máquina Von Neumann. Eles não possuem muita performance, pois não podem executar tarefas em paralelo.

Por sua vez, o grupo dos processadores de domínio específico possuem muita performance porque são otimizados para uma aplicação particular. O set de instruções requerido para uma dada aplicação pode ser construído em um *chip* A alta performance é possível devido ao *hardware* ser sempre adaptado à aplicação.

Se considerarmos duas escalas, uma para a performance, e outra para flexibilidade, os computadores Von Neumann podem ser colocados em uma extremidade e os de Domínio Específico em outra, conforme é ilustrado na figura 2.20. (Bobda, 2007)

Idealmente, gostaríamos de ter a flexibilidade dos GPP (Processadores de Propósito Geral) e a performance de um ASIP (Processadores de Aplicação Específica) no mesmo circuito. Gostaríamos de ter um dispositivo capaz de se adaptar à aplicação durante o funcionamento. Este dispositivo é chamado de dispositivo de hardware reconfigurável ou dispositovo reconfigurável, ou Unidade de Processamento Reconfigurável (RPU) por analogia a Unidade de Processamento Central (CPU).

Definição de Computação Reconfigurável: é definida como o estudo da computação usando dispositivos reconfiguráveis. Para uma dada aplicação, em um determinado momento, a estrutura espacial do dispositivo será modificada para atingir o melhor rendimento para acelerar a aplicação. Se uma nova aplicação necessita ser computada, a estrutura do dispositivo será modificada novamente para realizar a nova tarefa. Ao contrário do computadores que seguem a arquitetura Von Neuman, que são programados através de um set de instruções a serem executadas sequencialmente, a estrutura dos dispositivos reconfiguráveis é modificada através da mudunça total ou parcial do hardware no momento da compilação ou no momento da excecução, pela reprogramação do dispositivo.

O progresso da computação reconfigurável tem sido extraordinário nas duas últimas

décadas. Isto se deve principalmente pela larga aceitação dos Dispositivos Lógicos Programáveis (FPGAs) que estão agora se estabelecendo como os dispositivos mais usados entre os dispositivos reconfiguráveis. O número de workshops, conferências e simpósitos que tratam deste tópico também tem crescido, seguindo a evolução das FPGAs. Dispositivos reconfiguráveis podem ser usados em um número muito grande de aplicações, como prototipagem rápida, customização de sistemas *on ship*, computação multi-modal, computação paralela, sistemas adaptativos e evolutivos.

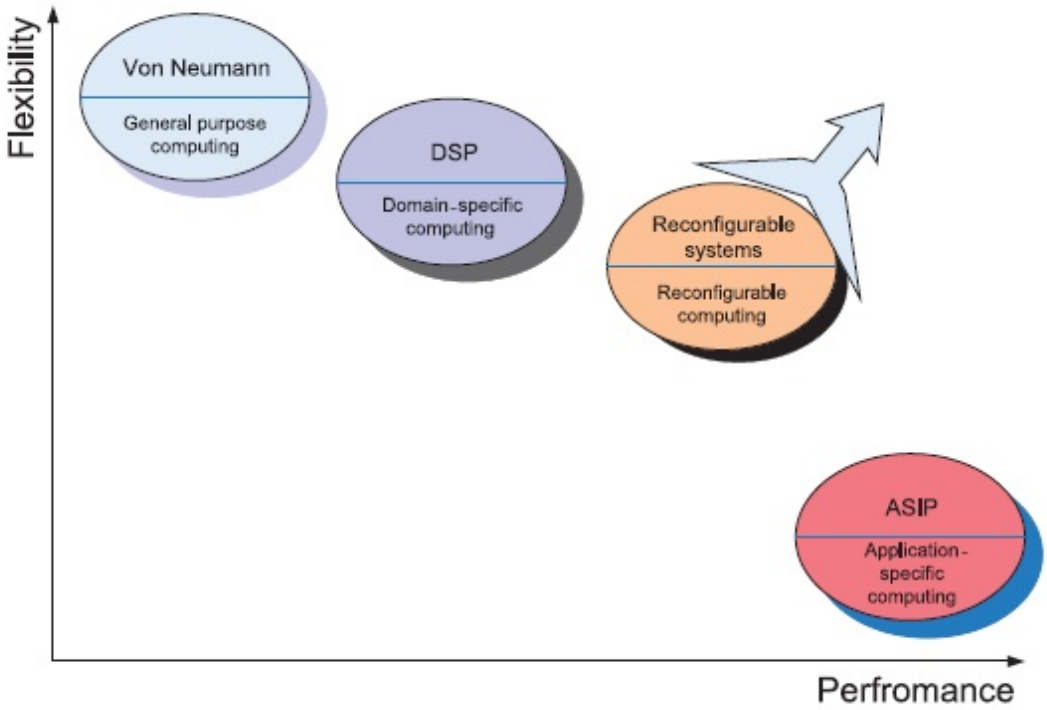

Figura 2.20: Flexibilidade versus Performance de classes de processadores (Bobda, 2007).

2.5.1 FPGAs

Introduzidas em 1985 pela empresa Xilinx, matrizes de portas de campo programáveis, FPGA *(Field Programable Gate Arrays)* ou ainda Matriz de Portas de Campo Programáveis é um circuito integrado que contém um grande número (na ordem de milhares) de unidades lógicas idênticas. Neste aspecto estas unidades lógicas podem ser vistas como componentes padrões que podem ser configurados independentemente e interconectados a partir de uma matriz de trilhas condutoras e *switches* programáveis. Um arquivo binário é gerado para configuração da FPGA a partir de ferramentas de software seguindo um

determinado fluxo de projeto. Esse arquivo binário contém as informações necessárias para especificar a função de cada unidade lógica e para seletivamente fechar os *switches* da matriz de interconexão. Resumindo, o *array* de unidades lógicas e a matriz de interconexão, que podem ser programados pelo usuário, formam a estrutura básica da FPGA para especificação de crcuitos integrados complexos. A figura 2.21 mostra a estrutura de uma FPGA, com seus principais componentes como as saídas ou entradas programáveis, os blocos lógicos, as interconexões e a matrix de conexões.

As funções implementadas na FPGA são particionadas em módulos, cada qual pode ser implementado em um bloco lógico. Os blocos lógicos são conectados usando a programação de interconexão. Os três componentes básicos da FPGA (blocos lógicos, interconexões, e entradas e saídas) podem ser programadas pelo usuário. FPGAs podem ser programadas uma ou várias vezes, dependendo da tecnologia usada.

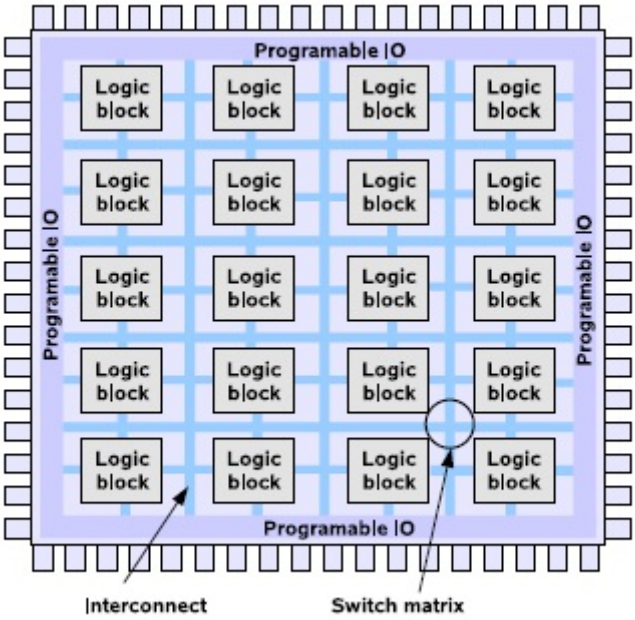

Figura 2.21: Estrutura de uma FPGA (Bobda, 2007)

Além dos recursos padrões, uma FPGA pode disponibilizar no arranjo bidimensional recursos bastante sofisticados, tais como multiplicadores dedicados, MACs programáveis (Multiplicador e Acumulador, também denomidados de *slice XtremeDSP*), blocos de memória, DCM (Digital Clock Manager ou Gerenciador de Relógio Digital, utilizados para multiplicar ou dividir a frequência de um sinal de clock), microcontroladores (IBM PowerPC, Nios, etc.) e trans-receptores multi-giga bit (que servem para implementar

interfaces seriais para transferência de bits a altíssima velocidade). A utilização de tais recursos embarcados possibilita otimizar o consumo de área e também desenvolver projetos mais eficientes.

O termo *Field Programable* por sua vez significa que as funções das FPGAs são definidas por um programa do usuário em vez de serem definidas pelo fabricante do dispositivo. Em circuitos integrados típicos (ASIC - Application Specific Integrated Circuit ou Circuito Integrado de Aplicação Específica) a implementação é realizada no tempo da manufatura. Nas FPGAs, dependendo do dispositivo, o programa pode ser transferido para o dispositivo permanentemente, semi-permanentemente como parte do processo de montagem da placa, ou trasnferido a partir de uma memória *flash* cada vez que o dispositivo é ligado. No último caso, a tecnologia utilizada para implementação da FPGA é a de memória estática (SRAM). Por este motivo, toda vez que o dispositivo é desligado perde-se a programação. Esta flexibilidade de programação associada a potentes ferramentas de desenvolvimento e modelagem, possibilita ao usuário acesso a projetos de circuitos integrados complexos sem os altos custos de engenharia associados aos ASICs (Cardoso, 2007).

Apesar da maior velocidade e da performance superior de processamento da FPGA quando comparada a DSPs e a microcontroladores, uma única FPGA pode conter uma solução completa implementada em um único circuito integrado, sem requerer suporte lógico externo devido aos seus recursos intrínsecos, com uma evidente redução de custo. Por outro lado, a versatilidade das linguagens de descrição de hardware proporcionam a versatilidade e a portabilidade para desenvolvimentos presentes e futuros, algo que é virtualmente impossível para os processadores de uso geral (et al., 2010).

2.5.2 Estruturas das FPGAs

FPGAs são constituídas por várias células lógicas programáveis dentro de um componente a fim de constituir um vetor de recursos computacionais. O resultado desta estrutura é dependente do fabricante. Dependendo do arranjo dos blocos lógicos e de sua interconexão, as FPGAs podem ser classificadas em quatro categorias: vetor simétrico *symmetrical array*, baseada em colunas*row-based*, baseada em hierarquia *hierarchy-based* e mar de portas*sea of gates*.

A figura 2.22 mostra os tipos básicos de FPGAs (Bobda, 2007).

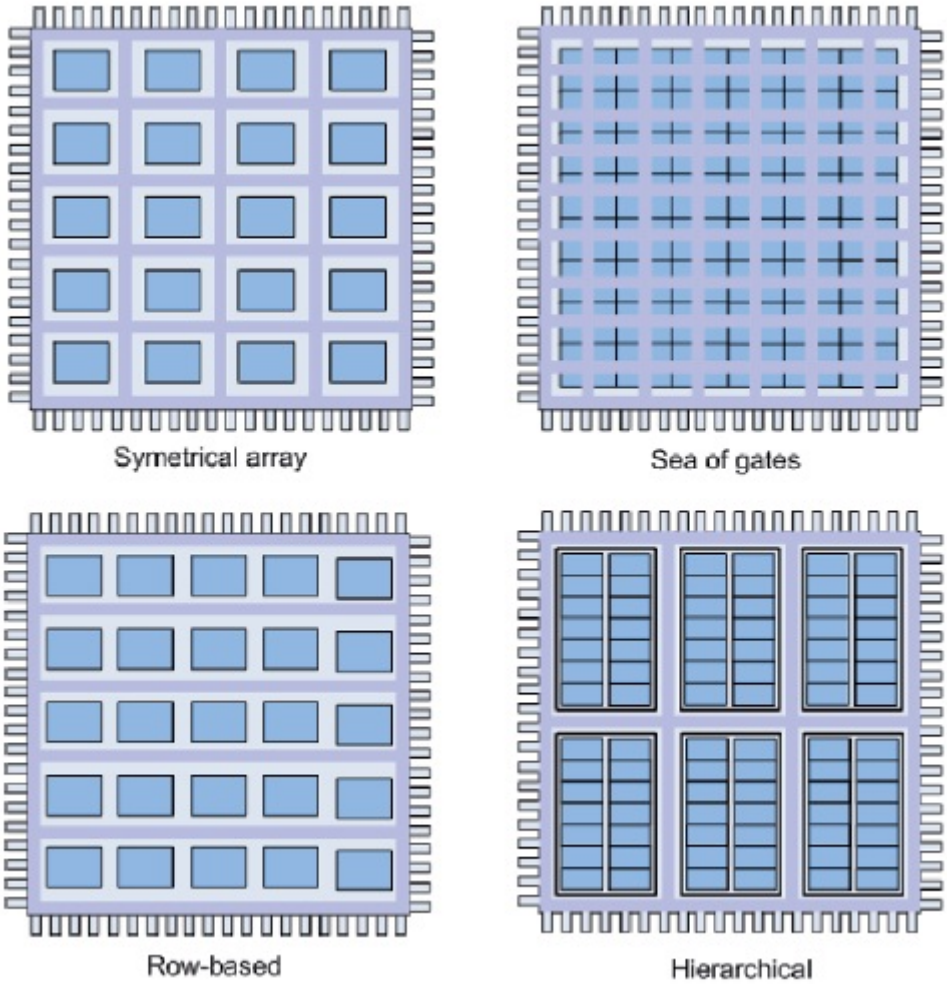

Figura 2.22: Quatro tipos básicos de estruturas de FPGAs (Bobda, 2007)

Matriz Simétrica

FPGAs baseadas em matrizes simétricas consistem em uma matriz bi-dimensional de blocos lógicos inserida em um conjunto de linhas verticais e horizontais. Elementos de comutação existem nas intersecções das linhas verticais e horizontais para permitir a conexão das linhas verticais e horizontais. Exemplos de FPGAs baseados na estrutura de matriz simétrica são a Xilinx FPGA Virtex e os dispositivos da Atmel.

FPGAs baseadas em Linhas

Uma FPGA baseada em linhas consite em linhas alternadas de blocos lógicos ou macro células e canais. O espaço entre os blocos lógicos é chamado de canal e é usado para rotear os sinais.

Mar de Portas

Como nas matrizes simétricas, as macro células são organizadas em uma matriz de estrutura bidimensional de tal forma que a entrada da matriz corresponde à coordenada de uma dada célula macro . A diferença entre a matriz simétrica e mar-de-portas é que não há espaço entre as células macro para roteamento. Os fios de interconexão são fabricados no topo das células.

FPGAs baseadas em Hierarquia

Neste tipo de FPGA, as macro células são hierarquicamente colocadas no dispositivo. Os elementos com a menor granularidade são colocados no menor nível de hierarquia. Eles são agrupados para formar os elementos para o próximo nível. Cada elemento de um nível i consiste em um dado número de elementos do nível i -1. As FPGAs da Altera (Flex, Cyclone II e Stratix II) possuem dois níveis hierárquicos.

Capítulo

3

Plataforma de Desenvolvimento

Este capítulo descreve os recursos de *hardware* e *software* que são utilizados no desenvolvimento deste trabalho e a maneira como estes recursos são utilizados. O *hardware* apresentado é composto pelo Kit de desenvolvimento DE2 70 e por uma câmera digital de tecnologia CMOS de 5 mega pixels; o *software* apresentado é composto por um conjunto de ferramentas de desenvolvimento de sistemas em *chips* programáveis, que são integradas pela ferramenta EDA (Electronic Design Automation) Quartus II.

3.1 Kit de Desenvolvimento DE2 70

A implementação e validação do sistema proposto em FPGA (Field Programmable Gate Array), é realizada utilizando-se o Kit de desenvolvimento DE2 70 da Terasic (edição Ciclone II) (Altera, 2009), o qual é basicamente composto por uma placa com FPGA da família Cyclone II e por ferramentas de desenvolvimento de *software* embarcado, fornecido pela empresa Terasic. O propósito deste Kit é fornecer uma estrutura para acelerar o desenvolvimento de sistemas embarcados em *chips*, conhecidos como SOPC (System-On-a-Programable-Chip). Nas seções seguintes são apresentadas as informações relativas ao Kit de desenvolvimento que são consideradas relevantes ao sistema proposto.

3.1.1 Placa de Desenvolvimento DE2 70

A placa de desenvolvimento NiosII fornece uma plataforma para o desenvolvimento de *hardware* de sistemas embarcados baseados no dispositivo de FPGA EP2C70F896C6N da família Cyclone II. Este dispositivo é composto por 68.416 elementos lógicos (LEs) e 1152 Kbits de memória *on-chip*.

Características da Placa

As características citadas a seguir apresentam os recursos fornecidos pela placa de desenvolvimento DE2 70. Estas informações são importantes, pois através delas é possível constatar quais funcionalidades podem ser implementadas neste *hardware*.

- FPGA EP2C70F896C6N da família Cyclone II. Suas principais características são apresentadas através da tabela 3.1;

Elementos Lógicos (LE)	68.416
Blocos M4K RAM (4 Kbits mais 512 bits de paridade	250
Total RAM bits	1.152.000
Multiplicadores *embedded*	150
Blocos DSP	6
PLL (Phase Locked Loop)	4
Pinos disponíveis ao Usuário	622

Tabela 3.1: Características do dispositivo EP2C70F896C6N

- Componente de configuração serial Altera - EPCS16
- USB *Blaster* agregado para programação utilizando controle APU via*JTAG*
- 2 Mbytes de memória SSRAM
- 2 módulos de 32-Mbyte de memória SDRAM
- 8 Mbyte de memória Flash
- Conector para cartão SD
- 4 botões de controle
- 18 chaves de controle

- 18 Leds vermelhos
- 9 Leds verdes
- Osciladores de 50-MHz e 28.63-MHz para base de tempo
- Conector VGA DAC (10 bits, alta velociade, DACs triplas) com conector com saída VGA.
- 2 decodificadores de TV (NTSC/PAL/SECAM) e conector para TV
- Controlador *Ethernet* 10/100 com conector
- Controlador USB *Host/Slave* com conectores USB tipos A e B
- *Transceiver* RS-232 com conector de 9 pinos
- Conector PS/2 para mouse e teclado
- *Transceiver* IrDA.
- 1 conector SMA
- 2 *Headers* de expansão de 40 pinos com proteção de diodos

A figura 3.1 apresenta a placa de desenvolvimento DE2 70, o FPGA e as portas de comunicação existentes. Todos os acessórios necessários para a conexão desta placa, como a fonte de alimentação e cabo de programação, são fornecidos junto ao kit de desenvolvimento.

3.1.2 Ferramentas de Desenvolvimento de Software Embarcado

O ambiente de desenvolvimento é composto por um conjunto de ferramentas de desenvolvimento de sistemas em *chips* programáveis, conhecido como SOPC, as quais são integradas pela ferramenta EDA (Electronic Design Automation) Quartus II. Este ambiente compreende todas as fases de desenvolvimento de sistemas em FPGA e CPLD, que envolvem a descrição do projeto, síntese da lógica programada, roteamento e inserção da lógica obtida a partir da síntese no dispositivo, análise de tempos, simulação e por fim, programação e configuração do dispositivo (Altera, 2003b).

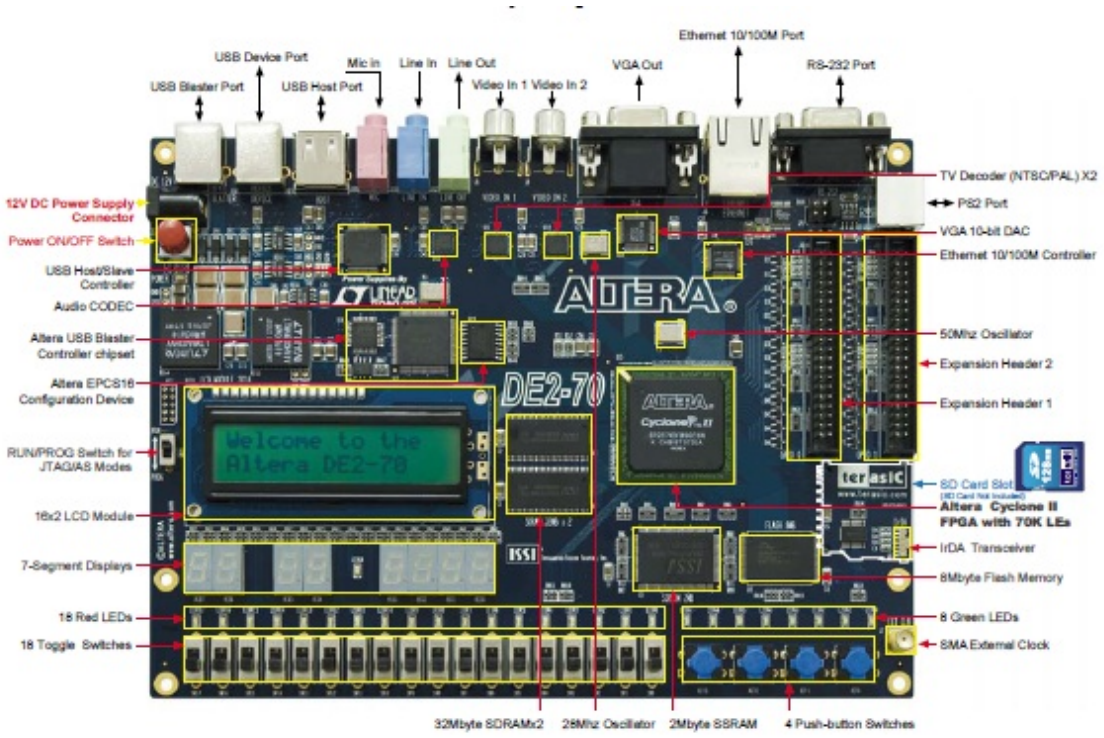

Figura 3.1: Placa de desenvolvimento DE2 70 (www.terasic.com, 2008)

O Quartus II permite a instalação de diversos recursos que facilitam e otimizam o esforço necessário para o desenvolvimento de projetos. Neste ambiente há um *megawizard* para auxiliar o usuário durante a inserção de funções em seu sistema, dentre estas funções podemos citar os componentes que realizam operações aritméticas, memórias, portas lógicas, DSP e processadores embutidos.

3.1.3 Processador RISC Customizável

O processador RISC (Reduced Instruction Set Computer) de propósito geral, customizável, é constituido por um *soft-core* desenvolvido pela empresa Altera para ser embutido nos dispositivos lógicos programáveis (PLD), que é conhecido como processador NIOS II (Altera, 2003c). As principais características da arquitetura Nios são:

- Instruções de 32 bits
- 32 registradores de propósito geral
- Registradores opcionais *shadow*
- 32 fontes de interrupção

- Controlador externo de interrupções para mais fontes de interrupção
- Instrução de multiplicação de 32 bits, e divisão com resultado em 32 bits
- Instruções de multiplicação dedicadas para resultados em 64 e 128 bits
- Instruções de ponto flutuante
- Instrução de *barrel shifter*
- Acesso a uma variedade de periféricos *on chip*, e interfaces para memórias e periféricos *off chip*
- Módulo de *debug* assistido por hardware, que habilita *start, stop, step e trace* através do controle das ferramentas de desenvolvimento Nios II
- Módulo de gerenciamento de memória opcional (MMU - Optional memory management unit) para suporte a sistemas que requerem MMUs
- Módulo de proteção de memória opcional (MPU - Optional memory protection unit)
- Ambiente de desenvolvimento de *Software* baseado na ferramenta GNU C/C++ e nas ferramentas de desenvolvimento para o Eclipse
- Integração com as ferramentas de *design* e análise em tempo real da Altera *SignalTapR II Embedded Logic Analyzer*
- Set de instruções compatível com todos os sitemas Nios II
- Performance até 250 DMIPS

Um processador Nios II é equivalente a um microcontrolador ou computador em um chip que inclui um processador e uma combinação de periféricos e memória em um único chip. Um sistema com processador Nios II consiste de um núcleo Nios II, uma série de periféricos *on chip* como memórias e interfaces para memórias externas, tudo implementado em um único dispositivo Altera. Como na maioria das famílias dos microcontroladores, todos os sitemas Nios II utilizam um set de instruções consistente dentro de um modelo de programação.

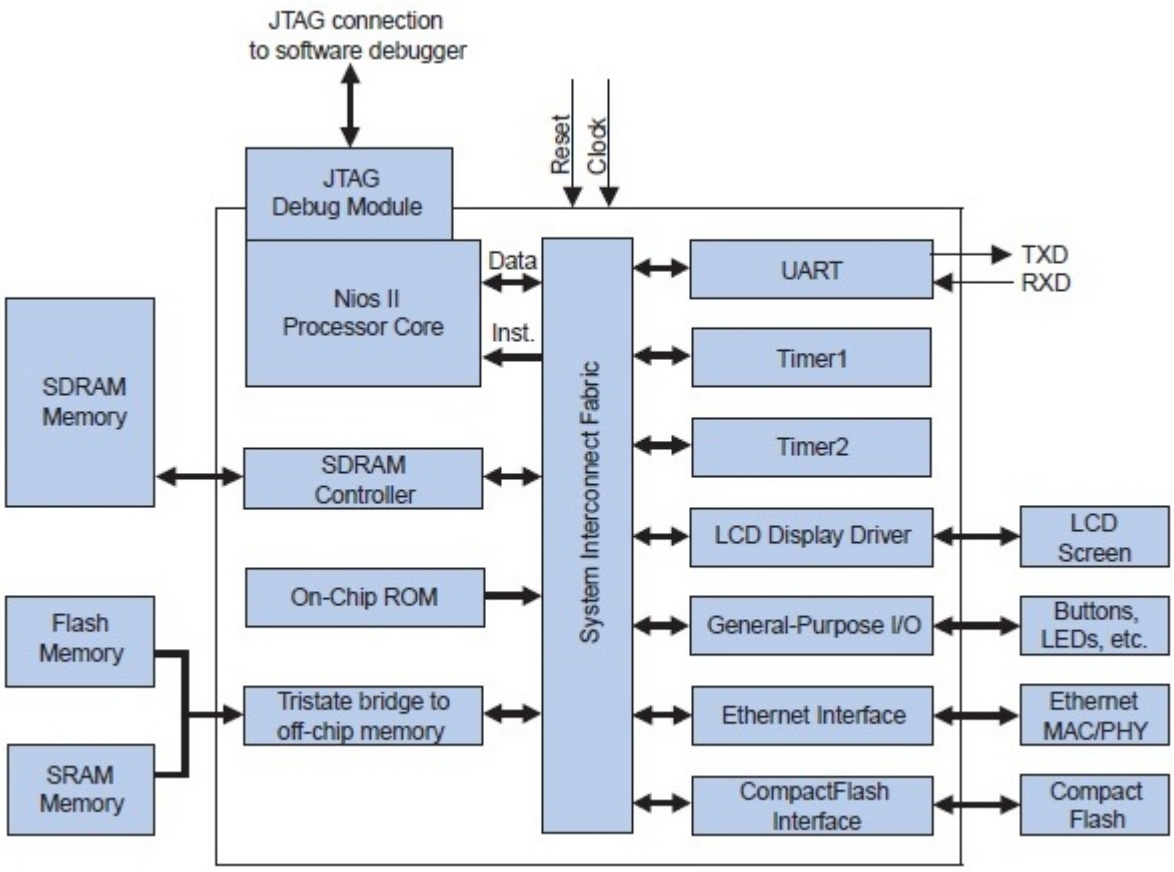

Figura 3.2: Componentes do sistema com processador Nios II

Na prática, a maioria dos *designs* são implementados com circuitos externos ao sistema do processador. As FPGAs da Altera possuem flexibilidade e recursos para aumento de performance do processador Nios II. Paralelamente, é possível eliminar circuitos e *features* que não estejam sendo usados para deixar o projeto otimizado e com menor custo.

Sistemas baseados em processadores embutidos Nios II podem incluir um ou mais processadores.

A figura 3.2 mostra um exemplo de um sistema particular contendo uma CPU Nios com *cache* de instruções e de dados, um *on-chip debug core*, um controlador de acesso direto a memória (DMA), diversos periféricos (UART, IO, *ethernet* e interface de memória).

On-chip debug core é um módulo que oferece recursos ao projetista para realizar o *debug* de processador embutido. Este recurso é opcional, mas caso seja selecionado, o módulo de *debug* provê um conjunto de características incluindo controle de execução, pontos de parada do *hardware*, pontos de parada do *software*, *on-chip trace* e *off-chip trace* .

3.1.4 Ferramentas GNUpro para Software Embarcado

As ferramentas do GNUpro são baseadas no padrão industrial de *software* livre do GNU, distribuído pela empresa Red Hat. Todos os processadores embutidos da Altera suportam as ferramentas de desenvolvimento e de *debug* do GNUpro (Altera, 2003a), que são otimizadas para este tipo de processador, fornecendo um ambiente de desenvolvimento familiar para o projetista, incluindo:

- Compilador C (gcc) and C++ (g++);
- *Debugger* (gdb) a nível de código e *assembler*;
- *Assembler* GNU (gas);
- *Linker* GNU (ld);
- *Software code profiler* (gprof) GNU;
- *Insight Debugger*;
- Utilitários binários específicos do processador NiosII.

3.2 Módulo da Câmera CMOS

Uma foto tirada de um microscópio do sensor da câmera digital de 5 Mega Pixel do Kit de desenvolvimento utilizado na implementação do projeto pode ser vista na figura 3.3

O vetor de pixels da câmera D5M consiste em 2.752 colunas por 2004 linhas de pixels endereçados por coluna e linha. O endereço 0,0 (coluna 0, linha 0) representa o canto superior direito do vetor, olhando para o sensor como é mostrado na figura 3.4.

O vetor consiste em 2592 colunas por 1944 linhas de região ativa central que representa a imagem de saida, circundada por uma região limite(tambem ativa), e por uma borda de pixels negros (conforme as tabelas). A região de fronteira pode ser usada para evitar efeitos de pico ao fazer processamentos de cor, para que se possa atingir uma imagem de 2592 x 1944, enquanto as colunas e linhas negras podem ser usadas para monitorar o nível de negro.

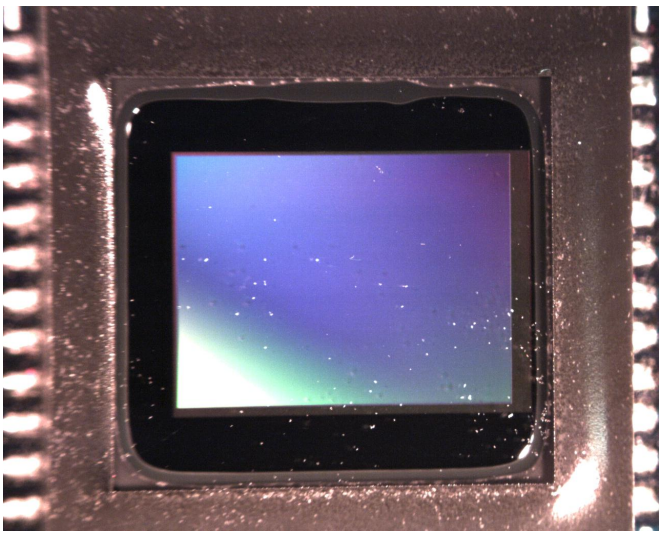

Figura 3.3: Sensor CMOS Visto ao Microscópio

Os pixels sao fornecidos no padrão Bayer, consistindo de quatro cores: Verde1, Verde2, Vermelho e Azul (G1, G2, R, B) representando as três cores de filtros. Quando os modos de espelho estão habilitados, a saída da primeira linha alterna entre os pixels G1 e R, e a saída da segunda linha alterna entre os pixels B e G2. Os pixels Verde1 e Verde2 possuem a mesma cor de filtro, mas são tratados de maneira alternada pelo caminho de dados e pela cadeia de sinal analógica.

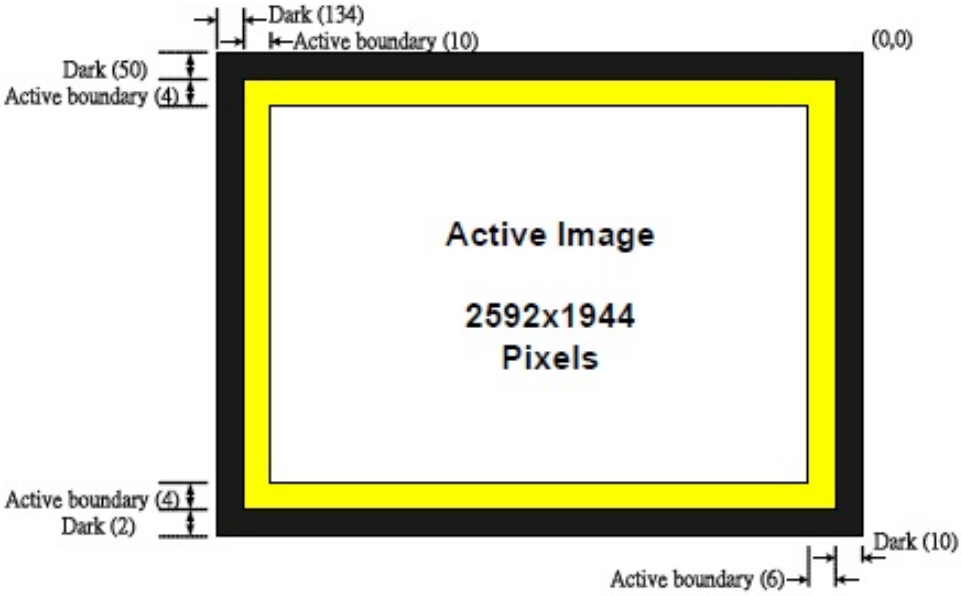

Figura 3.4: Descrição do Vetor de Pixels (www.terasic.com, 2008)

A imagem é adquirida através de um módulo de câmera CMOS com saída digital colorida . Este módulo fornece os sinais da imagem continuamente em formato binário

de 30 *bits* de largura através de uma porta. Estes sinais são lidos e tratados pelo sistema a fim de obter os *frames* da imagem. Neste módulo de câmera também existe sinais de controle da imagem como ganho analógico e ganho digital, que são programados através da interface I2C presente na câmera. Na figura 3.5 é apresentado uma foto desta câmera junto a placa de desenvolvimento DE270.

Figura 3.5: Câmera CMOS D5M conectada à placa de desenvolvimento DE2-70 (www.terasic.com, 2008).

A imagem é captada pelo sensor e transferida ao módulo de captura da imagem através das linhas de comunicação DATA, FVAL, LVAL.

As imagens capturadas são divididas em frames, que depois são divididas em linhas. Por default, o sensor produz 1944 linhas de 2592 colunas cada. Os sinais FVAL (Frame Válido) e LVAL (Linha válida) indicam as fronteiras entre os frames e linhas, respectivamente. PIXCLK pode ser usado como *clock* para transmissão dos dados. Para cada ciclo de PIXCLK, um dado de 12 *bits* é colocado na porta de saída DATA ou DOUT. Quando as linhas FVAL e LVAL são acionadas, o pixel é válido.

O modo default de transmissão é mostrado na figura 3.6.

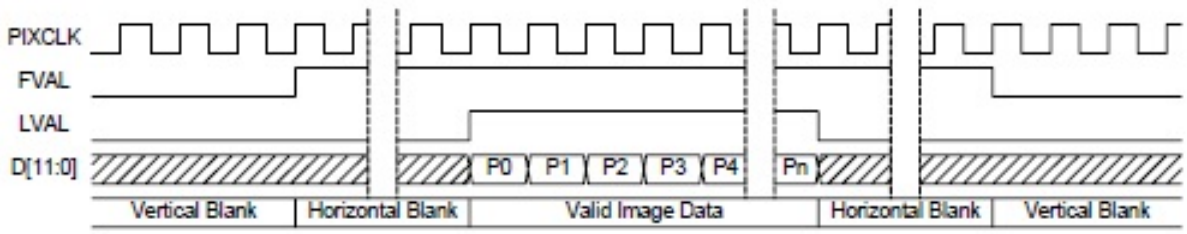

Figura 3.6: Temporização Default de Saída do Frame (www.terasic.com, 2008).

3.3 Conexão da Câmera com a Placa de Desenvolvimento D2 - 70

A câmera CMOS e a placa de desenvolvimento DE2 - 70 são conectadas fisicamente para que os sinais possam ser enviados entre elas. Para realizar esta tarefa, é necessário analisar a compatibilidade dos sinais, principalmente em relação aos níveis de tensão e consumo de corrente. Na placa de desenvolvimento DE2 - 70 há conectores de expansão de propósito geral de 40 pinos onde há pinos que fornecem tensões de alimentação nos níveis de 1,5, 3,2 e 5,0 Vcc; sinais de terra (*ground*) e pinos que podem ser utilizados como meio para entrada ou saída, dependendo da programação, de sinais no sistema. Cada pino dos conectores estão conectados a dois diodos e um resistor que promovem a proteção contra alta e baixa voltagens.

3.4 Monitoramento da Imagem no Sistema

Os sinais fornecidos pela câmera CMOS são digitais, podendo ser utilizados diretamente pelo FPGA. Essa é uma das vantagens em relação ao uso de câmeras que fornecem sinais analógicos, pois o mesmo teria de ser convertido para digital. Em contra partida, para reproduzir as imagens seria mais fácil, pois é comum encontrar aparelhos eletrônicos, como televisores e computadores, com entrada de vídeo analógico capaz de reproduzir imagens em diversos padrões. Durante o desenvolvimento, foi utilizada a saída de vídeo padrão VGA.

Capítulo

4

Implementação do Sistema Proposto

Este capítulo apresenta a implementação do sistema proposto e os resultados obtidos neste trabalho. Toda a implementação foi realizada utilizando-se a plataforma de desenvolvimento DE2 da Altera, linguagem de programação Verilog, e informações obtidas no site da empresa Terasic, que fabrica e comercializa o kit de densenvolvimento DE-2 70. O projeto foi baseado em uma aplicação também fornecida pela Terasic (http://www.terasic.com.tw, 2010), na qual módulos base implementados em linguagem Verilog forneceram uma base para a realização da implementação proposta.

A figura 4.1 mostra os módulos de construção do histograma, cálculo do erro e do controlador PID. O histograma é montado através da leitura dos dados transmitidos pelas linhas paralelas de dados DATA, e pela linha de controle FVAL. As linhas de dados DATA transmitem em 12 bits a voltagem de cada pixel do sensor até a FPGA. Se a transmissão do frame, indicada pela linha FVAL é válida, o histograma é então analisado e o controlador PID pode ser alimentado com o erro calculado. Após o cálculo do novo valor de ganho realizado pelo controlador, este valor é transmitido ao sensor via comunicação I2C. A frequência do *clock* da FPGA utilizada foi 50MHz e a velocidade de comunicação da linha I2C foi de 100 kbits/s.

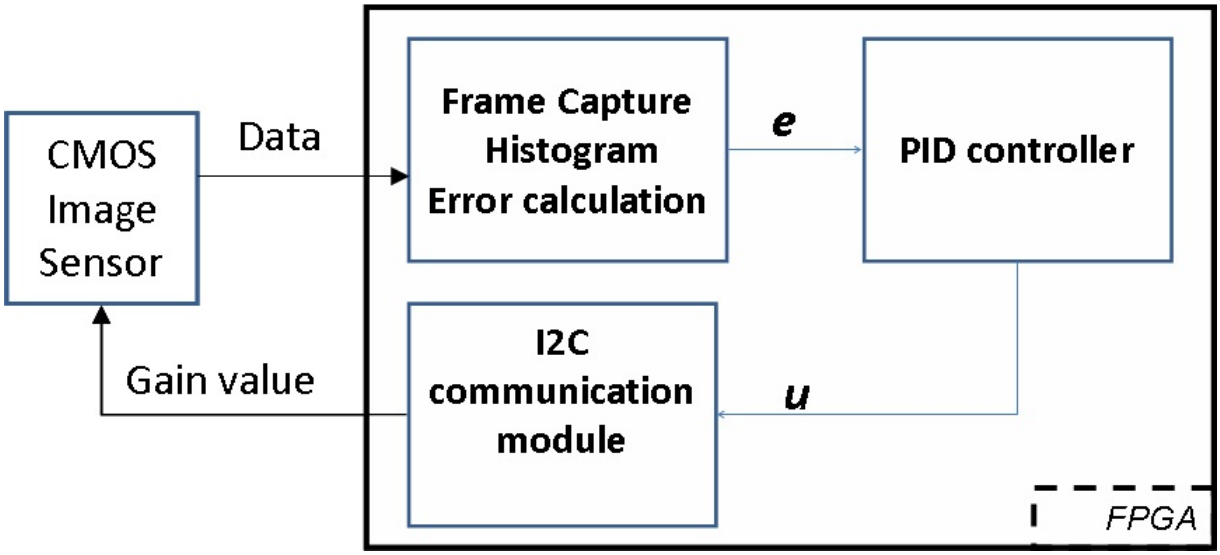

Figura 4.1: Implementação do sistema utilizando FPGA

4.0.1 Cálculo do histograma e cálculo do erro

O Erro do sistema é calculado a partir de uma análise do histograma da imagem aquisitada. Como podemos ver na figura 4.2, o histograma exemplo é dividido em 10 faixas de valores. O histograma também possui outra divisão, que se encontra neste caso entre as faixas 5 e 6, e uma divisão horizontal, que indica o valor da média aritimética do histograma. Valores nas faixas 1 a 5 acima do valor médio são considerados como partes escuras da imagem e valores nas faixas 6 a 10 acima do valor médio são consideradas partes claras da imagem. A escala de proporcionalidade de contribuição do erro também é distruibuída entre as faixas do histograma. No exemplo, a faixa 1 é a que mais contrubui para o aumento do ganho, pois é a mais escura, e indica valores de pixels que captaram baixa intensidade luminosa. Valores acima do valor médio dessa faixa contribuem com um multiplicador maior que os da faixa 2 e assim respectivamente até a faixa 5. Já os falores das faixas 6 a 10 contrubuem para a diminuição do ganho, uma vez que indicam que a imagem é clara. O maior peso está na faixa 10, que indica valores de pixels possivelmente saturados, e por isso uma diminuição do ganho deve ser feita.

Uma abordagem também baseada na análise do histograma é apresentada em (Nadi R. Itani, 2009), onde o erro também é calculado baseado na análise do histograma, porém o método apresentado torna-se instável quando aplicado em imagens onde a distribuição do histograma é bimodal.

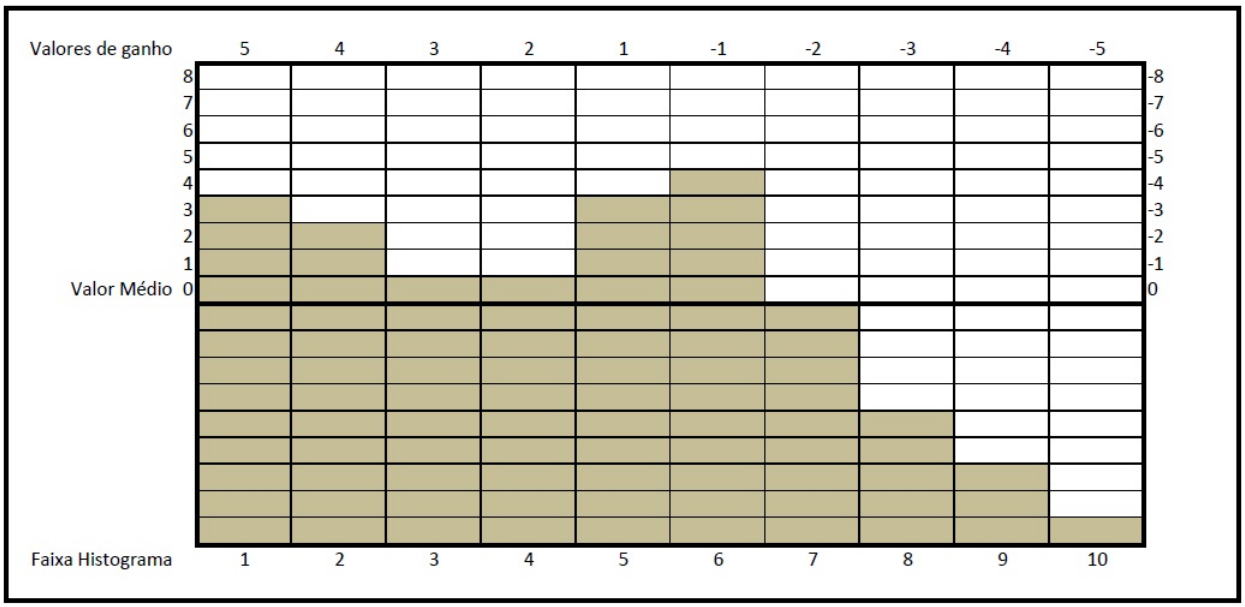

Figura 4.2: Cálculo do Erro Através da Análise do Histograma

As vantagens deste tipo de abordagem sáo várias. Ela é independente do tipo de sensor, velocidade do *hardware* e pode ser implementada em sistemas que possuem sensores CMOS ou CCD. É um algoritmo que pode ser adaptado para valores de pixels com diferentes conversores Analógico-Digital. É um método que possui uma implementação genérica.

Implementação e ajuste do controlador PID

A figura 4.3 mostra a arquitetura do controlador PID implementado. Ele utiliza 3 registradores, 2 adicionadores, 3 multiplicadores e um subtrador. A aritmética das multiplicações foi realizada baseada em registradores e instruções de rotacionamento de bits (*shift registers*, portanto as multiplicações e divisões foram baseadas em potências de 2.

A figura 4.4 mostra a arquitetura do circuito implementado para o calculo do erro através da análise do histograma. Este circuito utiliza 20 registradores, 1 adicionador e 10 multiplicadores. A montagem do histograma e o cálculo do erro são feitos em paralelo a leitura dos dados do sensor. À medida que os dados do sensor CMOS são aquisitados, eles são utilizados para formar o histograma e para o cálculo de erro. Como a leitura do sensor é feita através de leituras de suas linhas, cada linha é computada pelo circuito, e quando todas as linhas são lidas, e o sinal de frame válido é encontrado, indicando que a

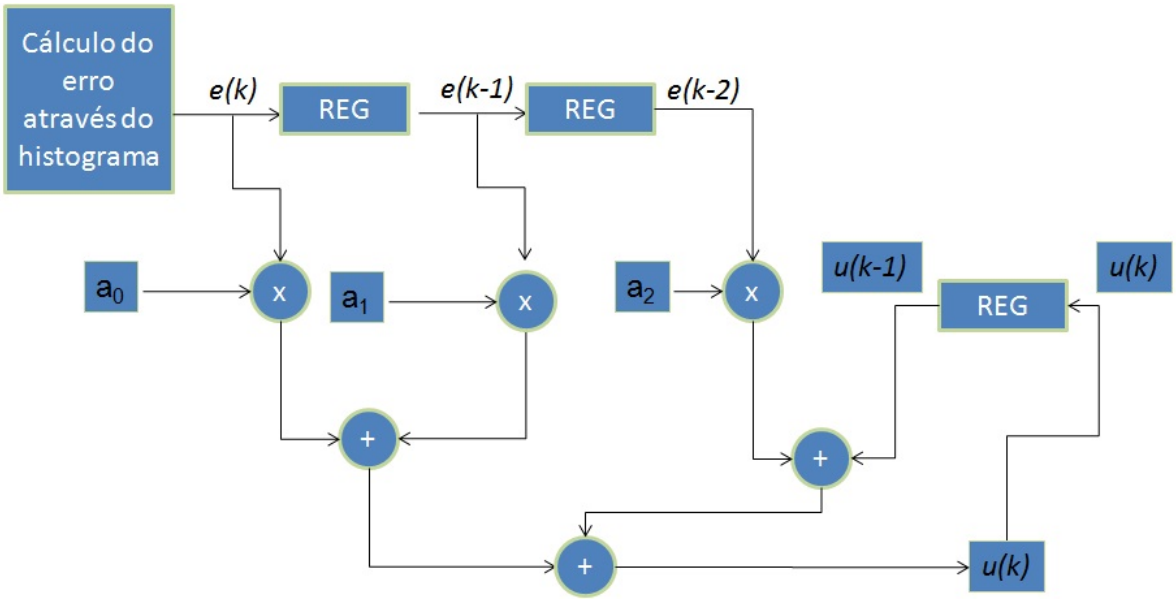

Figura 4.3: Arquitetura simplificada do controlador PID

transmissão de todas as linhas foi bem sucedida, o cálculo do erro final necessita apenas de um ciclo de relógio para somar todos os registradores de erro para calcular o erro total. Portanto, devido à velocidade de processamento e ao paralelismo de suas operações, isto permite que este circuito possa ser implementado dentro de um sensor CMOS inteligente. A limitação de tempo neste trabalho foi a linha de comunicação I2C, que necessita de um tempo aproximadamente 160 vezes maior para transmitir o novo valor de ganho do que o tempo necessário para o circuito configurado calcular este valor. Para maiores informações sobre a leitura e a comunicação entre a FPGA e o sensor, consultar o manual do sensor *THDB D5M Hardware specification*.

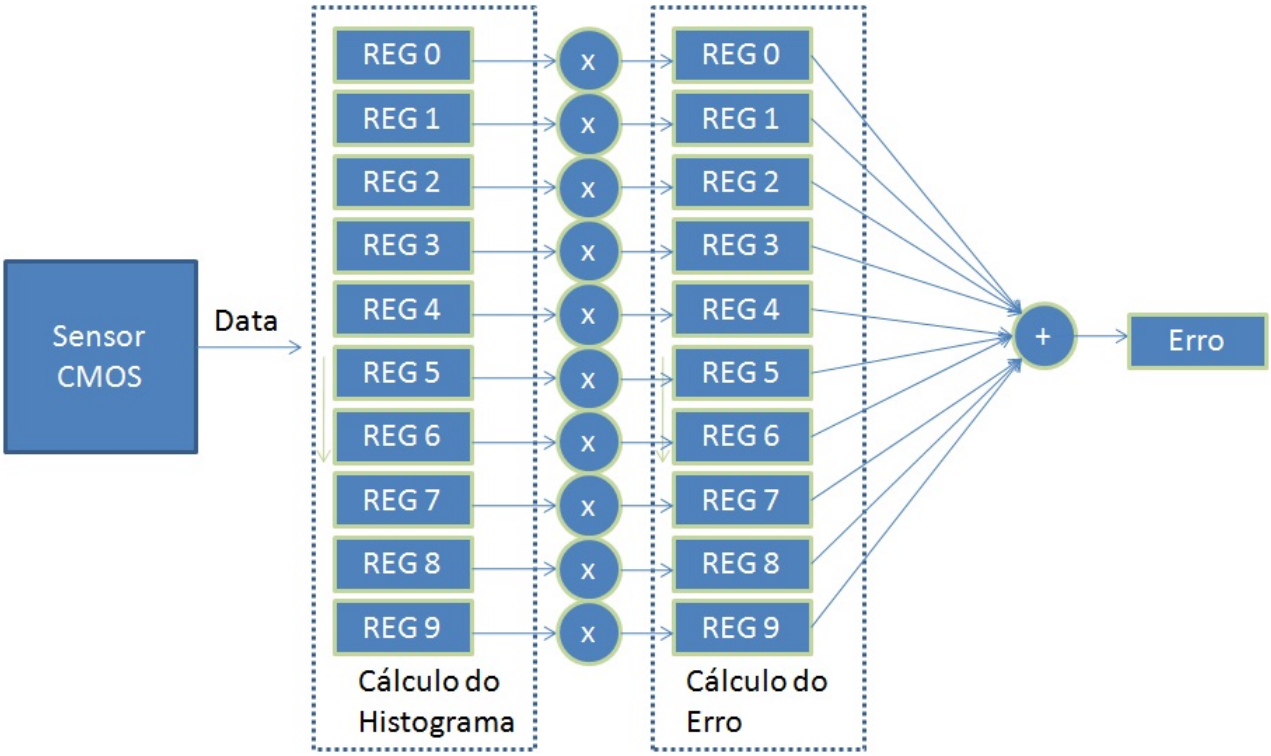

Figura 4.4: Arquitetura simplificada do cálculo do histograma e cálculo do erro.

Capítulo

5

Resultados do Sistema Proposto

O sistema foi exaustivamente testado e os resultados são mostrados a seguir. O controlador demonstrou um bom comportamento em diferentes tipos de ambiente e para a sintonia do controlador o sistema foi submetido a uma mudança de luz ambiente. Primeiramente o ambiente estava escuro, e uma luz foi acendida para analise da resposta do sistema.

Inicialmente somente a parte proporcional do controlador foi analisada.

O controlador PID foi sintonizado empiricamente e o sinal de controle é apresentado nos graficos desta secão. Nesta fase de testes, por motivos de implementação, os valores de ganho foram modificados a partir da formula 5.1.

$$k = 1/2^n \tag{5.1}$$

Os valores dos pârametros do controlador PID, *kc*, *ti* e *td*, foram ajustados utilizando operações de deslocamento registradores ou *shift registers*. Estes parâmetros podem variar de 1 até 2.33E-10, enquanto n varia de 0 até 32. A resposta do sistema (sinal de controle, u) pode variar apenas entre 8 e 127, que, de acordo com o manual da câmera, são os valores plausíveis para o registrador de ganho do sensor. Os valores de ganho do controlador foram acquisitados utlizando o analizador de comunicação I2C *Beagle I2C*

Protocol Analyzer (http://www.totalphase.com, 2011). Os testes foram realizados utilizando uma frequência de *clock* de 50MHz para a FPGA e uma velocidade de comunicação de 100Kbps para a linha de comunicação I2C. Primeiramente somente a parte proporcional do controlador foi testada. A figura 5.1 mostra diferentes respostas do sistema para diferentes ganhos proporcionais. Os testes foram realizados utilizando valores de n que variaram de 1 a 32, cobrindo assim todos os valores possíveis. O sistema demonstrou um comportamento estável com valores variando entre 2.33E-10 a 6.1E-05, o que significa n variando entre 14 a 32. Para valores de kc maiores que 1.2E-03 (n=13) o sistema demonstrou um comportamento instável.

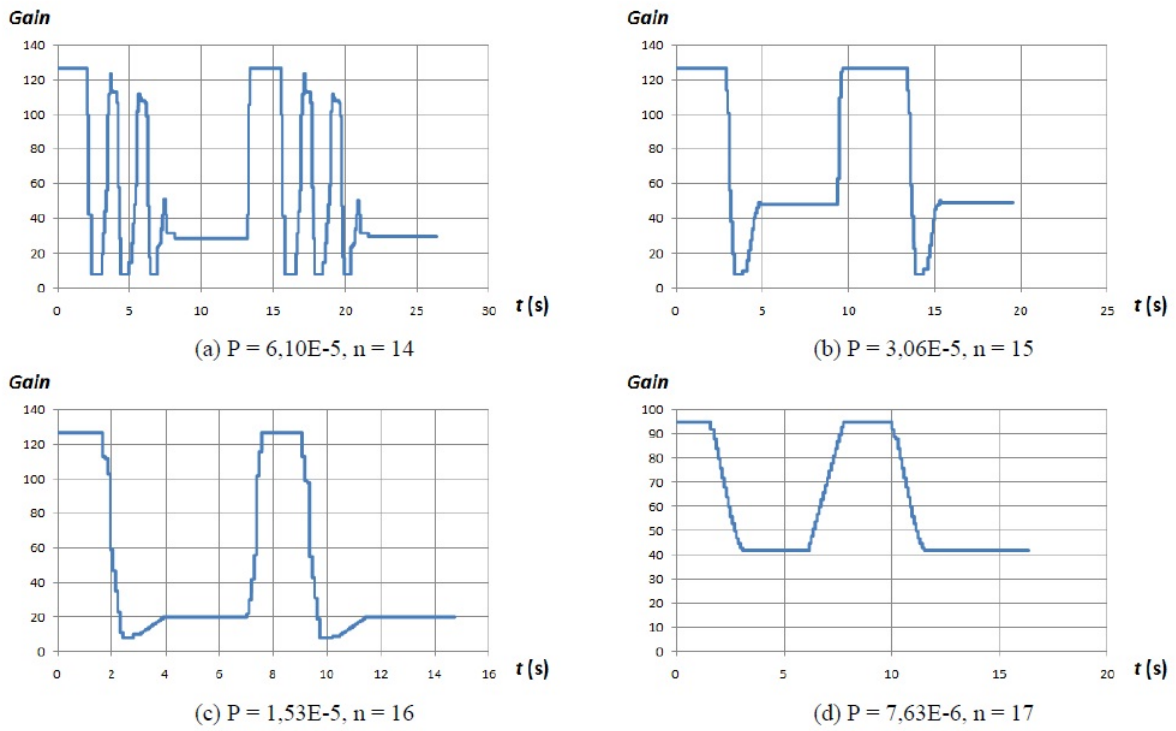

Figura 5.1: Resposta do Sistema com variação do Ganho Proporcional

Após o ajuste da parcela proporcional do controlador, a parcela integral foi analisada. Para a análise, foi escolhido o ganho proporcional *kp* de 1,53E-05, que demonstrou ser o melhor valor do ganho proporcional ajustado ao sistema. Os valores de ganho integral n variaram de 1 a 32, cobrindo assim todos os valores possíveis. O sistema demonstrou um comportamento estável com valores de *ti* variando entre 0,125 a 6.1E-05, o que significa n variando entre 3 a 32. Para valores de kc maiores que 0,25 (n=2) o sistema demonstrou um comportamento instável. A figura 5.2 mostra diferentes respostas do sistema para

diferentes ganhos integrais. Podemos observar a influência da sintonia do parâmetro ti na resposta do sistema. Na figura 5.2(a) observamos uma resposta oscilatória numa tendência de instabilidade. Isto pode ser justificado pelo fato de que, o zero do controlador começa a se afastar demasiadamente do pólo na origem e o controlador tende a comportar-se como um integrador puro. A medida que a ação integral é diminuída, e a resposta do sistema tende a se estabilizar mais rapidamente, isto significa que o erro em regime tende a ser anulado mais rapidamente.

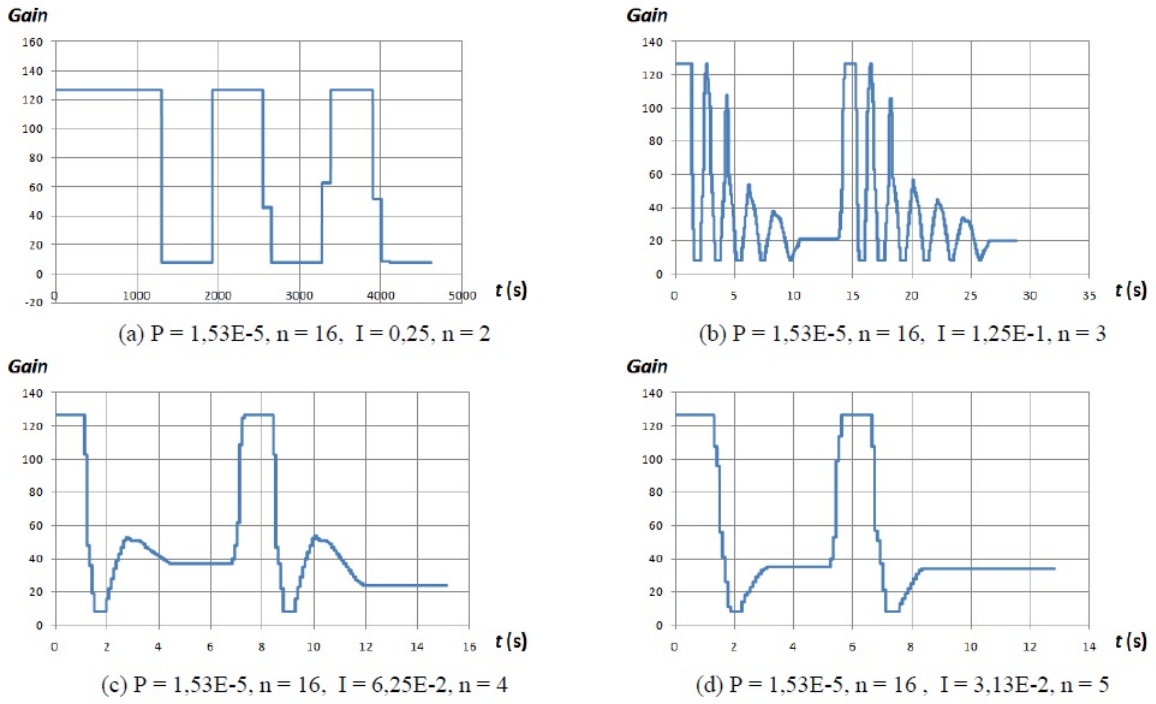

Figura 5.2: Resposta do Sistema com variação do Ganho Integral

A última parcela do controlador, a parte derivativa *td*, foi ajustada após o ajuste das parcelas proporcional *kp* e integral *ti*.

Os valores de ganho derivativo n variaram de 1 a 32, cobrindo assim todos os valores possíveis. O sistema demonstrou um comportamento estável com valores de td variando entre 0,5 a 2.33E-10, o que significa n variando entre 1 a 32. Para valores de td maiores que 0,5 (n=1) o sistema demonstrou um comportamento instável. O controlador proporcional combina as vantagens do controlador PI e PD. A ação integral está diretamente ligada à precisão do sistema sendo responsável pelo erro nulo em regime permanente. O efeito desestabilizador do controlador PI é contrabalançado pela ação derivativa que tende a aumentar a estabilidade relativa do sistema ao mesmo tempo que torna a resposta do

sistema mais rápida devido ao seu efeito antecipatório (Silva, 2000).

A figura 5.3 mostra o resultado da resposta do sistema com a variação do ganho derivativo.

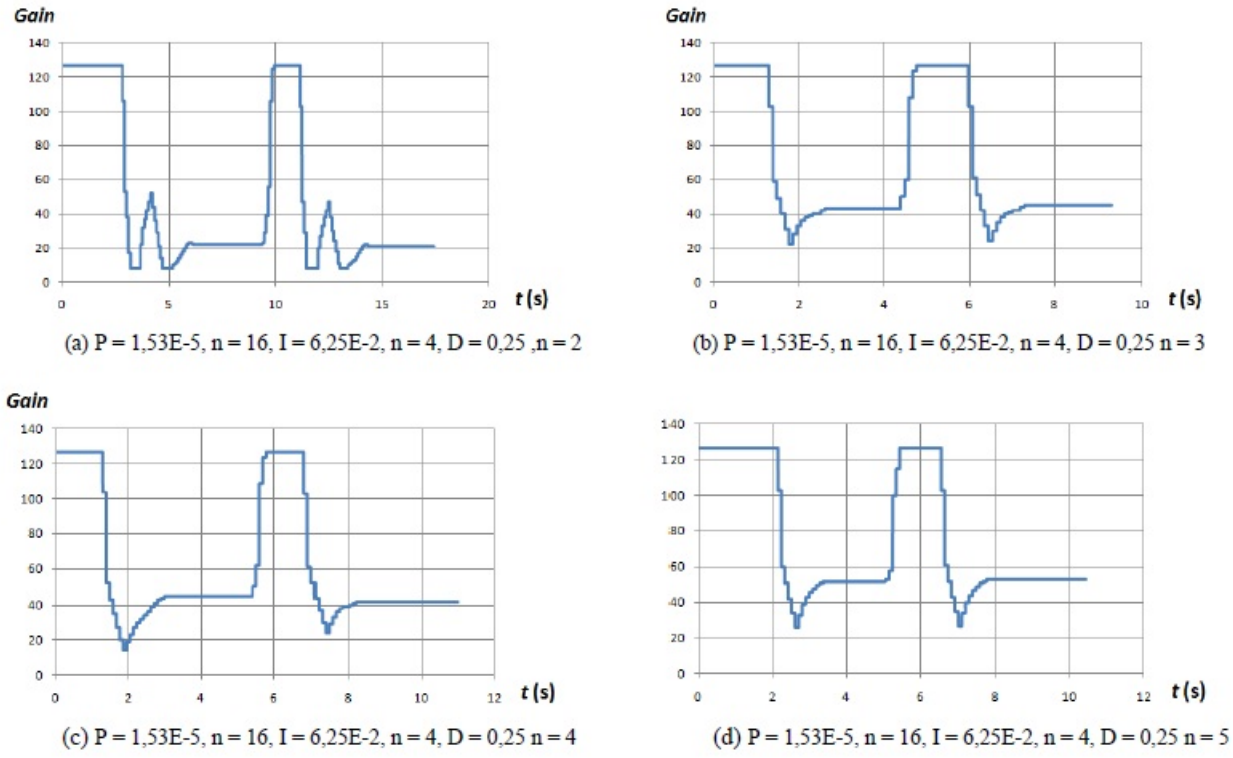

Figura 5.3: Resposta do Sistema com variação do Ganho Derivativo

Capítulo

6

Conclusão

O trabalho provou ser possível a aplicação de um controlador PID ao controle de ganho de um sensor CMOS. O algoritmo de cálculo do erro demonstrou ser uma ferramenta eficiente nesta aplicação.

O recursos do Kit de desenvolvimento DE2 70 da Terasic foram suficientes para a implementação do sistema proposto, e as ferramentas de desenvolvimento, assim como o projeto base fornecido pela Terasic, foram de suma importância no desenvolvimento do sistema proposto, pois além de acelerar a implementação, o trabalho pode ser concentrado no módulo específico do controlador. A implementação do *hardware* utilizando a linguagem Verilog, demonstrou-se eficiente, e o projeto proposto foi implementado. A qualidade da imagem obtida utilizando o controlador demonstrou-se satisfatória, e podemos afirmar que este tipo de algoritimo de cálculo de erro, utilizado a um controlador PID pode servir de base para o ajuste de sensores em diversas aplicações. O algoritmo de cálculo de erro através da análise do histograma pode ser visto como a principal inovação neste projeto de mestrado, pois ocontroladores PID já foram implementados para aplicações como controle de motores (Giorgini, 2001) , atuadores (et al., 2010), etc. Durante o desenvolvimento e a implemetação do sistema SoC *System on a Chip* foram encontrados desafios e problemas que difícilmente são encontrados em desenvolvimento de softwares embarcados

onde a plataforma utlizada é baseada em microcontroladores, como tempo de propagação de portas lógicas, alto tempo de compilação e principalmente, dificuldade para encontrar erros do sistema. Porém os resultados obtidos com o uso desta tecnologia, permitiu uma implementação que possui uma resposta extremamente rápida, que dificilmente seria conseguida utilizando uma tecnologia de microcontroladores de propósito geral.

Finalmente, o trabalho envolveu várias áreas de conhecimento, tais como aquisição de imagens, montagem e entendimento de histogramas, conhecimento de controladores digitais PI, PD e PID, sintonida de controladores digitais, linguagens de descrição de hardware, principalmente Verilog e conhecimento da tecnologia de *hardware* reconfigurável (FPGAs), além dos conhecimentos adquiridos através das disciplinas cursadas no ICMC, que acrescentaram um conhecimento muito grande ao mestrando, numa área de crescente demanda e carente de recursos no país.

Capítulo 7

Trabalhos Futuros

Os seguintes trabalhos futuros são propostas para melhorias do trabalho realizado:

- Ajuste do controlador PID através de métodos como Zigler-Nichols, uma vez que o ajuste do controlador foi efetuado empiricamente.
- Novos ajustes do algoritmo de cálculo de erro, como o aumento de faixas de valores do histograma e parametrização de valores de ganho, podem promover melhorias no sistema proposto, uma vez que o algoritmo de erro é a base para o controle da imagem.
- Este trabalho também pode ser implementado em outros sistemas de tempo real, para que possa ser analisado o seu desempenho em outras aplicações, tais como automóveis, VANTS, sistemas de segurança, ou ainda como um hardware dedicado já implementado juntamente a sensores inteligentes, que já possuem um *hardware* dedicado à uma determinada aplicação.
- A limitação de tempo do sistema foi a linha de comunicação I2C. A mesma abordagem poderia ser realizada em outra plataforma de hardware, ou com outro sensor que possua um tempo de realimentação mais rápido, utilizando um barramento de comunicação tipo AMBA.

- Implementação de um sistema de aquisição de dados que possa capturar as imagens do sensor para uma futura análise quantitativa da qualidade do controlador e ajustes mais precisos.

- Implementar o sistema proposto no projeto de detecção de pedestres para que ele cumpra a sua tarefa de ajudar outros alunos do grupo do LCR em suas pesquisas.

Referências Bibliográficas

A. El Gamal, H. E. Cmos image sensors. *IEEE Circuits and Device Magazine, Vol. 21,*, pgs. pp. 6–20, 2005.

Ahmed, I. *Implementation of pid and deadbeat controllers with the tms320 family.* Relatório técnico, Texas Instruments, 1997.

et al., R. O.-R. The application of reconfigurable logic to high speed cnc milling machines controllers. *Control Engineering Practice*, 2010.

Altera Gnu tools, 2003a.
Disponível em `http://www.altera.com/products/software/embedded/products/gnupro` (Acessado em 2003)

Altera Introduction to quartus II, 2003b.
Disponível em `http://www.altera.com/literature/manual/intro` (Acessado em 07/2003)

Altera Nios 3.0, 2003c.
Disponível em `http://www.altera.com/literature/ds/ds/nioscpu.pdf` (Acessado em 11/2003)

Altera Nios development kit, cyclone ii edition, 2009.
Disponível em `http://www.terasic.com.tw/cgi-bin` (Acessado em 03/2010)

Blanc, N. *CCD versus CMOS-has CCD imaging come to an end.* Relatório técnico pp. 131-137, Herbert Wichmann Verlag, Heidelberg, 2001.

Bobda, C. *Introduction to reconfigurable computing.* 2007.

Boyle, W.; Smith, G. *Charge coupled semiconductor devices.* Relatório técnico vol 49 pp. 587, Bell Systems Technical Journal, 1970.

Cardoso, Fabbryccio A. C. M.; Fernandes, M. A. C. Ie-334b - tópicos em comunicações - fpga e e fluxo de projeto, 2007.
Disponível em `http://www.decom.fee.unicamp.br/~cardoso/ie344b/Introducao_FPGA_Fluxo_de_Projeto.pdf`

Company, E. K. CMOS vs CCD and future of imaging, 2003.
Disponível em `http://www.kodak.com` (Acessado em 05/2003)

Company, E. K. Charge-coupled device CCD image sensors, 2008a.
Disponível em `http://www.kodak.com` (Acessado em Revision 2.0 MTD/PS-0218 p5)

Company, E. K. *Kodak image sensors users manual - revision 2.0 mtd/ps-0218.* Relatório técnico, Kodak, 2008b.

Filho, O. M.; Neto, H. V. *Processamento digital de imagens.* Rio de Janeiro: Brasport Livros e Multimídia, 1999.

Flora M. Li, A. N. *Ccd image sensors in deep-ultraviolet degradation behavior and damage mechanisms.* Waterloo Canada: Springer, 2004.

Giorgini, A. L. *Implementação de um controlador pid digital para robótica baseado em computação reconfigurável.* Dissertação de mestrado, 2001.

Hauck, S. *Reconfigurable computing the theory and practice of fpga based computation.* Elsevier, 2008.

de Holanda, J. A. M. *Proposta de uma arquitetura multi-core para detecção de pedestres baseada em câmeras.* Tese de doutorado, 2010.

HTTP://SENSORCLEANING.COM *http://sensorcleaning.com/whatisasensor.php.* Relatório técnico, sensorcleaning.com, 2010.

HTTP://WWW.TERASIC.COM.TW *De2 70 camera utility support qaurtus ii 10.0.* Relatório técnico, http://www.terasic.com.tw, http://www.terasic.com.tw, 2010.

HTTP://WWW.TOTALPHASE.COM *Beagle i2c/spi protocol analyzer.* http://www.totalphase.com/products/beagle$_i sm/$, 2011.

IRFAN, A. *Implementation of pid and deadbeat controllers with the tms320 family.* Relatório técnico, 1997.

ISERMANN, R. *Digital control systems, vol.1: Fundamentals, deterministic control,.* Springer-Verlag Berlin Heidelberg, 1989.

J. G. ZIEGLER, N. B. N. Optimum settings for automatic controllers. *Transactions on ASME*, pgs. 759–768, 1942.

K. ÅSTRÖM, T. H. *pid controllers: Theory, design, and tun-ing, in instrument society of america.* Research Triangle Park, 1995.

KLYOHARU ALZAWA, KATSUHIKO SAKAUE, Y. S. *Image processing technologies, algorithms, sensors, and applications.* MARCEL DEKKER, INC, 2004.

K.NOHSOH; S.OZAWA A simultaneous estimation of road structure and camera position from continuous road images. *IEICE Trans.Inf. Syst,vol.J76-D-II, no.3 pp.514-523 March 1993 (in Japanese).*, 1993.

L. MUHIDIN, G. Z. A reference guide to pid controllers in the nineties. *IFAC Workshop on Digital Control: Past, present and future of PID Control*, pgs. 73–82, 2000.

LIMA, J. *Sintonia automática de controladores pid: uma abordagem neuro-genética.* Tese de doutorado, Portugal, 2004.

LITWILLER, D. *Cmos vc ccd: Maturing technologies, maturing markets.* Relatório técnico, Dalsa, 2005.

M. H. Moradi, M. A. J. New techniques for pid controller design. *Proceedings of 2003 IEEE Conference on Control Applications*, pgs. 903–908, 2010.

M. H. Moradi, M. R. Katebi, M. A. J. The mimo predictive pid controller design. *Advances in PID Control Special Issue of Asian Journal of Control*, vol. 4, no. 452 - 463, 2002.

M. Morari, E. Z. *Robust process control.* Prentice Hall, 1989.

M. R. Katebi, M. H. M. Predictive pid controllers. *IEE Proceedings of Control Theory Application*, vol. 148, no. 6, pg. 478 487, 2001.

Moon, P.; Spencer, D. E. Geometric formulation of classical color harmony. 1944.

Nadi R. Itani, Caiyi Wang, D. R. W. United states patent us 7,522,193 b2. 2009.

Ohta, J. *Smart cmos image sensors and applications.* CRC Press, 2008.

de Oregon, U. Evolving towards the perfect CCD, 2003.
Disponível em `http://zebu.uoregon.edu/ccd.html` (Acessado em 06/2003)

Q. G. Wang, C. C. Hang, X. P. Y. Single-loop controller design via imc principles. *In Proceeding Asian Control Conference*, 2000.

R. Drausio, M. Eduardo, B. V. L. J. A pid controller applied to the gain control of a cmos camera using reconfigurable computing. *Reconfig 2011*, 2011.

Silva, J. M. G. *Ajuste de controladores pid.* http://www.ece.ufrgs.br/ jmgomes/pid/A-postila/apostila/apostila.html, 2000.

Åström, K. J.; Hägglund, T. *Advanced pid control, isa instrumentation, systems and automation society.* 2006.

Taylor, S. A. *CCD and CMOS imaging array technologies: Technology review.* Relatório técnico EPC-1998-106, Cambridge Laboratory, 1998.

Technology, M. I. 2003.
Disponível em `http://www.micron.com` (Acessado em 06/2003)

T.Kobayashi, N.; S.Ozawa Analysis of a road image seen from a vechicle -a study of the superelevation. *IEICE Trans.Inf.Syst.,Vol.J73-D-II,No.l pp. 125-129 January 1990 (in Japanese).*, 1990.

www.circuitstoday.com *Charge coupled devices (ccd).* Relatório técnico, www.circuitstoday.com, http://www.circuitstoday.com/charge-coupled-devices-ccd, 2010.

www.terasic.com *Terasic d5m hardware specification.* Relatório técnico, Terasic, 2008.