

# O CPLD (Dispositivo Complexo de Lógica Programação aplicado em automação industrial\*

*Tiago Tobias Freitas, Thiago Luiz Pasqualinoto e Juliano Carlos Leão*

\*Trabalho apresentado para “Feira SENAI Paulista de Inovação Tecnológica - INOVASENAI 2005”, na categoria Equipamento. Realizado pelos alunos do curso técnico em “Eletrônica-Automação da Manufatura” turma 2004/2005 do Centro de Treinamento SENAI/Lençóis Paulista-SP.

## RESUMO

Amplamente aplicada em varias áreas como a de telecomunicações e informática, a tecnologia dos PLDs (Programmable Logic Device), mostra características que favorecem seu uso em automação industrial. Partindo desse princípio analisamos as evoluções dessa tecnologia e dispositivos que compõem esse grupo, visando encontrar o dispositivo que nos permita desenvolver um controlador lógico versátil, eficiente e de baixo custo para automação de uma planta didática que simula um processo industrial. Desta forma, escolhemos para o desenvolvimento do controlador um CPLD (Complex Programmable Logic Device) da família MAX 3000A fabricado pela Altera Corp., ideal para o processo que automatizamos na planta didática escolhida para demonstração, pois suas capacidades e arquitetura se fizeram suficiente para tal.

Mostra também a poderosa ferramenta EDA (Electronic Design Automation) de programação Quartus II®, que desenvolvemos, compilamos e simulamos os circuitos lógicos que foram implementados no chip em questão. Para carregarmos o dispositivo com a programação dos circuitos lógicos usamos o Kit – PLD Versão Altera MAX 3000A construído sobre uma plataforma CNZ.

Como resultados práticos, obteve-se um controlador lógico tão confiável e eficaz quanto os controladores lógicos já existentes no mercado, e com um custo bem inferior à tecnologia mais usada atualmente. Esses resultados nos levam a concluir que essa tecnologia surge como uma nova vertente para automação industrial, trazendo uma nova opção para indústria e criando um novo artifício para os técnicos em automação, que por exigência de mercado precisam ser cada vez mais versáteis e qualificados.

## **1 Introdução**

Para as industrias de pequeno e médio porte, é importante que as novas tecnologias sejam tecnicamente possível e economicamente viável. Desta forma, a busca por controladores de baixo custo, fácil utilização e de programação que mostre uma interface amigável e de linguagem moderna tornou-se um fator de grande importância para investimentos em automação industrial.

Apoiada em poderosas ferramentas Eletronic Design Automation (EDA) que aceleram o ciclo de desenvolvimento, a tecnologia dos Dispositivo de Lógica Programável Complexa (CPLD), mostra as características necessárias para o desenvolvimento de controladores lógicos programáveis como, alta durabilidade (mesmo em ambientes agressivos), alta versatilidade (pode ser re-configurado cerca de 1.000.000 de vezes), velocidade de resposta (pode responder na ordem de nano segundos) e facilidade de programação no ambiente windows isso com a mesma eficiência dos controladores já existentes no mercado.

Tais características contribuem para que a tecnologia CPLD torne-se de uso crescente e irreversível, somando-se a isso a vantagem de re-configuração do hardware e principalmente o baixo custo, justifica a pesquisa desta tecnologia na aplicação de automação industrial de pequeno e médio porte melhorando a relação custo x beneficio em comparação com uma das tecnologias mais utilizadas atualmente, o Controlador Lógico Programável (CLP).

Como será apresentado nessa pesquisa, trata-se de uma tecnologia emergente e promissora, apontada como um novo recurso pelos pesquisadores mais entusiastas, que oferece, além dos benefícios de baixo custo, um enorme potencial para desenvolver novas aplicações em automação industrial. Essa pesquisa pretende explorar as técnicas de configuração dos CPLDs da família MAX 3000 – Altera, visando uma analise precisa do seu desempenho como controlador lógico programável.

O resultado obtido com esta pesquisa será aplicada em uma planta simuladora de um processo industrial automatizado, que deverá usar a tecnologia CPLD para o controle da automatização do processo.

## **1.1 Objetivo**

O Principal objetivo foi desenvolver um controlador para automação industrial tão eficiente quanto os já existentes no mercado, com um custo reduzido, utilizando para isso uma tecnologia simples, com linguagem de programação de fácil compreensão e alta velocidade de resposta aos sinais de entrada.

Para demonstrar de forma prática o controlador que foi desenvolvido, foi também objetivo automatizar uma planta didática que simula um processo industrial. Por consequência esperamos criar através desse estudo uma nova possibilidade para a área de automação industrial sem desqualificar qualquer outra tecnologia, apenas trazendo para o mercado uma nova opção.

## **1.2 Justificativa**

A tecnologia CPLD é amplamente aplicada em varias áreas como a de telecomunicações e informática, somando-se a isso a vantagem de re-configuração do hardware e principalmente o baixo custo, justifica-se a pesquisa desta tecnologia na aplicação de automação industrial de pequeno e médio porte melhorando a relação custo x beneficio em comparação com uma das tecnologias mais utilizadas atualmente, o CLP (Controlador Lógico Programável).

## **2 Tecnologia dos Dispositivos Lógicos Programáveis.**

### **2.1 Os Circuitos Digitais**

Evoluindo de maneira assustadora nas ultimas décadas, os circuitos digitais ainda passam por constantes mudanças, que vêm transformando o processo de projeto de hardware. Os controladores usados em automação são diretamente beneficiados por estas mudanças, pois o surgimento de novas possibilidades nos circuitos digitais criam novas vertentes e tendências de projeto para automação.

Os componentes dos circuitos digitais evoluíram de transistores individuais para circuitos integrados VLSI (very large scale integration). A utilização de ferramentas EDA (electronic design automation) tem

simplificado e acelerado todo o ciclo de projeto. Atualmente, não é mais necessário desenhar portas lógicas individuais e planejar todas suas interconexões. As linguagens de descrição de hardware (HDLs) estão hoje consolidadas no meio acadêmico e industrial como forma padrão na elaboração de projetos. Existem também, ferramentas de síntese lógica automática, disponíveis para mapear circuitos em diversas tecnologias (CHAN, 1994, p.256).

A simplificação e o aceleração do ciclo de projeto têm exigido uma prototipação cada vez mais rápida, pois o tempo necessário para o desenvolvimento do projeto tem ficado cada vez mais longo em relação ao lançamento de novas tecnologias.

## 2.2 Tecnologias para Projetos de Sistemas Digitais

Existem diversas tecnologias utilizadas na construção dos circuitos integrados digitais, por esta razão a escolha de uma tecnologia depende das exigências do projeto a ser realizado. A Figura 2.1 mostra como podem ser agrupadas as categorias de projetos.

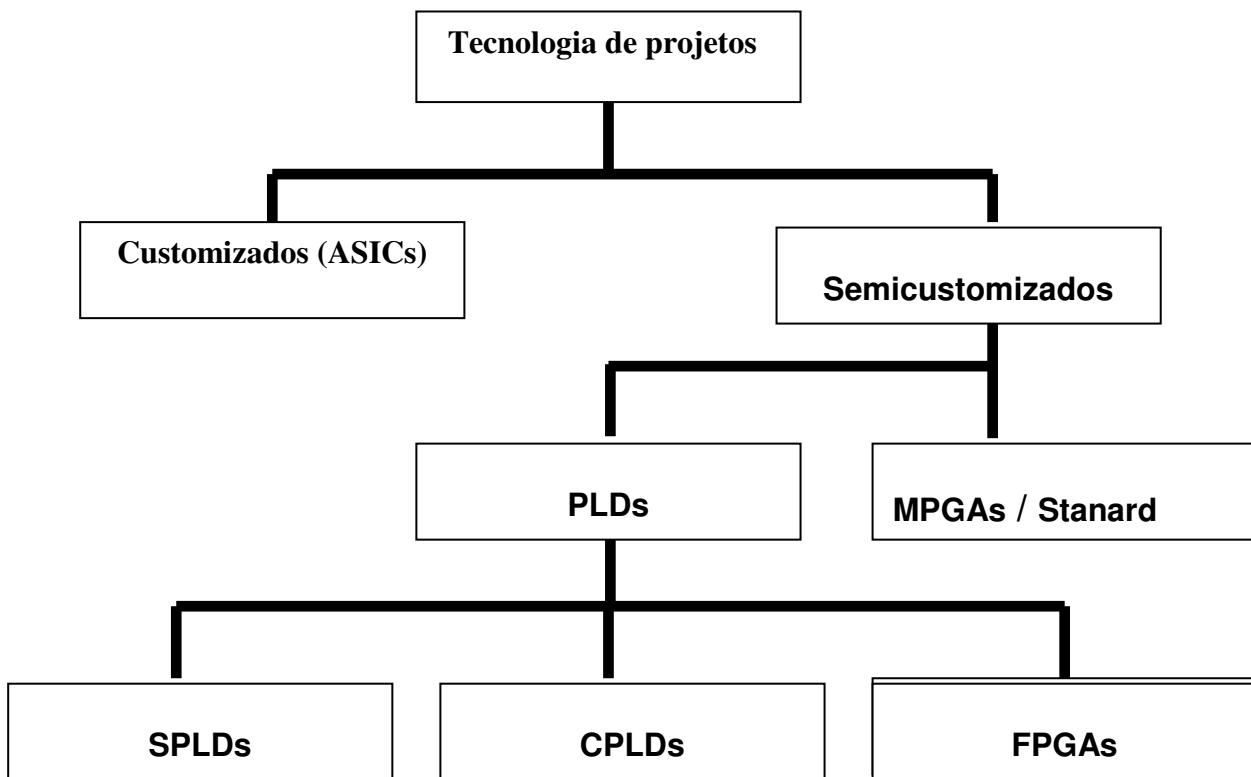


Figura 2.1 – Tecnologias para projetos de sistemas digitais

### **2.2.1 ASICs (Application Specific IC)**

Os ASICs ou Cls customizados, se caracterizam principalmente pela necessidade de um processo de fabricação especial, que requer máscaras específicas para cada projeto. Essa característica acarreta em altos custos de projeto e um longo tempo de desenvolvimento para esse tipo de implementação, em caso de grandes implementações esse alto custo é amortizado.

### **2.2.2 MPGAs (Mask Programmable Gate Arrays)**

A fabricação nesse tipo de implementação é agilizada pelo uso de máscaras genéricas de módulos pré-projetados. A necessidade de máscaras específicas para a interconexão dos módulos é um problema no uso desta tecnologia, em contra partida o projeto é facilitado por uma biblioteca de células, proporcionando um tempo de desenvolvimento e custos menores em relação aos Cls customizados.

### **2.2.3 Standard Cells**

Semelhante aos MPGAs as Standard Cells também tem os projetos facilitados pelo uso de módulos pré-projetados. Os módulos Standard Cells são na maioria das vezes armazenados em bancos de dados, onde o projetista seleciona as células desejadas para a realização do seu projeto. Comparando com os Cls customizados os circuitos implementados em Standard Cells têm um custo de desenvolvimento mais baixo, porém, são menos eficientes em tamanho e desempenho.

### **2.2.4 PLDs (Programmable Logic Device)**

Os PLDs apresentam como característica principal à capacidade de configuração pós-fabricação feita pelo usuário, tornando circuitos implementados como essa tecnologia projetos de fácil modificação. Os custos de desenvolvimentos

e ciclo de tempo do projeto são extremamente baixos quando comparados às outras tecnologias.

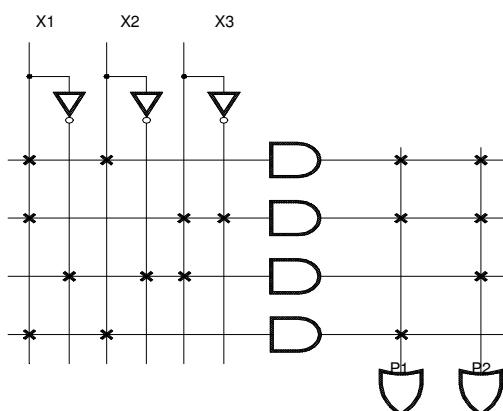
O mercado de PLDs encontra-se em plena expansão, de forma que atualmente existem diversos fabricantes e modelos de dispositivos desse tipo. Uma das principais tarefas do projetista hoje é pesquisar e selecionar, dentre as opções disponíveis no mercado, qual a que melhor atende suas necessidades (RIBEIRO, 2002, p.11).

### 3 Evolução dos Dispositivos Lógicos Programáveis.

#### 3.1 O Primeiro Dispositivo Lógico Programável

A memória PROM (*programmable read-only memory*), foi o primeiro tipo de *chip* programável pelo usuário, que podia implementar circuitos lógicos. As linhas de endereço eram utilizadas como entradas do circuito lógico, e as linhas de dados como saídas desses circuitos. As funções lógicas, entretanto, raramente requerem mais que alguns termos de produto, e uma PROM contém um decodificador completo para seus endereços de entradas. As PROMs, portanto, se mostram como uma arquitetura ineficiente para realização de circuitos lógicos, e são muito raramente utilizadas para esse fim (BROWN, 1996, p.42).

Com a ineficiência de sua arquitetura logo ficou constatado que as PROMs apenas serviram de inspiração para outros dispositivos introduzidos pela Phillips no inicio dos anos 70, os PLAs. A figura 3.1 mostra o esquema de um PLA.



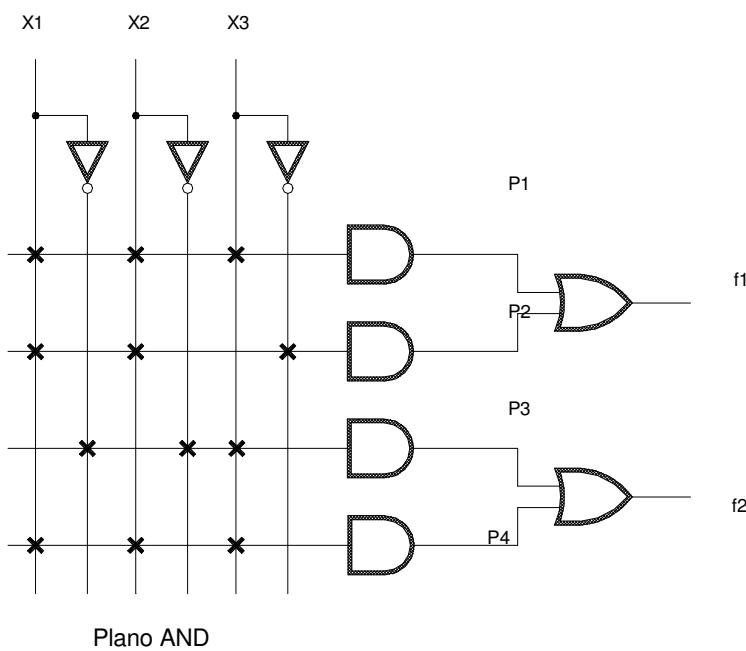
**Figura 3.1 - Esquema simplificado de um PLA.**

Foram os PLAs (Programmable Logic Arrays) os primeiros dispositivos desenvolvidos especificamente para implementação de circuitos lógicos. Estes dispositivos consistem de dois níveis de portas lógicas: um plano de portas

*wired-AND* seguido por um plano de portas *wired-OR*, ambos programáveis. Um PLA é estruturado de tal forma que cada saída do plano AND pode corresponder a qualquer termo produto das entradas. Da mesma forma, cada saída do plano OR pode ser configurada para produzir a soma lógica de quaisquer saídas do plano AND.

Essa estrutura faz dos PLAs dispositivos adequados para implementações de funções lógicas na forma de soma de produtos, além da versatilidade de possuir muitas entradas, tanto nos termos AND quanto nos termos OR. Porém, essa tecnologia apresenta desvantagens devido aos dois níveis de lógica configurável, gerando alto custo de fabricação e significativos atrasos de propagação dos sinais elétricos.

Para superar as deficiências encontradas nos PLAs foram desenvolvidas tecnologias que possuem um único nível de programação, custos mais baixos e melhor desempenho. Essa tecnologia é conhecida como PAL (Programmable Array Logic) e possui na sua estrutura apenas um único nível de programação, constituindo de um plano de portas AND programáveis que alimenta um plano OR fixo. Diversos modelos de PALs foram construídos para compensar a inflexibilidade do plano OR fixo. A figura 3.2 mostra o esquema simplificado de um PAL.



**Figura 3.2 -** Esquema simplificado de um PAL.

Os PALs geralmente contem flip-flops conectados as saídas das portas OR para que circuitos seqüências possam ser implementados. Dispositivos PALs foram importantes pois, quando introduzidos, tiveram um profundo efeito no projeto de hardware digital, e também foram à base para algumas das novas e mais sofisticadas arquiteturas. Variantes da arquitetura básica do PAL são encontrados em outros produtos conhecidos por diferentes siglas. Todos os pequenos PLDs, como PLAs, PALs, e outros dispositivos similares são agrupados em uma única categoria chamada SPLD (Simple PLDs), cujas características mais importantes são o baixo custo e alto desempenho.

### 3.2 CPLD (Complex Programmable Logic Device)

A dificuldade de aumentar a capacidade da arquitetura dos SPLDs, devido à estrutura dos planos lógicos programáveis aumentarem muito rapidamente à medida que se aumenta o numero de entradas, impulsionou o avanço da tecnologia tornando possível à produção de dispositivos com maior capacidade que os SPLDs.

O único modo viável de produzir dispositivos com maior capacidade baseados na arquitetura SPLD foi integrar múltiplos SPLDs em um único chip e prover interconexão programável para os blocos SPLDs. Muitos produtos PLDs são encontrados no mercado atualmente com essa estrutura básica, e são coletivamente chamados de CPLDs (Complex PLDs). A figura 3.3 mostra a arquitetura interna da estrutura CPLD.

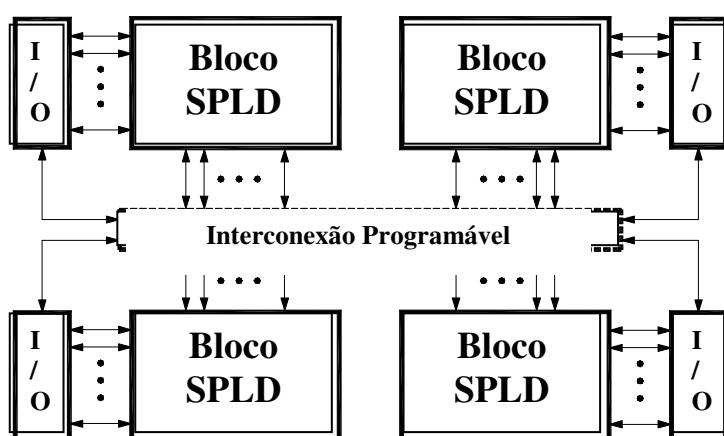


Figura 3.3 - Estrutura de um CPLD

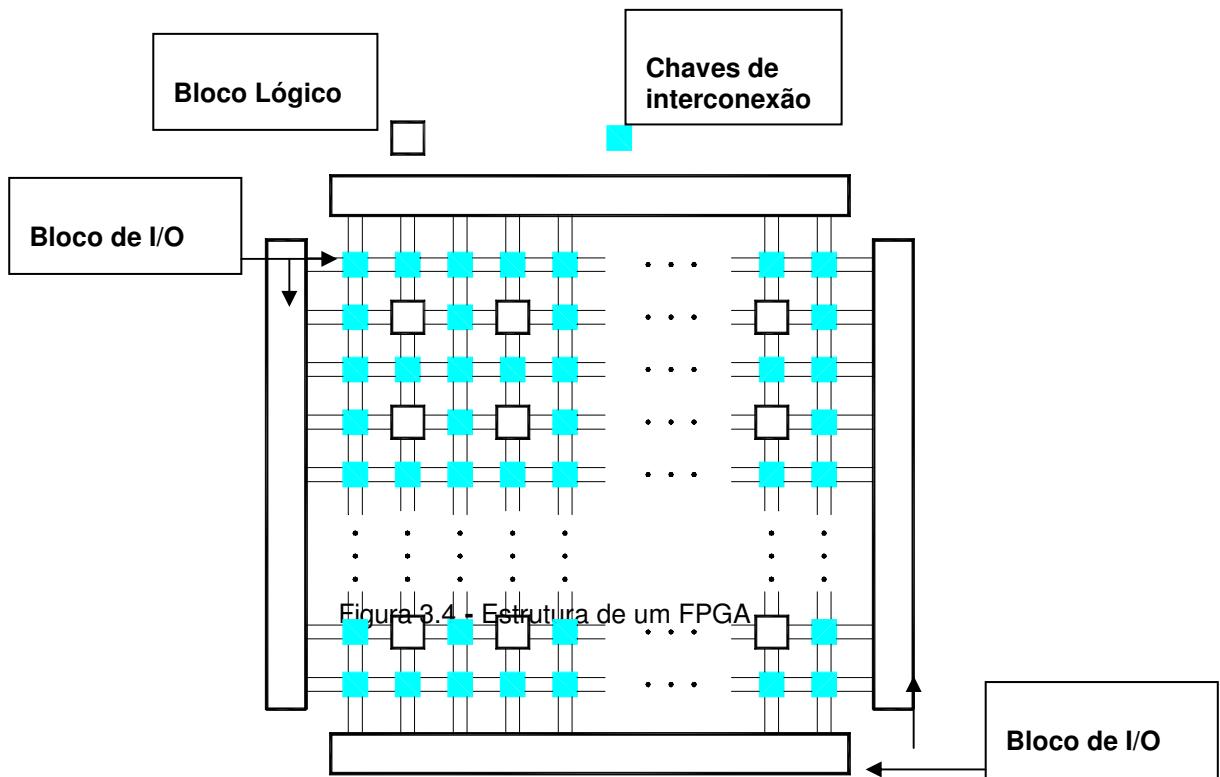
Os CPLDs foram introduzidos pela Altera Corp. com uma família de chips chamada Classic EPLDs (Erasable PLDs), logo em seguida refletindo o sucesso de mercado, outras séries adicionais foram lançadas como; a série MAX 3000, MAX 5000, MAX 7000 e MAX 9000. Devido ao rápido crescimento do mercado para PLDs outros fabricantes desenvolveram diversas opções de CPLDs, que atualmente provêm uma capacidade lógica de até 50 dispositivos SPLDs típicos. Baixo custo e alta velocidade são características destes dispositivos, porém é difícil extender essa arquitetura para maiores densidades.

### **3.3 FPGA (Field Programmable Gate Array)**

Um FPGA consiste de um grande arranjo de células configuráveis (ou blocos lógicos) contidos em um único chip. Cada uma dessas células contém certa capacidade computacional para implementar funções lógicas e/ou realizar roteamento para permitir a comunicação entre as células. Essas operações podem acontecer simultaneamente no arranjo das células.

A Xilinx Inc. em 1984 desenvolveu e disponibilizou comercialmente o primeiro dispositivo FPGA, desde então, diversos fabricantes têm desenvolvido este tipo de dispositivo. Atribui-se a isso o reduzido tempo de projeto e ao relativo baixo custo destes dispositivos programáveis de alta capacidade.

Os FPGAs não possuem planos de portas OR ou AND. Em vez disso, estes componentes consistem de um grande arranjo de células configuráveis (ou blocos lógicos) que podem ser utilizadas para a implementação de funções lógicas. A estrutura de um FPGA é ilustrada na Figura 3.1. Um FPGA possui três tipos principais de recursos: blocos lógicos, blocos de entrada e saída (I/O), e chaves de interconexão. Os blocos lógicos formam um arranjo bi-dimensional, e as chaves de interconexão são organizadas como canais de roteamento horizontal e vertical entre as linhas e colunas de blocos lógicos. Estes canais de roteamento possuem chaves programáveis que permitem conectar os blocos lógicos de maneira conveniente, em função das necessidades de cada projeto.



Encontra-se FPGAs comerciais disponíveis em varias arquiteturas, os principais aspectos que definem a arquitetura de um FPGA são:

- Tipo de tecnologia de programação;
- Arquitetura das células
- Estrutura de roteamento

Mesmo influenciando diretamente o desempenho e a densidade das arquiteturas FPGA, não podemos afirmar que uma arquitetura “A” é melhor que uma arquitetura “B”, mais sim que existem arquiteturas que se encaixam melhor em cada situação.

### 3.4 Analise dos Dispositivos

Cada tipo de PLD apresenta vantagens que os tornam mais adequados para algumas aplicações do que outros. Um projetista hoje depara-se com a difícil tarefa de pesquisar os diferentes tipos de *chips*, entender qual sua melhor utilização, escolher um fabricante específico, aprender a utilizar as ferramentas EDA, para só então começar a projetar o

hardware. Para demonstração de nossa pesquisa usaremos um CPLD da Altera Corp. família MAX3000, uma vez que o SPLD possui capacidade muito reduzida e o FPGA possui um custo mais elevado, pois sua capacidade é bem maior que a do CPLD, ultrapassando a necessidade do projeto.

Nos próximos capítulos serão apresentados os chips da família MAX3000, também o Kit de desenvolvimento e a ferramenta de programação usada para demonstração dessa tecnologia em automação industrial.

## 4 Família **MAX3000A**

### 4.1 Os Dispositivos da Família **MAX3000A**

Após uma análise da tecnologia PLD e suas evoluções, pesquisaremos especificamente os CPLDs da família MAX 3000A da Altera Corp. , buscando nessa família o dispositivo que usaremos na aplicação em automação industrial. O dispositivo EPM3064ALC44 –10, onde LC44 indica um encapsulamento PLCC 44 e –10 indica um atraso de até 10 nano – segundos através de uma macro célula, mostra condições favoráveis para o desenvolvimento do controlador com tecnologia CPLD, mas para decidir-mos por ele será necessário um estudo de toda a família.

Conhecendo a capacidade de cada dispositivo evitaremos super dimensionar o projeto restando apenas uma análise do processo a ser automatizado para definirmos o dispositivo que mais se aproxima das necessidades do projeto.

Esta família é composta pelos dispositivos EPM3032A, EPM3064A, EPM3128A EPM3256A e EPM3512A, os quais veremos detalhadamente neste capítulo, com o objetivo de usar um desses dispositivos como controlador lógico programável.

Uma das vantagens de desenvolvimento com esse tipo de tecnologia é o fato de não existir a necessidade de conhecer a fundo a arquitetura interna do dispositivo, pois o compilador faz todo o trabalho de configuração e conexões internas na estrutura do chip. Isso facilita e agiliza o desenvolvimento do hardware, pois o projetista usa uma linguagem de programação simples e amigável no ambiente Windows, e os arranjos lógicos baseados na tecnologia CMOS é feito pelo compilador na gravação do chip.

## **4.2 Características Gerais dos Dispositivos**

Por serem dispositivos da mesma família, os CPLDs Max3000A possuem a mesma arquitetura interna e mesmo método de desenvolvimento, tornando os dispositivos praticamente iguais em suas características. São características comuns a todos os dispositivos desta família:

- A alta performance e baixo consumo da tecnologia CMOS, baseados em EEPROM, mantêm a programação ao serem desligados, os dispositivos lógicos programáveis (PLDs) são construídos sobre arquitetura MAX®.
- Alta densidade dos PLD com ajuste de alcance de 600 até 10.000 portas utilizáveis.
- Tempo de atraso através de uma macro célula de 4.5 a 10 nano-segundos, com freqüências máxima de até 227,3 MHz.
- Suporta sem danos temperatura do ambiente industrial.
- Sistema MultiVoltTM I/O, permiti que o dispositivo seja alimentado com 3.3 V, enquanto os pinos de I/O são compatíveis com 5.0V, 3.3V, e 2.5V.
- Flexibilidade de programação em circuito “In – system” (ISP) através da interface Joint Test Action Group (JTAG) embutida, padrão IEEE 1149.1 – 1990, incluindo circuito “boundary – scan test” (BST).

Cabe destacar que o fato de podermos trabalhar com os dispositivos dessa família em ambiente industrial, que na maioria das vezes é um ambiente extremamente agressivo, valida seu uso em automação da manufatura.

### **4.2.1 Características Particulares dos Dispositivos**

As diferenças entre os dispositivos desta família estão dispostas nas tabelas 4.1 e 4.2, Notaremos que a apenas a capacidade dos dispositivos são diferentes e podemos encontrar os dispositivos em vários encapsulamentos.

Característica	EPM3032A	EPM3064A	EPM3128A	EPM3256A	EPM3512A
Portas utilizáveis	600	1,250	2,500	5,000	10,000
Macrocells	32	64	128	256	512
Disposição dos Blocos Lógicos	2	4	8	16	32
Quantidade Pinos de I/O	34	66 / 34	98	161	208

Tabela 4.1 – Características dos dispositivos

Dispositivo	44 – pinos PLCC	44 – pinos TQFP	100 – pinos TQFP	144 – pinos TQFP	208 – pinos PQFP
EPM3032A	34	34			
EPM3064A	34	34	60		
EPM3128A			80	96	
EPM3256A				116	158
EPM3512A					172

Tabela 4.2 – Encapsulamento dos dispositivos

#### 4.2.2 Características Físicas dos Dispositivos

O encapsulamento determina a característica física do dispositivo que para automação industrial é uma informação importante, pois o número de pinos do dispositivo será o numero de entradas e saídas que o processo a ser automatizado poderá ter. A figura 4.0 mostra os tipos de encapsulamentos dessa família.

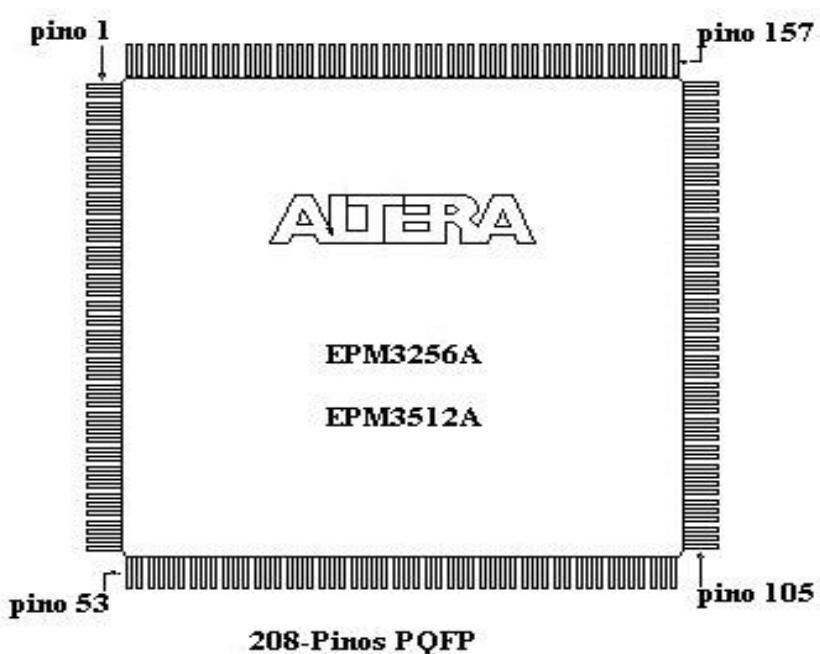
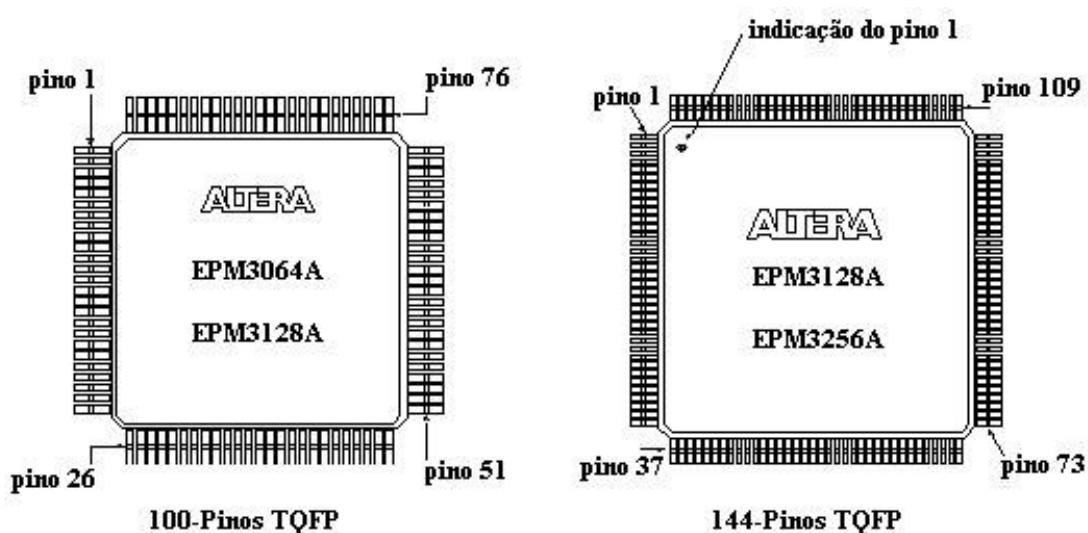
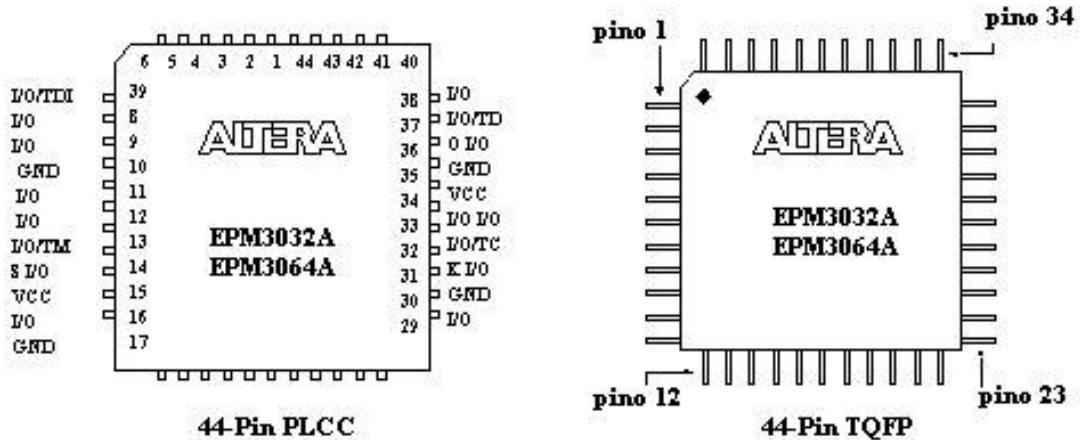


figura 4.0 – tipos de encapsulamento

### 4.3 Arquitetura MAX3000A

A arquitetura MAX 3000A é composta dos seguintes elementos:

- Blocos de ordem de lógica (LABs)
- Macrocells (Macro Células)
- Interconexões Programáveis (PIA)
- Blocos de controle de I/O

A figura 4.1 mostra a arquitetura Max 3000A e seus elementos.

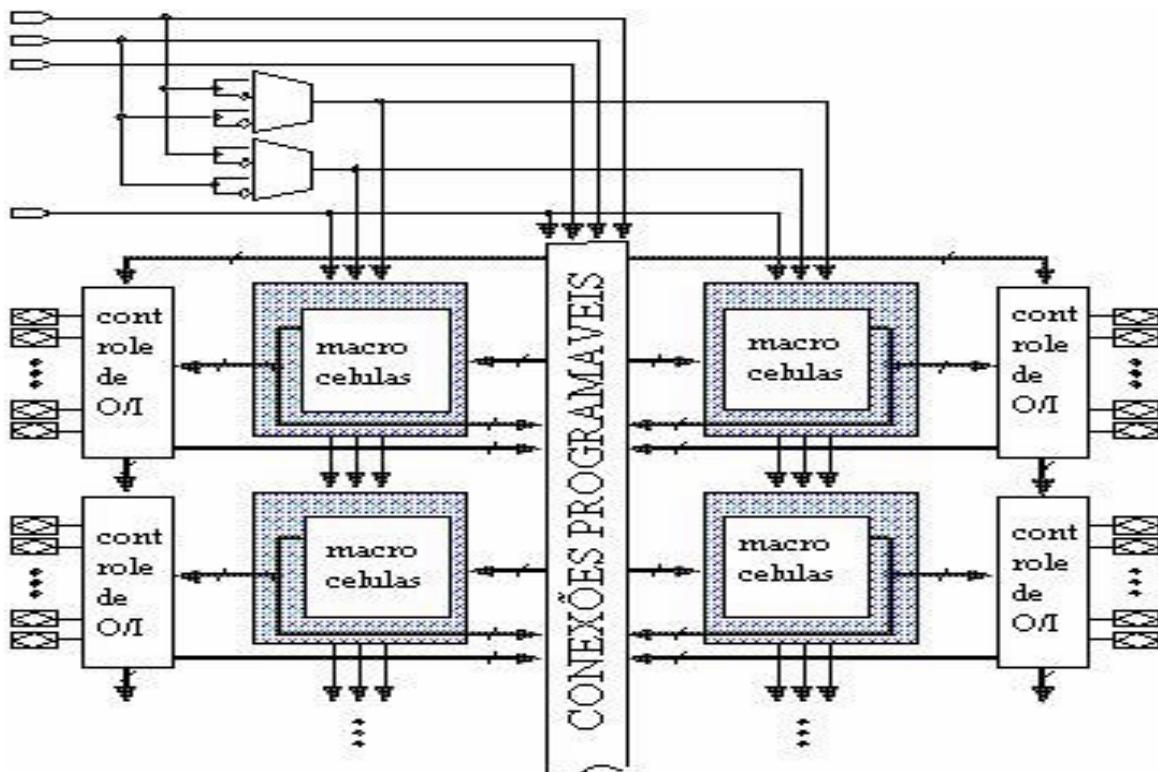


Figura 4.1 – Diagrama da arquitetura MAX3000A

Os dispositivos da arquitetura MAX3000A estão baseados na união de alto desempenho dos LABs. LABs consistem em 16 macro células em ordem, como mostrado na figura 4.1. Os múltiplos LABs são unidos pelo PIA, que se trata de um barramento global que liga: I/O, LABs e macro células.

Cada LAB é alimentado pelos seguintes sinais:

- 36 sinais do PIA que é usado para lógica geral
- Controles globais que são usados para funções de registros secundários

As macro-células podem ser configuradas individualmente, seqüencialmente ou operações de lógica combinacional. As Macro-células consistem em três blocos funcionais:

- ordem de lógica
- produto-termo e seleção de matriz
- registro programável

A lógica de Combinacional é implementada na ordem de lógica que provê cinco condições de produto por macrocell. O produto-termo e matriz de seleção aloca estas condições de produto para uso como qualquer lógica primária introduzida (para o OU e portões de XOR). O sistema desenvolvimento Altera® aperfeiçoa a distribuição de produto-termo automaticamente de acordo com as exigências de lógica designada pelo compilador.

Para funções registradas, cada uma das macro células podem ser programadas para implementar flipflop tipo D, T, JK, ou operação de SR com controle de clock programável individualmente. Os flipflop podem ser evitados para operação de lógica combinacional, o desenhista especifica o tipo de flipflop desejado; o sistema de desenvolvimento de software Altera® seleciona a operação de flipflop mais eficiente para cada função registrada e aperfeiçoa utilização dos recursos. A figura 4.2 mostra um diagrama de bloco das macro células da arquitetura MAX3000A.

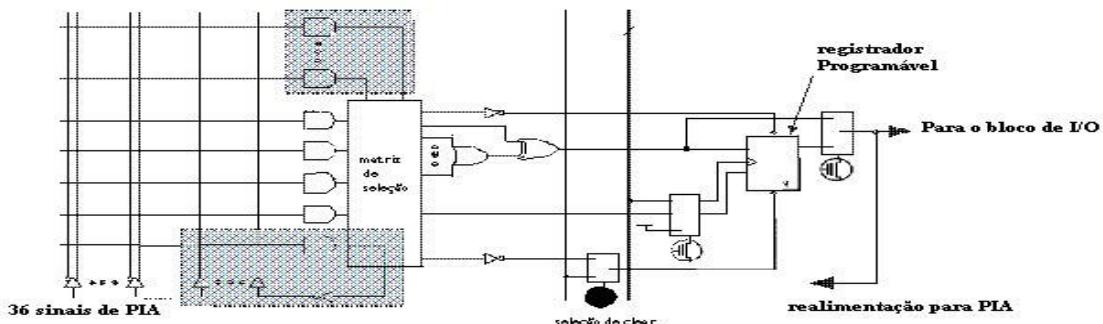


figura 4.2 – Diagrama de bloco das macro células

O PIA (Programmable Interconnect Array) é um caminho programável que conecta qualquer fonte de sinal para qualquer destino no dispositivo. Todos os dispositivos da família MAX 3000A utilizam esse barramento para interligar: I/O, LAB e macro células, que alimentam o PIA e disponibilizam os sinais ao longo do dispositivo inteiro. Cabe destacar que serão habilitados apenas os sinais requeridos por cada LAB. Figure 4.3 mostra como os sinais de PIA são introduzidos no LAB. Uma célula de EEPROM controla os sinais introduzidos através de uma porta lógica “E” que seleciona um sinal de PIA para introduzir no LAB.

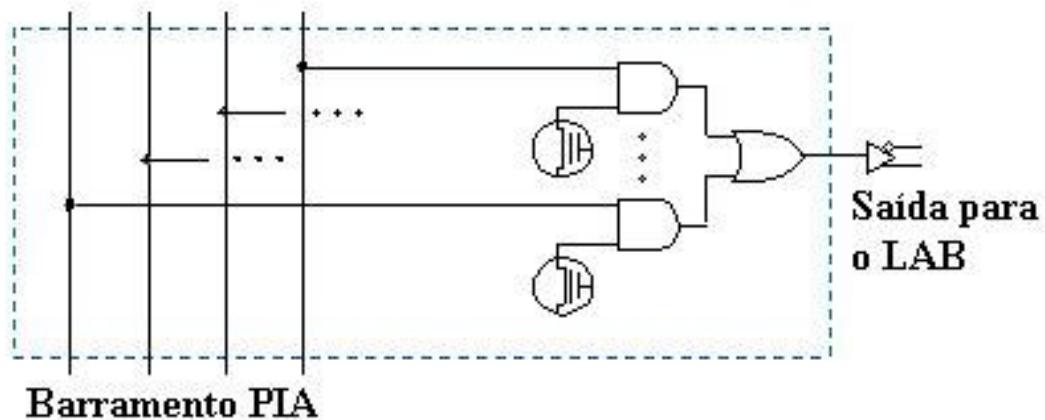


Figura 4.3 – Seleção de sinal no barramento PIA

Os blocos de controle de I/O permitem que cada pino seja configurado individualmente para entrada, saída, ou operação de bidirecional. Todos os pinos de I/O têm um tri-state que é individualmente controlado por uma saída global que habilita sinais diretamente, ao terra ou VCC. Figure 4.4 mostra o diagrama de um bloco de controle de I/O dos dispositivos da família MAX 3000A.

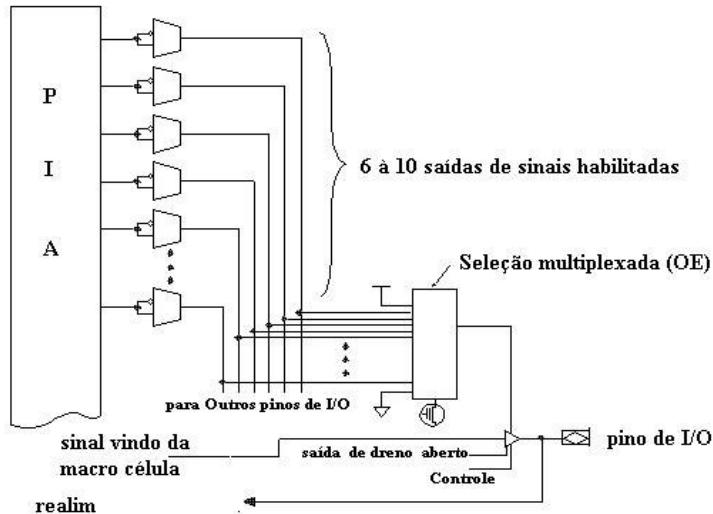


Figura 4.4 – Bloco de controle de I/O

Quando o controle é conectado ao fio terra, a saída tri-state é de alta impedância, e os pinos I/O são habilitados como entradas, e quando o controle é conectado a VCC os pinos são configurados como saídas.

A arquitetura MAX 3000A provê dupla realimentação, na qual a realimentação das macro-células e a realimentações dos pinos são independentes.

#### 4.4 Considerações Finais

Se pensarmos nesses dispositivos apenas considerando a tecnologia CMOS, seria como se ao entrar um sinal elétrico nesses dispositivos milhões de chaves abrissem e fechassem formando o arranjo lógico. Diferente dos micros controladores, que ao receberem um sinal elétrico, processam esse sinal respeitando um algoritmo e depois respondem com um sinal de saída, os CPLDs organizam arranjos lógicos entre as macro células, dispensando qualquer tipo de processamento do sinal de entrada, ou seja, um sinal na entrada gera um sinal na saída.

## **5 Ferramenta de desenvolvimento – Quartus II®**

### **5.1 Considerações iniciais**

O software de desenvolvimento Quartus II permite aos projetistas, o processamento de milhões de portas lógicas, com vantagens nunca vistas em ferramentas de desenvolvimentos de PLDs. Esse software suporta soluções em nível de sistema com editoração de blocos, trabalho em grupo e suporte para megafunções. Além disso, um sistema de análise lógica, embutido, que permite aos técnicos verificar a funcionalidade e a temporização do chip, observando sinais e valores internos gerenciados pelo clock interno.

Para que se torne compreensível o modo de trabalho dessa ferramenta será inicialmente desenvolvido e compilado, simulando um flipflop que é um dos componentes básicos da eletrônica digital. Em seguida mostraremos os circuitos lógicos e resultados do dispositivo usado para automação da planta didática demonstrativa.

### **5.2 Configurando o dispositivo**

Para iniciarmos um trabalho nessa ferramenta devemos escolher no menu FILE a opção NEW PROJECT WIZARD, como mostrado na figura 5.1, esta opção abre na tela uma janela que é mostrada na figura 5.2 onde o nome do projeto será escolhido no primeiro passo, se existir algum arquivo que possa ser incluído o programador terá essa opção no segundo passo.

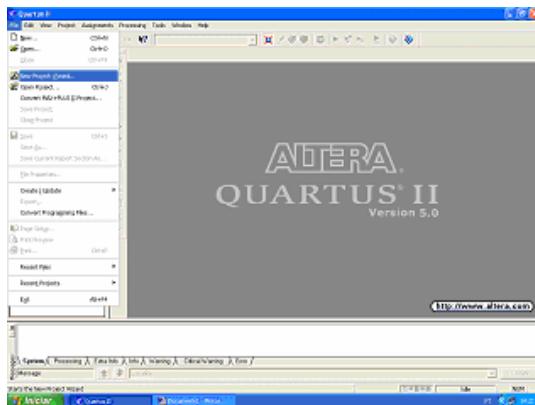


Figura 5.1 – iniciando um projeto

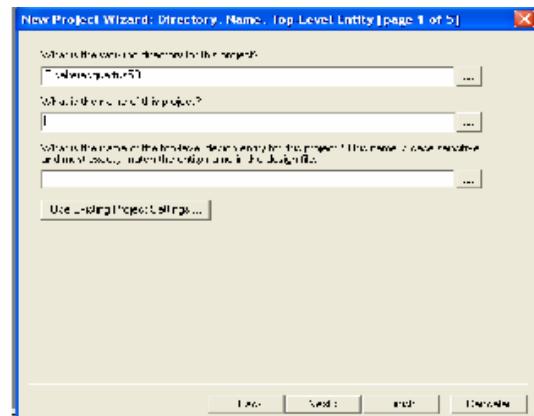


Figura 5.2 – nomeando um projeto

Após o segundo passo do projeto o dispositivo usado será configurado no terceiro passo, para que o compilador tenha noção exata da capacidade lógica e o número de pinos que o programador poderá dispor para suas implementações. A figura 5.3 mostra a tela exibida pela ferramenta e as configurações que iremos usar.

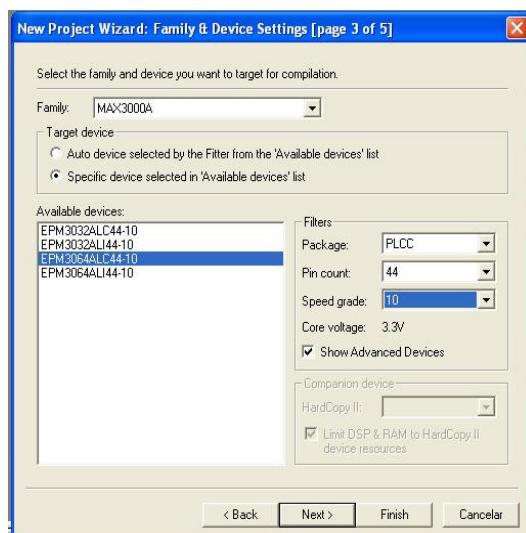


Figura 5.3 – configuração do dispositivo

O quarto passo da a possibilidade de colocação de algumas outras ferramentas EDA as quais não usaremos, por esse motivo nenhuma modificação será feita, o programador nesse passo apenas deve clicar em next para seguir rumo ao quinto passo.

No quinto e último passo apenas resta ao programador realizar uma conferência dos dados inseridos no projeto, e se tudo estiver correto clicar no botão finish. A figura 5.4 mostra o quinto passo e os detalhes a serem conferidos pelo programador.

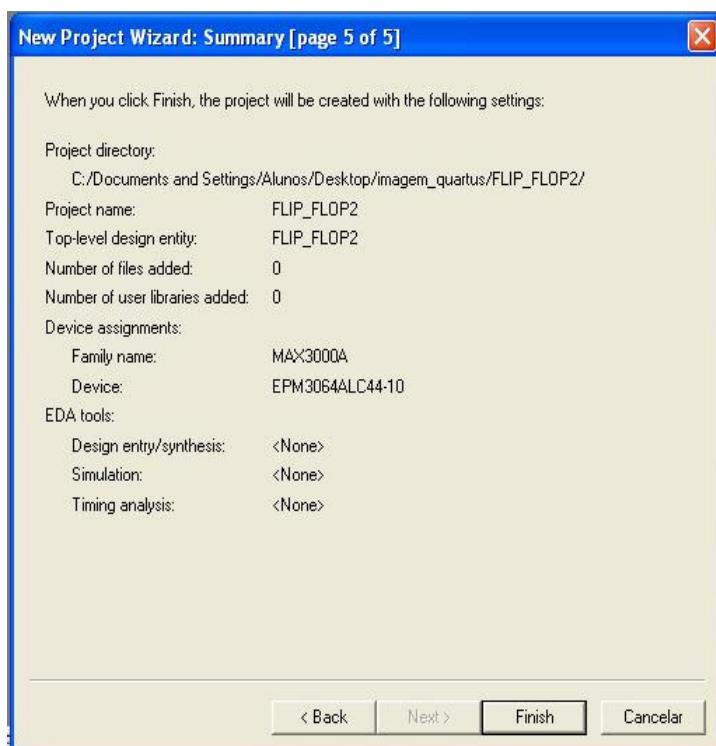


Figura 5.4 – Resultado da configuração do dispositivo

De forma muito simples e prática configuramos o dispositivo a ser implementado. Podemos observar na figura 5.3 a possibilidade de escolhermos outros dispositivos para implementação, o que não será feito, pois o dispositivo que utilizaremos no controlador com CPLD é o que acabou de ser configurado. Seguindo esses passos o programador não sentirá nenhuma dificuldade de configurar o PLD que estiver trabalhando, na próxima etapa os circuitos lógicos já poderão ser configurados em arquivos que depois de salvos obrigatoriamente deverão ser incluídos no projeto.

### 5.3 Implementação dos Circuitos Lógicos

Nessa parte do trabalho o texto exigirá do leitor um pouco de conhecimento das técnicas digitais, porém tomaremos o cuidado de não deixar o texto incompreensível a leitores que não sejam da área. Para implementarmos os circuitos lógicos no dispositivo já configurado é necessário que o programador escolha no menu FILE a opção NEW onde deverá ser escolhido o design de programação a ser

feita no dispositivo, conforme mostra a figura 5.4.1 Este arquivo depois de salvo deve ser adicionado ao projeto escolhendo no menu PROJECT a opção ADD CURRENT FILE TO PROJECT.

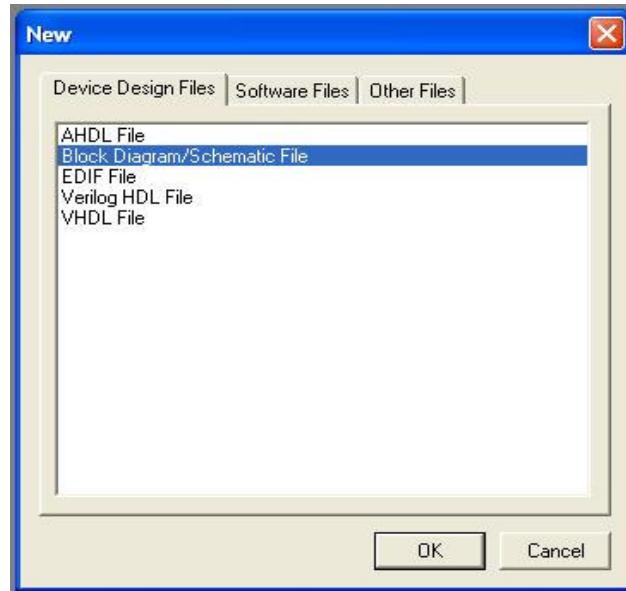


Figura 5.4.1 – Escolha do design

Para implementar o dispositivo CPLD de controle da planta didática utilizaremos o design de diagrama de blocos que mostra umas interfaces simples e amigáveis, podendo ser programada de três formas, em formato de porta lógica primitiva, diagramas de blocos lógicos ou seguindo a numeração da família de ASCI 7400, as quais serão vistas com detalhes nos próximos tópicos.

Após a escolha do design o programador deve clicar no símbolo de uma porta lógica “E” na barra de ferramenta do canto esquerdo da tela, a janela SYMBOL se abrirá e nessa janela os três tipos de simbologia serão encontrados. A figura 5.5 mostra o botão a ser clicado e a janela com os respectivos tipos de programação

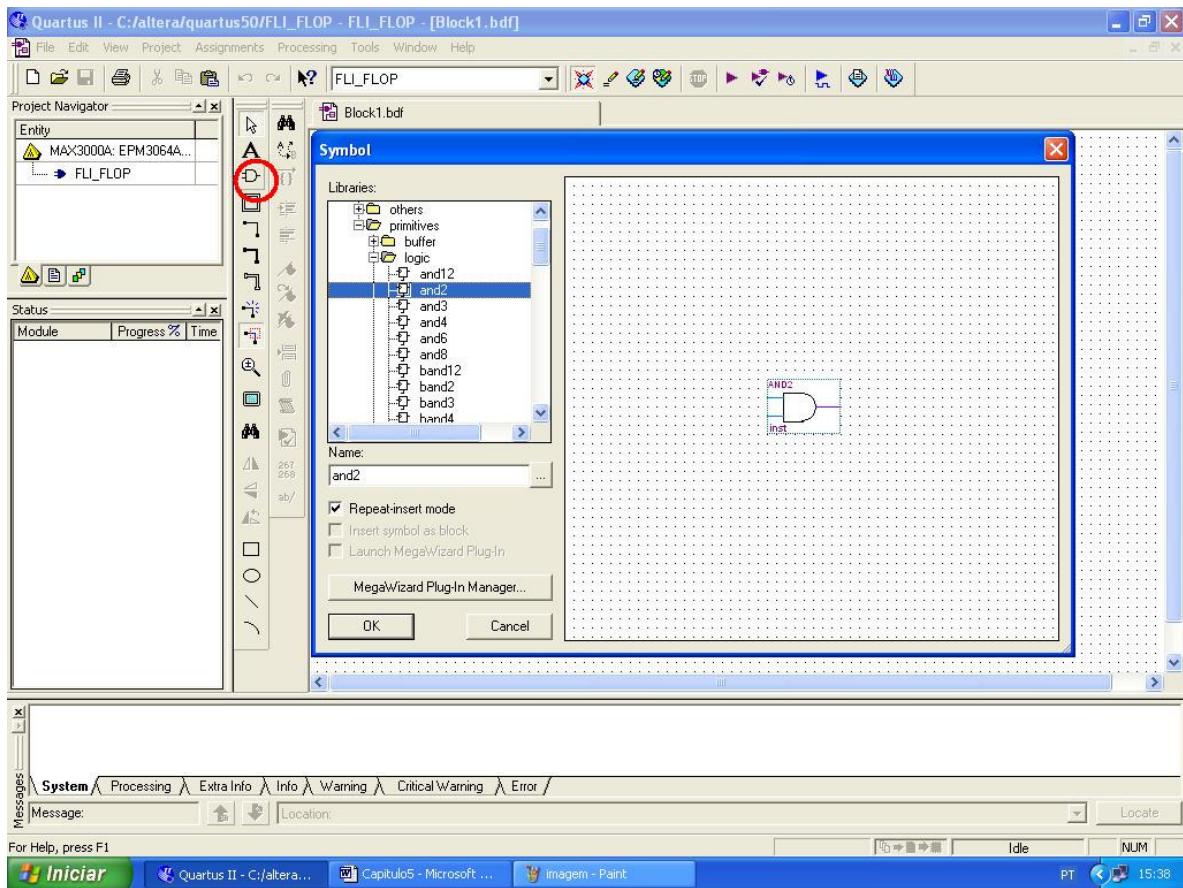


Figura 5.5 Localizando os Dispositivos

Observando a figura 5.5 notaremos dentro da janela de SYMBOL as pastas onde o programador encontrara todos os dispositivos disponíveis para que sejam feitas suas implementações. Caso o programador sinta a necessidade de utilizar um componente que não esteja disponível, ou deseje adaptar um componente já existente, ele poderá criar seu próprio dispositivo lógico através de uma ferramenta de criação de dispositivos.

### 5.3.1 Implementação de Circuitos com Portas Lógicas

Uma das formas de programação dos dispositivos é feita através das portas lógicas primitivas (“OU”; “E”; “NÃO” e derivados), que são o princípio da lógica digital.

Conforme foi citado anteriormente montaremos o flipflop, que nesse tópico será implementado apenas com o uso das portas lógicas como mostra a figura 5.6.

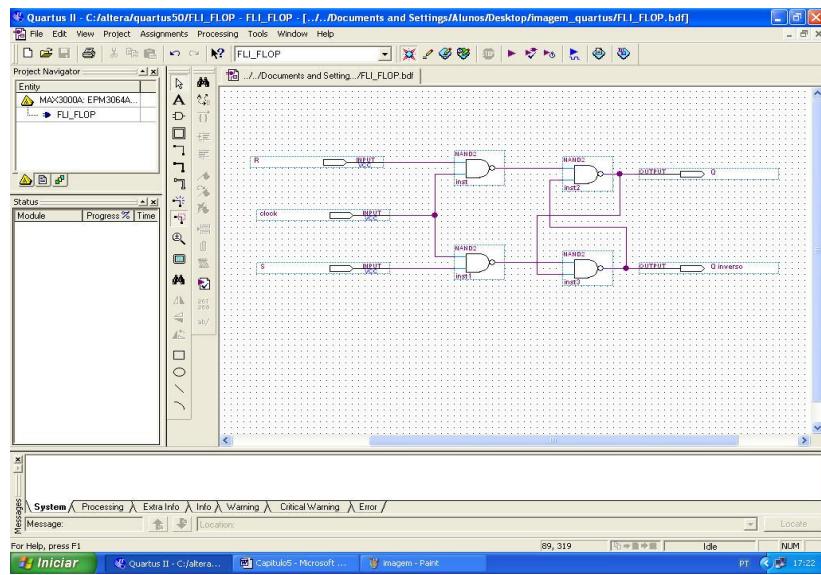


Figura – 5.6 – Implementação com porta lógica

### 5.3.2 Implementação de Circuitos com Blocos Lógicos

Outra forma de implementação é feita através de blocos lógicos que possuem na configuração interna uma combinação de portas lógicas, compactando os circuitos, diminuindo a área de trabalho e economizando macro células. A figura 5.7 mostra o mesmo flipflop, agora sendo representado por um bloco.

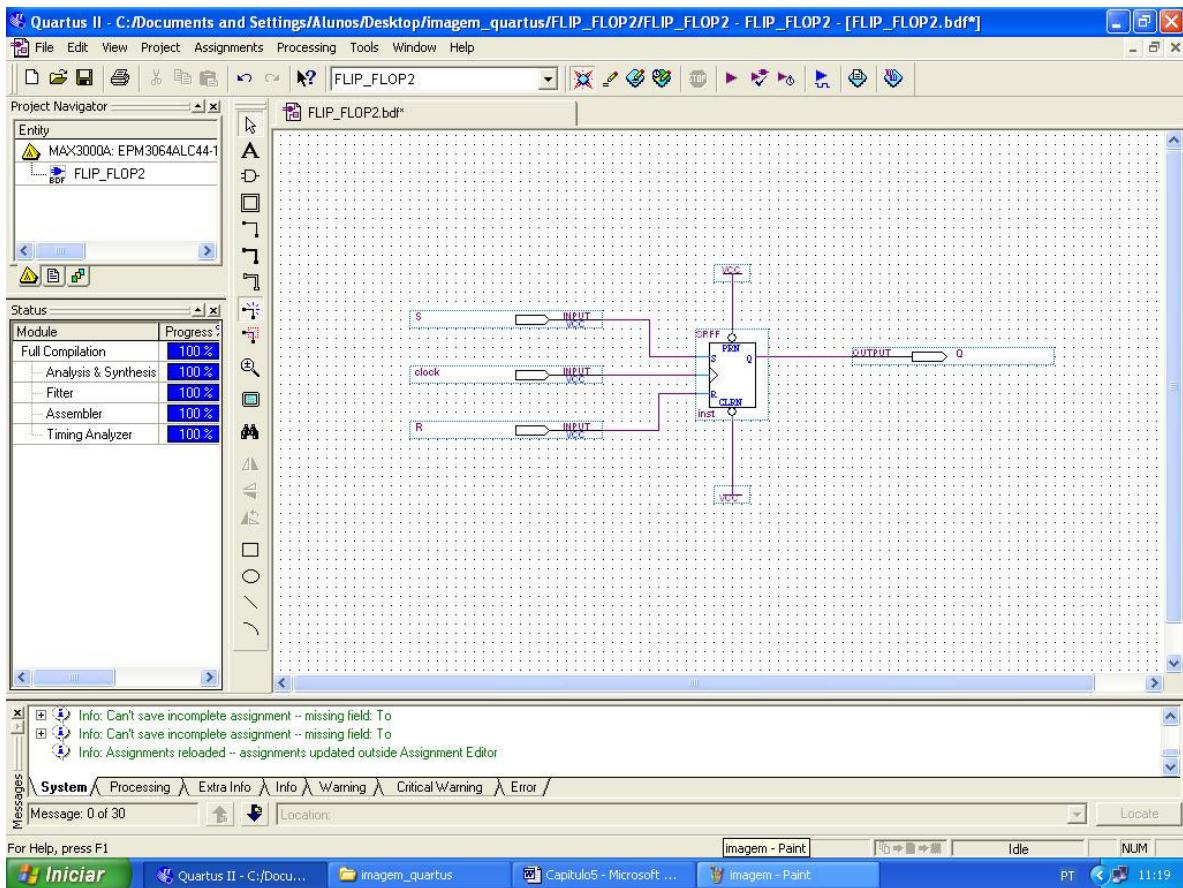


Figura – 5.7 Implementação com Blocos lógicos

### 5.3.3 Implementação de Circuitos com ASCIs

Esse tipo de implementação permite ao programador inserir nos dispositivos PLDs circuitos com lógica fixa previamente desenvolvida e projetar utilizando o raciocínio de implementação da família ASCI 7400. A figura 5.8 mostra um flipflop 74107 que terá praticamente a mesma função dos circuitos antes apresentados.

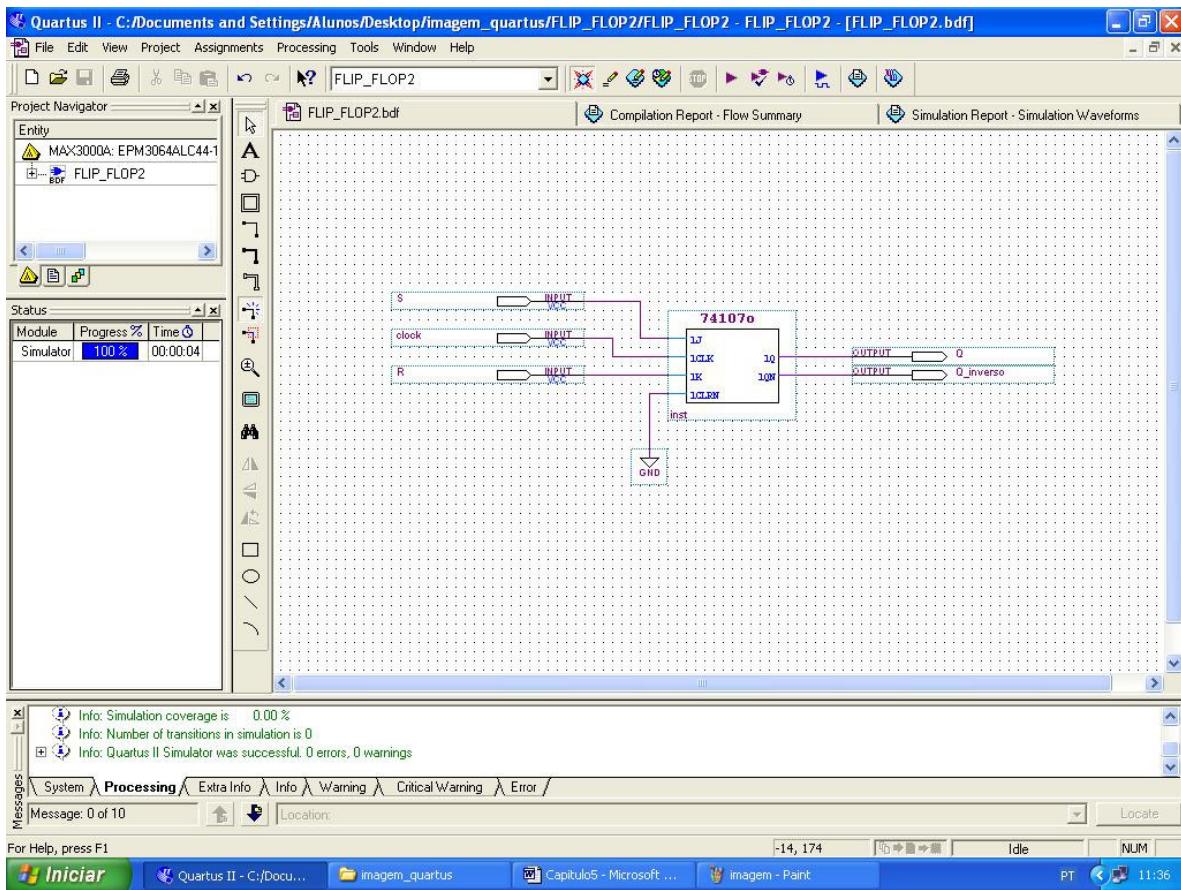


Figura – 5.8 Implementação com ASCIs

Destacamos que os tipos de programação também podem ser misturados, ou seja, podemos implementar circuitos com ASCIs e conectar nesses dispositivos portas lógicas ou blocos lógicos. Não importando a ordem dos fatores nem a seqüência que eles apresentem.

## 5.4 Compilação dos Circuitos Lógicos

Para compilarmos um circuito lógico já salvo e adicionado ao projeto, é preciso apenas que o programador clique no botão destacado por um círculo e mostrado na figura 5.9, que ainda mostra os resultados encontrados após a compilação.

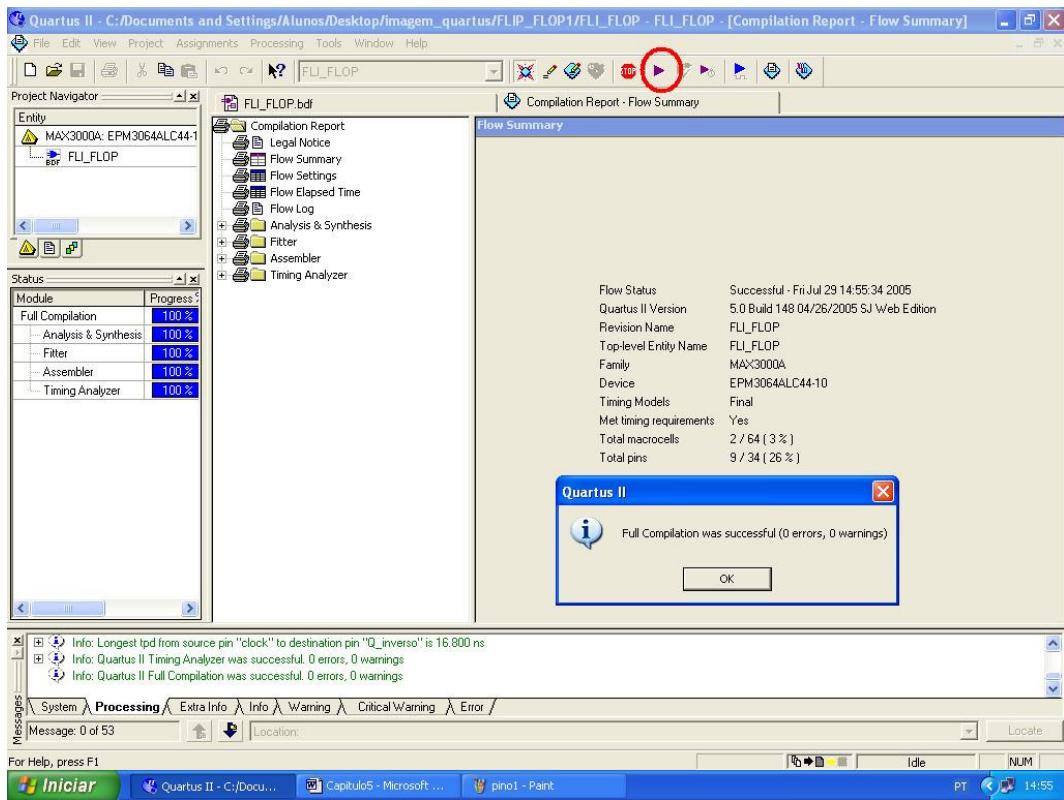


Figura – 5.9 Resultados da Compilação

Podemos notar no canto esquerdo a figura acima os módulos e o progresso da compilação e, no rodapé desta imagem encontraremos as mensagens de erro ou de sucesso dependendo do progresso do circuito lógico compilado.

## 5.5 Simulação dos Circuitos Lógicos

Para simularmos o circuito lógico que já foi compilado será necessário que todos os pinos de entrada e de saída sejam nomeados e configurados. Ao clicar duas vezes sobre o símbolo do pino a janela mostrada na figura 5.9 traz as propriedades do pino e abre a possibilidade de mudanças de seu nome, a mudança do nome do pino não se faz obrigatória, mas, no entanto ao mudar o seu nome o programador terá maiores facilidades no decorrer do projeto.

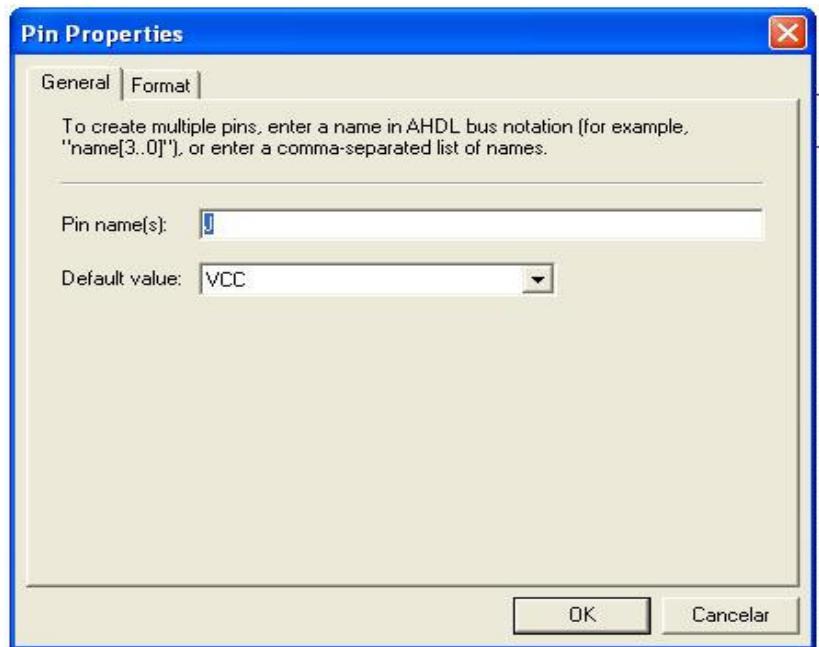


Figura – 5.9.1 Propriedade do Pino

Em seguida deve-se dar aos pinos a sua devida numeração, para isso o programador deve escolher no menu ASSIGNMENTS a opção PINS, que mudará a área de trabalho para a janela mostrada na figura 5.8.1, onde os pinos serão numerados de acordo com a escolha do programador.

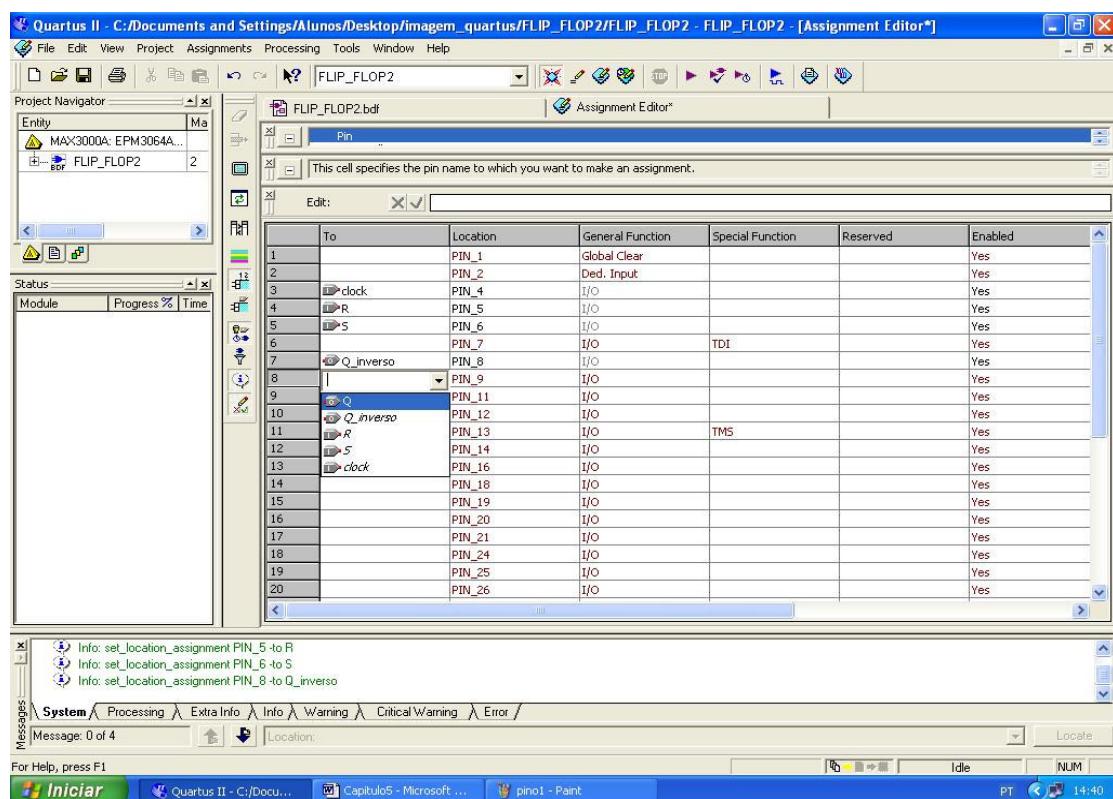


Figura – 5.9.2 Numerando Pinos

Uma vez numerados os pinos, devemos salvar novamente o arquivo para que sejam validadas as novas configurações. Como resultado teremos na área de trabalho caixas de texto contendo o número do pino no dispositivo. A figura 5.9.3 mostra o resultado dessa configuração, e traz também em destaque os botões que devemos clicar para mudarmos a área de trabalho dos diagramas de blocos para área de compilação e configuração dos pinos.

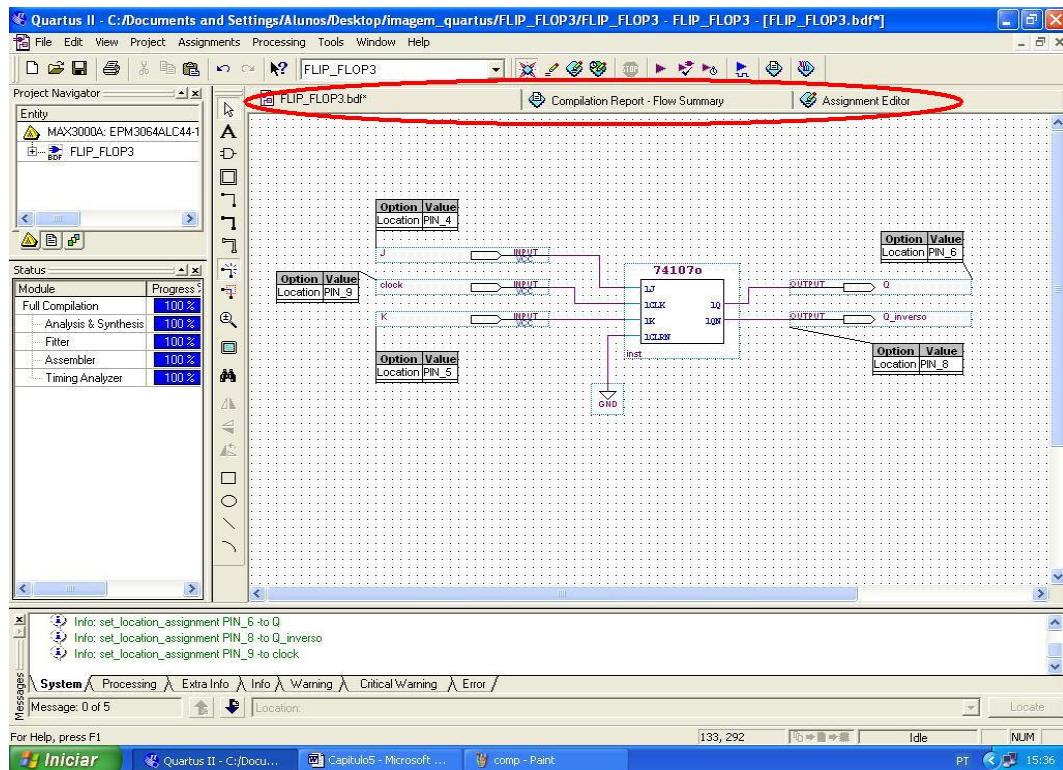


Figura – 5.9.3 Resultado da Numeração dos Pinos

O próximo procedimento para realizar a simulação exige do programador um novo arquivo de outro modelo. Para isso deve ser escolhida no menu FILE a opção NEW, o modelo de arquivo a ser escolhido está em destaque na figura 5.9.4, esse arquivo não precisa ser adicionado ao projeto apenas deve ser salvo para ter validade.

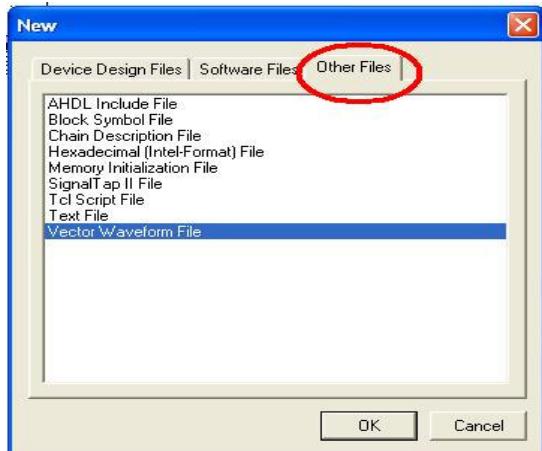


Figura – 5.9.4 Novo Arquivo para Simulação

Após a escolha do novo arquivo a área de trabalho mudará para a aparência mostrada na figura 5.9.5, podemos notar nessa figura dois pontos de destaque, o primeiro ponto indica o local onde o programador deve clicar com o botão direito e selecionar a opção INSERT NODE OR BUS, o segundo ponto de destaque mostra o nome e o tipo de arquivo utilizado nessa operação.

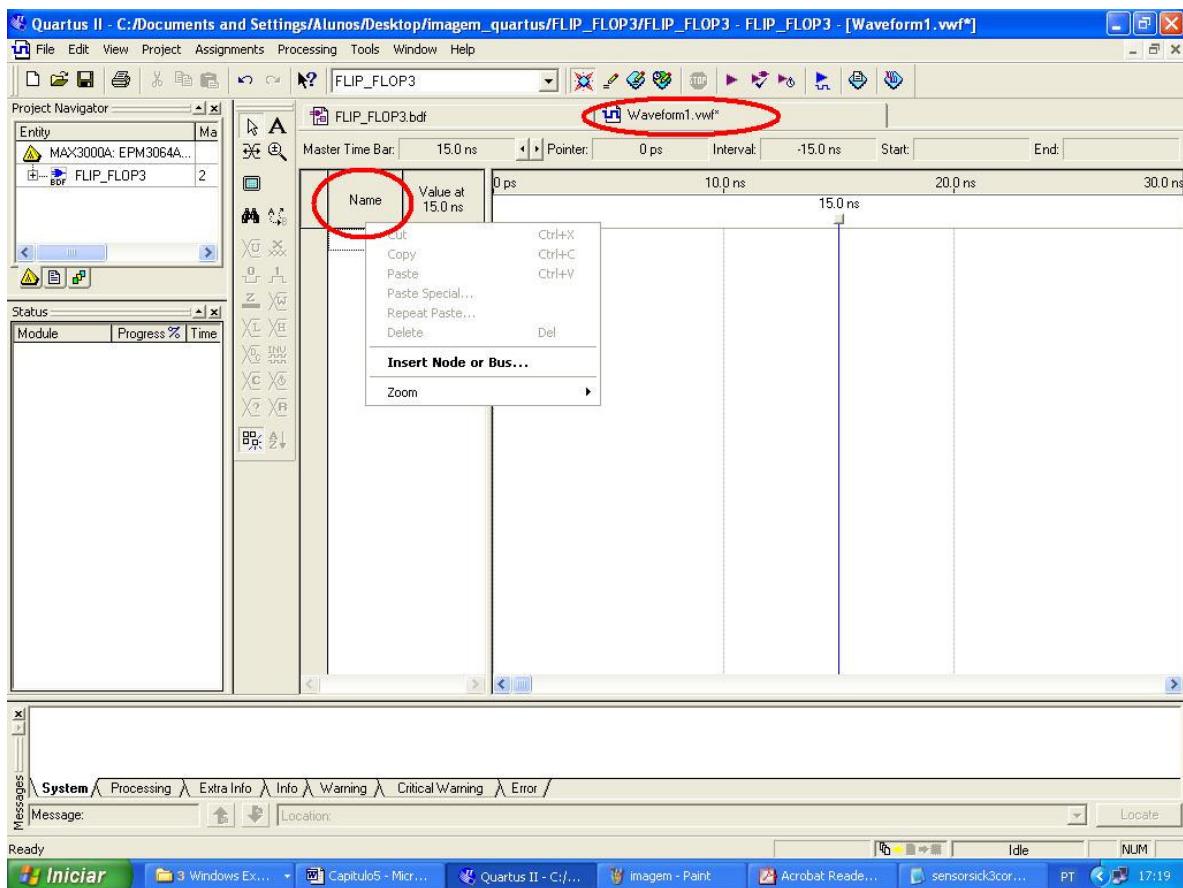


Figura – 5.9.4 Novo Arquivo para Simulação

Ao clicar na opção INSERT NODE OR BUS uma pequena janela se abrirá e o botão NODE FINDER deve ser pressionado, esse procedimento levará o programador à outra janela que tem como função escolher os pinos que participarão da simulação. A primeira atitude a ser tomada nessa etapa é pressionar o botão LIST, que se encontra no canto superior direito da janela, no canto esquerdo aparecerão os pinos listados no projeto, o programador deve usar os botões localizados no centro da janela para selecionar os pinos desejados, por fim o programador deve clicar OK nas duas janelas. Os detalhes dessa janela estão na figura 5.9.5.

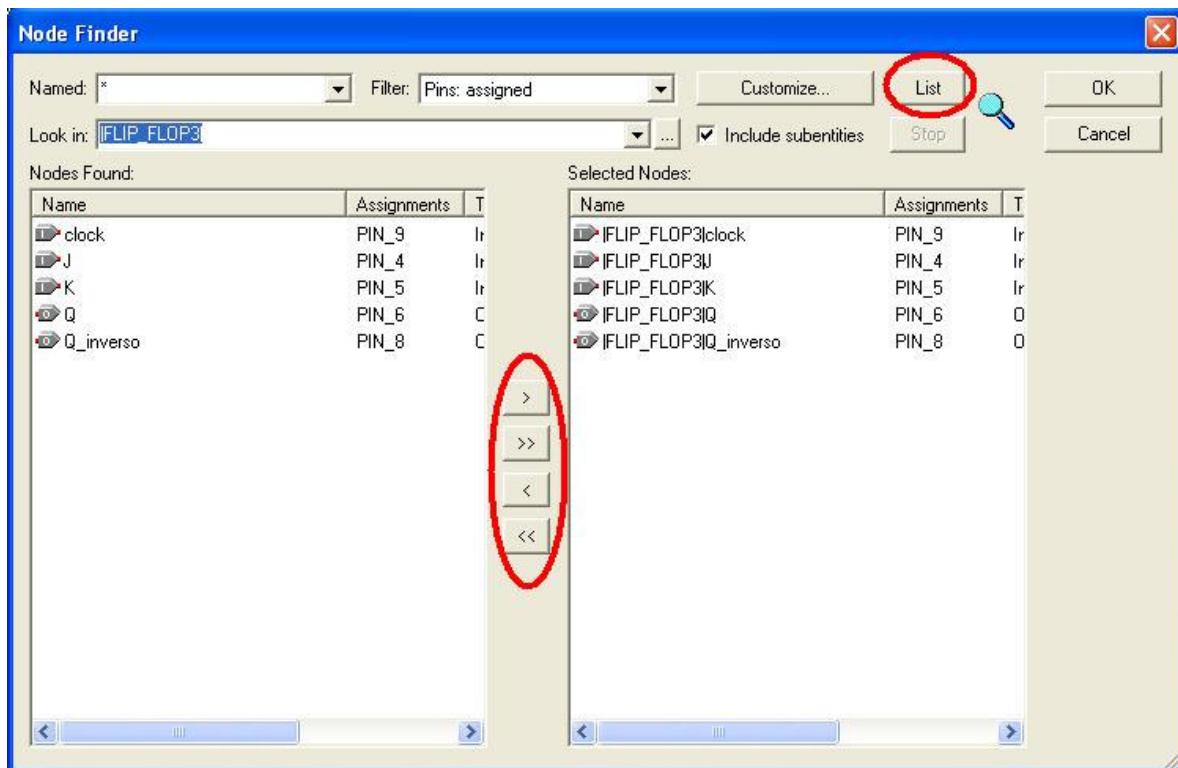


Figura – 5.9.5 Janela de Seleção dos Pinos para Simulação

Depois de listados, os pinos aparecem como mostra a figura 5.9.6, os botões que são destacados na esquerda dessa figura serão utilizados pelo programador para que se configure o sinal de entrada de acordo com o desejado, bastando apenas que sejam selecionadas as áreas de mudança de nível lógico. Após esses procedimentos basta apenas um click no botão START SIMULATION para que o

compilador inicie a simulação, note que no nosso exemplo as entradas J e K estão infinitamente em nível lógico alto e dois pulsos de clock serão dados.

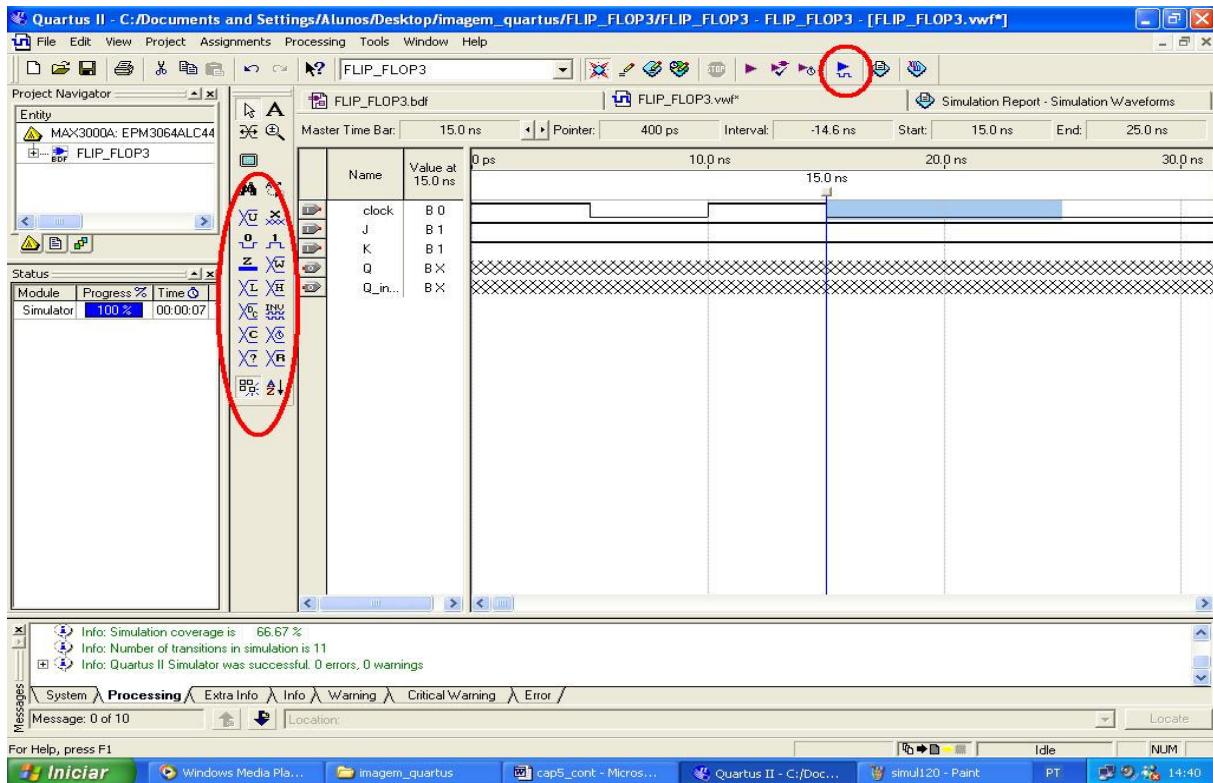


Figura - 5.9.6 configuração de entradas para simulação

Os resultados dessa simulação são mostrados na figura 5.9.8, onde podemos perceber pelo gráfico uma resposta na saída com atraso de 4.5 nano segundos em relação ao sinal de entrada. Com as entradas J e K infinitamente em nível lógico alto as saídas se alternaram entre nível lógico alto e baixo de acordo com a variação na entrada de clock.

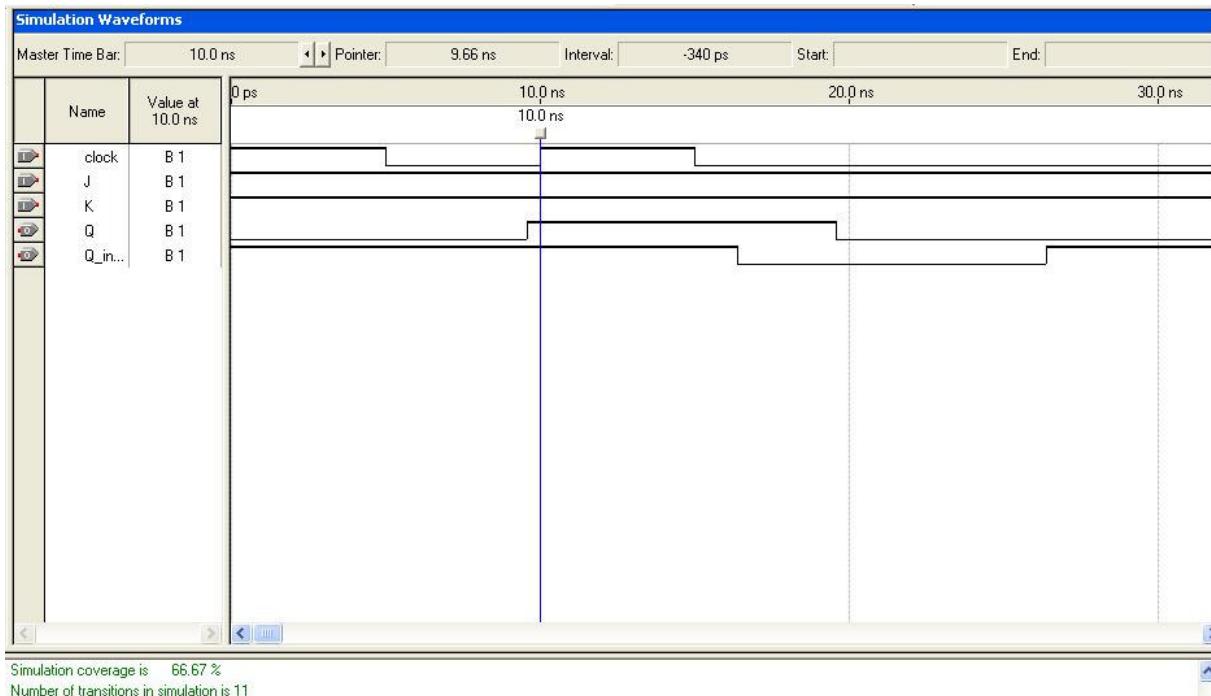


Figura - 5.9.8 Resultados da simulação

## 5.6 Resultados de compilação da planta demonstrativa

Agora que já analisamos todos os procedimentos de compilação, desenvolvimento e simulação, podemos analisar com convicção os resultados apresentados no dispositivo que será utilizado no controlador lógico programável com tecnologia CPLD.

Os diagramas de bloco com o desenvolvimento lógico, utilizam-se de um método conhecido como cadeia estacionária, muito utilizado em automação industrial. A lógica que comandará o processo da planta demonstrativa ocupará 36 macro-células, que equivalem a 56 % da capacidade do dispositivo, isso deixa margem para a melhoria do processo e o desenvolvimento de circuitos ainda mais complexos. A figura 5.9.9 mostra os resultados encontrados na compilação do dispositivo utilizado no controlador com tecnologia CPLD.

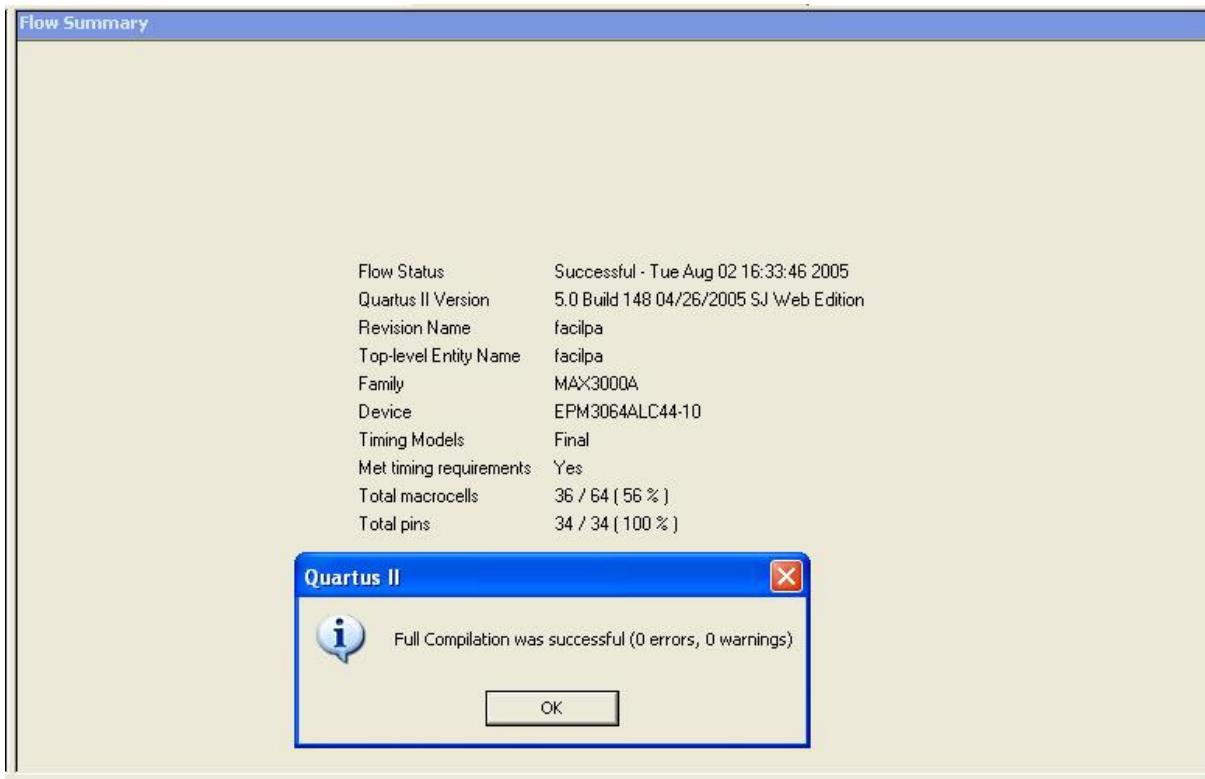


Figura - 5.9.9 Resultados da compilação

## 6. Kit–PLD Versão Altera MAX 3000A

### 6.1 O Kit–PLD e Sua Estrutura

O Kit-PLD foi concebido como uma plataforma de desenvolvimento modular para sistemas digitais baseados em dispositivos lógicos programáveis (PLDs), da família MAX3000A da Altera®. O Kit-PLD possui um soquete PLCC-44, compatível com os modelos EPM3032ALC44-x e EPM3064ALC44-x, possui também um circuito de programação embutido, compatível com o Byte Blaster MV da Altera®.

A estrutura do Kit-PLD é composta por quatro módulos, sendo um módulo do PLD onde o dispositivo é programado, e outros três de aplicações, onde são realizados testes e simulações. Esses módulos estão conectados a duas unidades da inovadora Plataforma CNZ, tornando o produto expansível e modular. A figura 6.1 mostra a aparência do kit e detalha os módulos que o compõem.

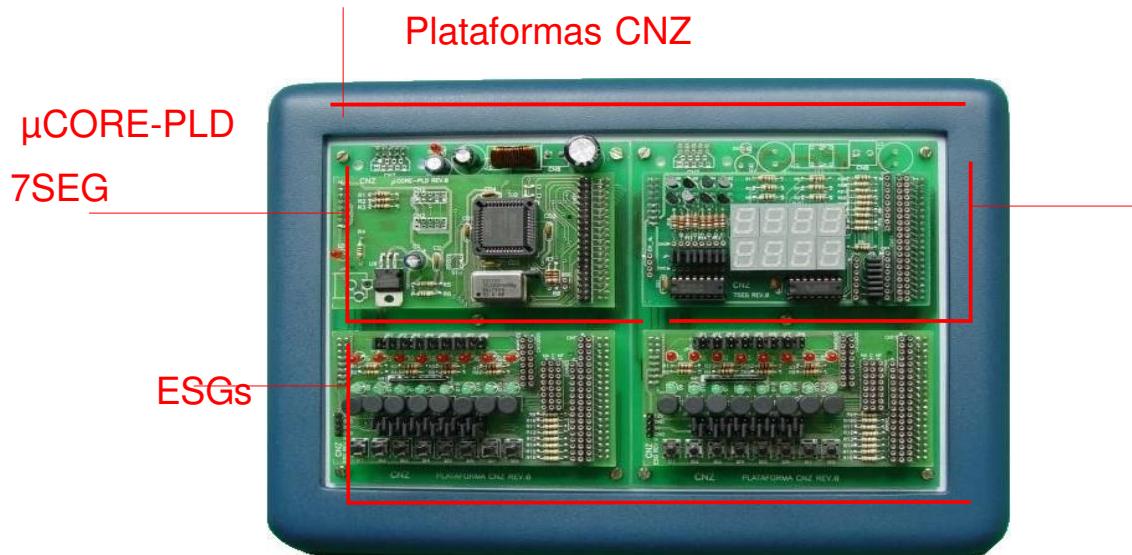


Figura 6.1 Kit–PLD Versão Altera MAX 3000A.

## 6.2 Plataforma CNZ

A Plataforma CNZ é a base para a montagem do Kit–PLD, introduzindo o conceito de modularidade, onde uma única plataforma suporta módulos de diversos PLDs, microcontroladores e até DSPs; além de diversos módulos de aplicação, desde módulos com botões e leds até módulos de acionamento de potência. A Plataforma CNZ possui as seguintes características:

- Fontes de alimentação reguladas de +5 VCC, +12 VCC e +13 VCC;
- Led indicativo de funcionamento;
- Conexão de interface;
- Barramento de conexão do PLD

O Kit–PLD emprega duas unidades da plataforma CNZ interligadas e instaladas sobre uma base, onde apenas uma das plataformas é montada com a fonte. A base além de conferir ótimo acabamento e aparências, ainda abriga:

- Alimentação universal automática (110/220V sem chave de seleção);
- Chave liga–desliga;
- Fusível de proteção de 0,5 A;
- Conexão do barramento do **μCORE**;
- Conector DB-25 para interface com o PC

- Circuito para programação compatível com o Byte Blaster MV

O Kit-PLD possui um circuito de gravação embutido, compatível com o Byte Blaster MV da Altera®. Esse circuito é conectado a um microcomputador através da interface paralela e compatibiliza os níveis desta com os da interface JTAG. A conexão com o PC é feita por um cabo DB-25 pino a pino, enquanto a conexão do circuito de programação ao µCORE-PLD é feita através da plataforma CNZ.

### 6.3 Módulo µCORE-PLD

O µCORE-PLD possui um PLD Altera da família MAX3000A com encapsulamento PLCC-44, é utilizado um soquete para possibilitar a troca do componente. Os terminais do PLD estão disponíveis para conexão e monitoração no conector CNT1, que é repetido em todos os outros módulos de aplicação.

O módulo possui ainda um led indicador de ligado, um regulador de +3,3V e um oscilador a cristal de 20MHz que pode ser substituído por outro de tamanho compatível com DIP-14, ou DIP-8, e alimentação de +5V. O oscilador está ligado ao pino 43 do PLD, Global Clock 1 e possui um jumper para desconectá-lo, além de um conector para o monitoramento ou outra utilização do sinal. A figura 6.2 mostra com detalhe este módulo do kit.



Figura 6.2 – Módulo µCORE-PLD

### 6.4 Módulo ESG

O módulo de entradas e saídas (ESG) é composto de vários circuitos independentes entre si exceto alimentação, e estrutura-se basicamente de:

- 8 leds com jumpers de seleção para acendimento em nível lógico baixo ou alto;
- 8 mini-chaves de retenção com led indicativo de pressionamento e saídas NA, comum, NF e polarizada (nível alto ou baixo);
- 8 botões (chaves tácteis) com seleção para operação em pull-up ou pull-down.

O módulo ESG dispõe de 8 leds (D1-D8) dispostos em uma barra horizontal. Os leds podem ser acessados tanto com nível lógico baixo como com nível lógico alto. Essa configuração é feita através dos jumpers JP1 a JP16. Cada par de jumpers corresponde a um led. A configuração dos jumpers e o esquema elétrico correspondente podem na figura 6.3 e os conectores dos anodos e dos catodos podem ser vistos na figura 6.5.

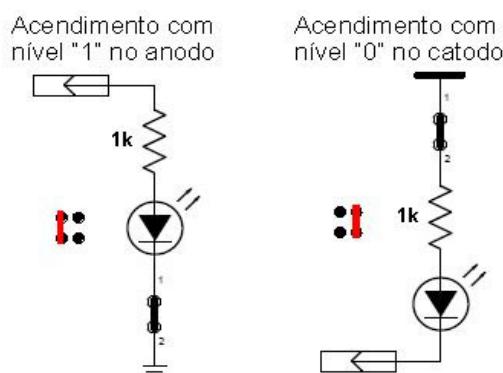


Figura 6.3 - esquema elétrico dos LED

As 8 chaves de 2 pólos com retenção (S1-S8) estão ligadas em uma dupla configuração que pode ser vista na figura 6.4. A primeira configuração com os terminais NA, comum e NF disponíveis diretamente, e a segunda, já polarizada, apresentando nível alto quando pressionada. Cada chave possui um led indicador de pressionamento na cor verde. Na figura 6.5, pode-se ver a localização das chaves e dos respectivos conectores.

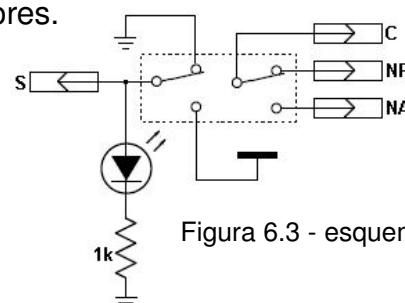


Figura 6.3 - esquema elétrico das chaves

O módulo ESG dispõe de 8 botões (B1-B8), do tipo chave táctil, já polarizados nas configurações “pull-up” e “pull-down”. A seleção das configurações para cada botão é feita através dos jumpers JP17 a JP31, conforme indicado na figura 6.4. A disposição dos botões pode ser vista na figura 6.5.

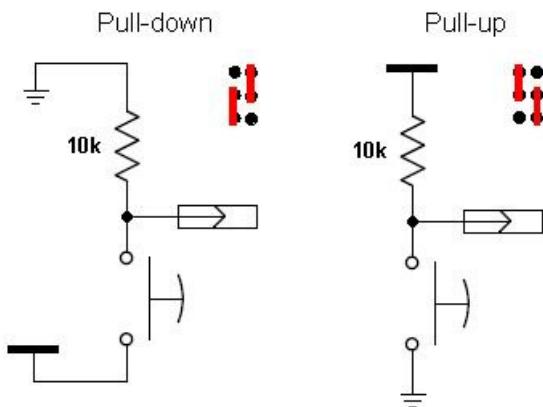


Figura 6.4 - esquema elétrico dos botões

A figura 6.5 ilustra todas as conexões do módulo **ESG**.



Figura 6.5 - conexões do módulo ESG.

## 6.5 Módulo 7 SEG

O módulo de aplicação 7 SEG do Kit-PLD é constituído por oito displays de sete segmentos, mais um CI 74138 para demultiplexação e, um CI 7447 para conversão BCD — 7 segmentos. A figura 6.6 mostra a disposição de cada um desses itens e a pinagem dos jumpers e conectores.

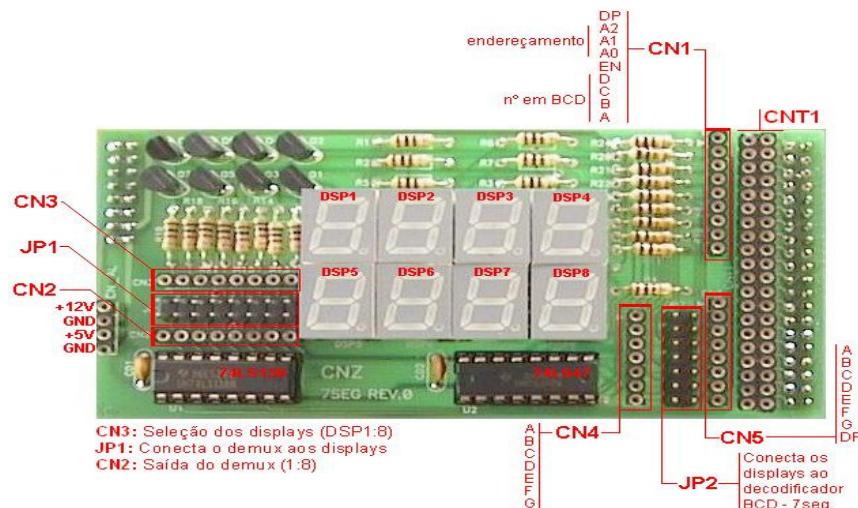


Figura 6.6 – Módulo 7 SEG

São 8 displays de 7 segmentos, tipo anodo comum, dispostos em duas fileiras. A seleção de um determinado display é feita com nível lógico baixo no pino correspondente. Para se acender um determinado segmento do display selecionado, deve-se levar o pino correspondente também ao nível lógico baixo. A figura 6.7 mostra o esquema elétrico dos displays. A localização física desses itens está na figura 6.6.

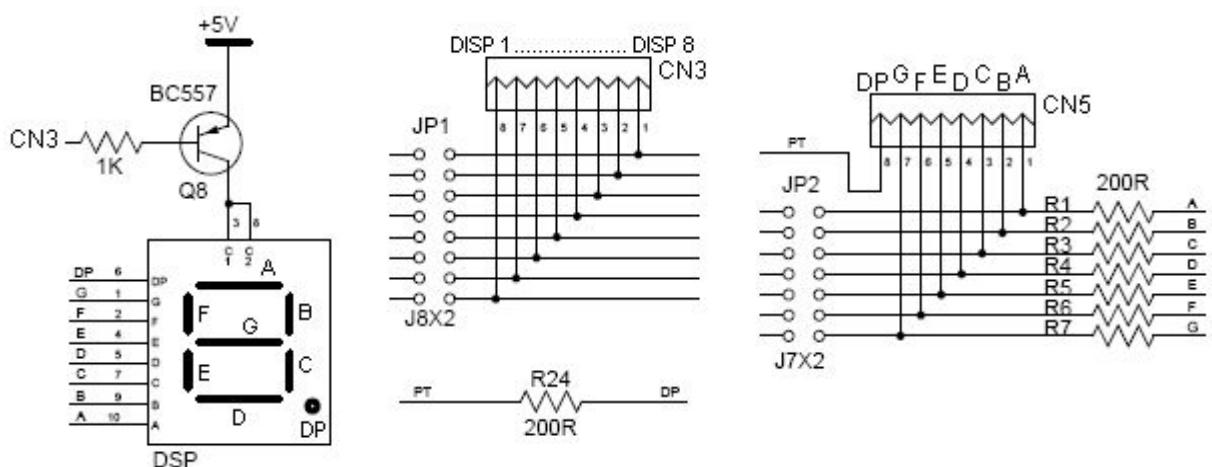


Figura 6.7 - esquema elétrico dos displays

O CI 74138 é um demultiplexador que aciona cada uma de suas oito saídas em função de uma função de uma valor binário, de três bits, presente em suas entradas. O CI pode ser utilizado para a multiplexação dos displays de sete segmentos, ou independentemente, em uma outra aplicação.

As entradas são os pinos A, B e C (000 a 111). O pino G1 funciona como

habilitação para a seleção. O acesso a esses pinos se dá pelo conector CN1. As saídas Y0 a Y7 estão disponíveis no conector CN2. Elas são ligadas aos pinos de seleção dos displays DSP1 a DSP8 através dos jumpers JP1 (quando conectados), viabilizando a multiplexação. O circuito é mostrado na figura 6.8 e a localização de cada item mencionado pode ser verificada na figura 6.6.

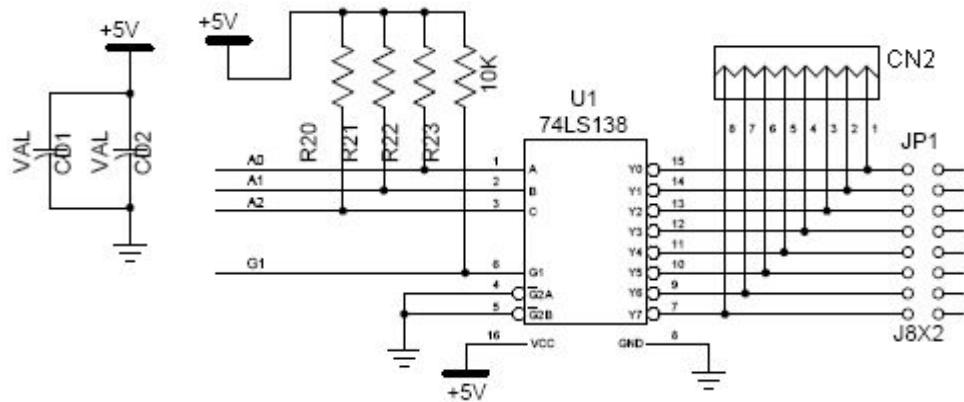


Figura 6.8 - Demultiplexador

O CI 7447 é um decodificador BCD para 7 segmentos, eliminando o trabalho de compor manualmente cada numeral no código do programa. Por trabalhar com a notação BCD, só são convertidos os algarismos de 0 a 9, sem incluir as letras A a F da notação hexadecimal. A entrada do circuito possui apenas os quatro dígitos: A, B, C e D, e está disponível em CN1. A saída disponibiliza cada um dos sete segmentos, de A a G, sem o ponto decimal, acessível através de CN4. Os jumpers em JP2, quando colocados, conectam o decodificador aos displays. O diagrama esquemático está na figura 6.9, enquanto os conectores e jumpers podem ser vistos na figura 6.6.

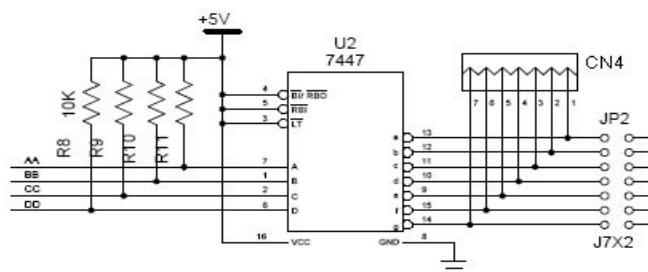


Figura 6.9 – Decodificador BCD

## 7 Demonstração do CPLD em Automação Industrial

### 7.1 Considerações iniciais

A demonstração do uso da tecnologia CPLD em automação industrial será feita através de uma planta didática que simula um processo de perfuração de placas, em seguida realiza um teste de qualidade, verificando se os furos foram feitos com perfeição e rejeitando as peças defeituosas.

O processo realizado não será o ponto mais importante do trabalho, ou seja, poderíamos estar automatizando qualquer processo industrial didático ou não. A figura 7.1 mostra uma visão geral da planta didática.

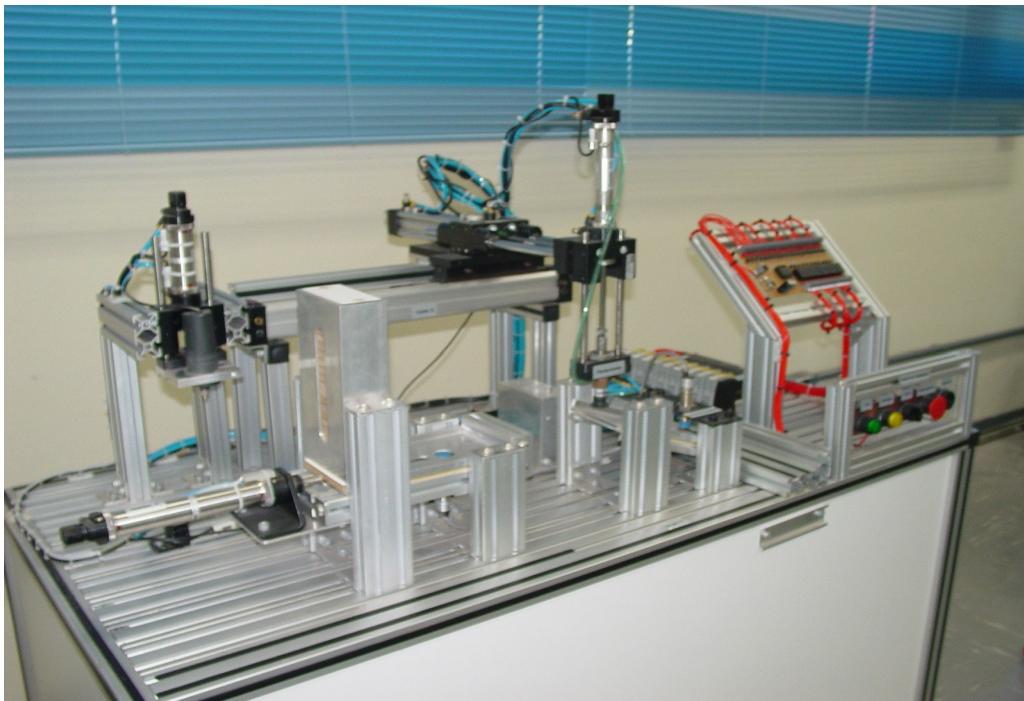
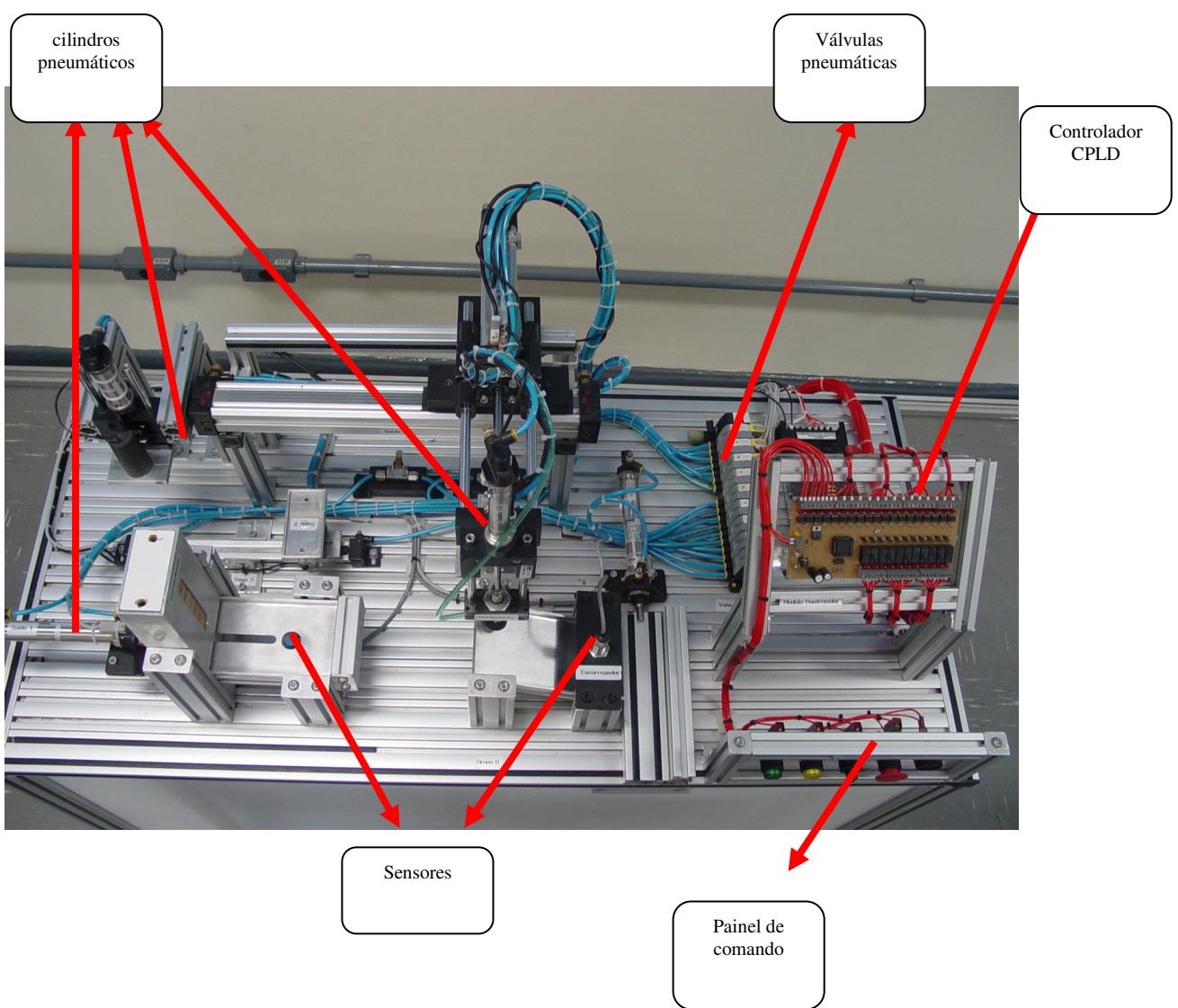


Figura 7.1 – visão geral da planta

### 7.2 Componentes da planta didática

A planta didática a ser automatizada é basicamente composta por um controlador de tecnologia CPLD, um painel de comando, válvulas eletro pneumáticas, atuadores (cilindros) e sensores. Toda essa estrutura está montada

sobre perfilados de alumínio, a figura 7.2 mostra os componentes que formam a estrutura do processo.



Figuras 7.2 – componentes da planta

### 7.2.1 Painel de Comando

Com a responsabilidade de fazer a interface do operador do processo, com o controlador de tecnologia CPLD, o painel de comando possui em sua estrutura um botão liga, que inicia o ciclo e um botão desliga que interrompe o processo no final do ciclo. O botão denominado ciclo tem como finalidade dar ao operador a opção de ter ciclos infinitos ou ciclo unitário dependendo da posição em que ele estiver.

Um botão de emergência também compõe o painel para que o ciclo seja interrompido a qualquer momento, em caso de perigo retomando o processo do ponto onde foi interrompido. O painel de comando da planta didática também possui uma sirene que é acionada sempre que o sistema detecta um erro na peça perfurada. Ao ouvir o bip da sirene o operador deve retirar a peça defeituosa e pressionar o botão liga para reiniciar o processo.

A figura 7.3 mostra o painel de comando do sistema.

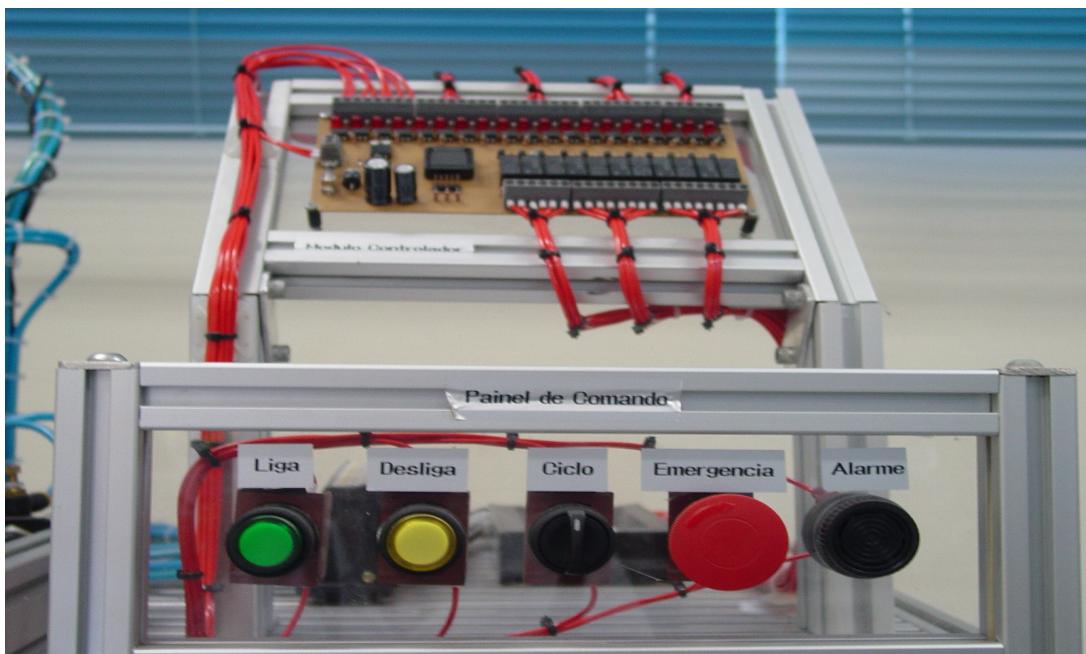


Figura 7.3 – Painel de comando

### 7.2.3 Válvulas, sensores e atuadores

As válvulas eletro pneumáticas, os sensores e os cilindros atuadores já mostrados na figura 7.2 e utilizados nessa simulação de processo industrial, são os comumente aplicados em qualquer automação industrial. Por essa razão pouco

temos a acrescentar sobre esses componentes, vale salientar a função de cada um deles nos sistema para que os leitores compreendam com mais facilidade o papel do controlador.

Os sinais de saída do controlador comandam as válvulas pneumáticas, que por sua vez avançam e recuam os cilindros atuadores responsáveis pelo transporte da peça durante o processo. Os sensores ficam monitorando o posicionamento da peça e mandam os sinais de entrada ao controlador, sensores também estão instalados nos cilindros e indicam ao controlador quando o cilindro está avançado ou recuado.

### 7.3 O Controlador com Tecnologia CPLD

O dispositivo utilizado nesse controlador será o MAX EPM3064ALC44-10 que possui no encapsulamento PLCC a quantidade de 34 pinos de I/O, a aplicabilidade desse tipo de controlador está intimamente ligada ao número de entradas e saídas que serão necessárias para automatizar o processo. Para essa planta serão necessários 30 pinos, sendo 19 entradas e 11 saídas.

O controlador com tecnologia CPLD tem como função comandar todo o processo. É ele quem recebe os sinais do painel de comando e dos sensores, e responde a esses sinais acionando as válvulas pneumáticas através do circuito de saída.

Esse tipo de controlador possui como características: O baixo custo (número reduzido de componentes), alta velocidade de resposta (podendo responder os sinais de entrada na ordem nano–segundos), versatilidade (podem ser re-gravados até um milhão de vezes), e segurança do dispositivo (o circuito de controle esta galvanicamente isolado do circuito de potencia). A figura 7.4 mostra o controlador com tecnologia CPLD.

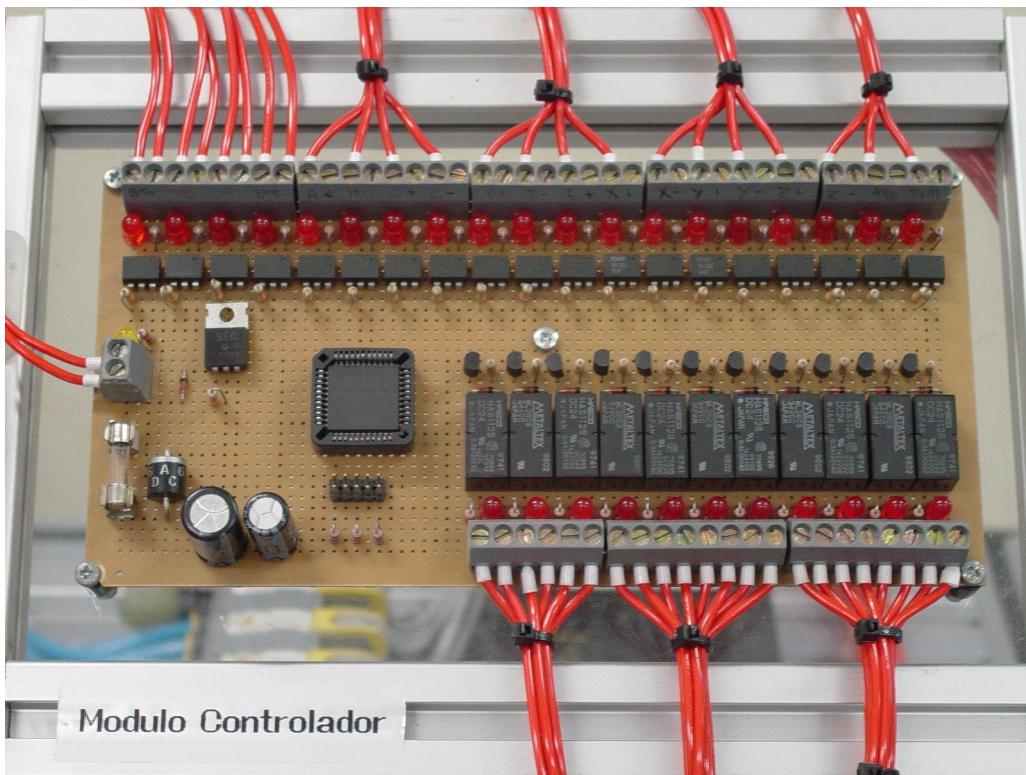


Figura 7.4 Controlador com tecnologia CPLD

### 7.3.1 Circuitos do Controlador

O controlador com tecnologia CPLD é composto por:

- Circuito de entrada
- Circuito de saída
- Circuito de alimentação
- Circuito lógico programável

O circuito de entrada é composto por 19 entradas que recebem os sinais vindos dos sensores e painel de comando, e são ligadas ao dispositivo por fotoacopladores do tipo TIL111, que isolam galvanicamente o dispositivo, já que o contato é feito por luz. Em cada uma das entradas foram ligados LEDs que indicam qual das entradas está recebendo um sinal a ser enviado para o dispositivo, os resistores que fazem parte desse circuito tem como função limitar a corrente no dispositivo e nos fotoacopladores. A figura 7.5 mostra o circuito de entrada e os componentes que fazem parte desse circuito.

As saídas do controlador com tecnologia CPLD, também isolam eletricamente o dispositivo lógico CPLD da parte de potencia composta pelas válvulas eletro-pneumaticas que comandam os atuadores, isso é possível através de relés eletromagnéticos que são chaveados por transistores do tipo C547B. Para esse processo foram necessárias 11 saídas, que ao serem acionadas acedem LEDs como modo de indicar seu funcionamento. A figura 7.5 mostra o circuito de saída e os componentes que fazem parte desse circuito.

O circuito de alimentação é responsável por manter a tensão necessária para alimentar o dispositivo e os demais componentes do controlador. Sua composição é feita basicamente por um fusível, que é o componente de segurança desse circuito, capacitores que filtram e atenuam as variações e oscilações de tensão, um diodo que evita as tensões reversas, um regulador linear de tensão 3.3 V e um LED que indica o funcionamento do controlador. Esse circuito está destacado na figura 7.5

O circuito lógico programável é composto apenas por dez pinos usados para gravação do dispositivo em circuito e o próprio CPLD, esse circuito é o responsável por receber e responder os sinais de entrada de acordo com a lógica que está gravado no CPLD utilizado no controlador. A figura 7.5 mostra os circuitos que compõem o controlador com tecnologia CPLD.

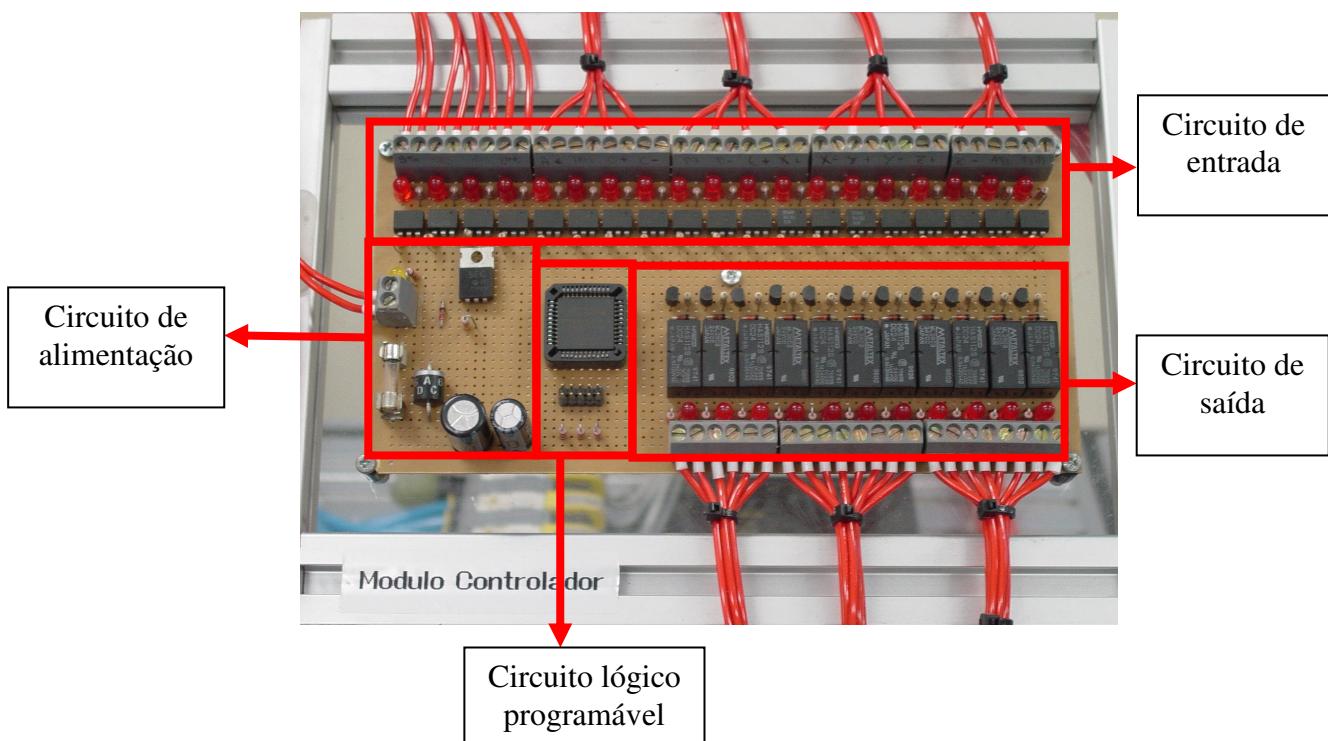


Figura 7.5 Circuitos que compõem o controlador com tecnologia CPLD.

## 7.4 Custos e Comparativos

Citado por diversas vezes, o baixo custo de desenvolvimento e alta velocidade de resposta desse tipo de controlador são as principais vantagens em relação aos controladores mais usados no mercado atualmente. A tabela 7.1 mostra um levantamento de componentes do controlador com tecnologia CPLD, que tem capacidade para automatizar processos com no máximo 34 I/O e um tempo de resposta na ordem de nano segundos.

Quantidade	Componente
1	Placa de circuito
1	Dispositivo programável de lógica complexa
1	Soquete para CI
1	Fusível
1	Porta Fusível
12	Diodo
1	Diodo retificador
65	Resistor
10	Pinos Hide
2	Capacitor
11	Transistor
1	Transistor de potencia
19	Foto acoplador
11	Rele magnético
31	LED
3	Suporte para borne

Tabela 7.1 – Relação de componentes do controlador

Caso decidíssemos optar pela aquisição de controladores lógicos programáveis de marcas conhecidas os chamados CLPs, os custos com o controlador do processo ficaria em média 84% mais caros que os controladores com

tecnologia CPLD. Outra desvantagem seria o fato dos CLPs de marcas conhecidas responderem aos sinais de entrada na ordem de milisegundos.

Apesar de já consolidada em outras áreas, o CPLD ainda não é difundido na automação da manufatura, isso acarreta um receio da indústria na aplicação e uma falta de mão de obra especializada na manutenção desse tipo de controlador. Devemos considerar que os CLPs são produtos taxados e que carregam o respaldo da marca de seus fabricantes e assistência técnica dos seus representantes, porém é valida a comparação, pois os controladores com tecnologia CPLD também podem ser considerados como uma opção ou solução para processos que exigem uma velocidade de resposta muito alta.

O fato dos CPLDs serem totalmente digitais, gera uma desvantagem, pois demanda maiores implementações de hardware e maiores conhecimentos técnicos para inserção de sinais analógicos, além da integração com outros sistemas e interfaces IHMs serem mais complexas sem o auxílio de microcontroladores.

## 8 Conclusão

Assim fica clara a possibilidade do uso da tecnologia CPLD no controle de automação industrial de pequeno e médio porte, apontando uma nova aplicação para o mercado crescente dos PLDs e sem exigir grandes conhecimentos técnicos ou acadêmicos para desenvolver um controlador lógico programável de baixo custo, alta performance e com linguagem de programação simples, atendendo aos requisitos propostos neste estudo.

Um processo controlado por um PLD da família MAX3000A, requer para manutenção apenas um conhecimento básico de técnicas digitais e treinamento na ferramenta de programação, essa facilidade de desenvolvimento viabiliza treinamentos em cursos de formação continuada e aprendizagem industrial ministrados por técnicos de nível médio.

O custo de manutenção desse tipo de controlador é baixo devido a sua segurança nos circuitos de entrada e saída de sinais. O dispositivo fica completamente isolado dos sensores por foto acopladores, o controlador aciona os atuadores e cilindros através de saídas à relé magnético. O número reduzido de

componentes para desenvolvimento desse tipo de controlador também é uma característica que facilita a manutenção e reduz os custos.

Constatamos que os controladores com tecnologia CPLD é capaz de responder aos sinais de entrada na ordem de nano segundos, isso se mostra também como uma vantagem em relação aos controladores CLPs de marcas conhecidas. Caso o técnico se depare com um processo de automação que requer do controlador uma resposta rápida o CPLD poderá ser aplicado.

Observamos durante as pesquisas uma escassez de material didático relacionado à tecnologia PLD, mesmo não influenciando no resultado desse trabalho, seria fundamental o aumento de publicações sobre o assunto para difusão da tecnologia em automação da manufatura. O surgimento destas publicações confirmariam as tendências de uso dos PLDs em processos automatizados.

Ao longo da pesquisa para realização deste trabalho a tecnologia dos dispositivos FPGAs mostrou um grande numero de publicações e diversas aplicações, sendo na maioria das vezes usados nos projetos baseados em tecnologia de hardware reconfigurável. Por ser um componente da família dos PLDs, os FPGAs possuem características semelhante aos dispositivos que foram analisados nesse estudo, mesmo não possuindo a mesma arquitetura interna. Essa diferença dá aos FPGAs capacidades maiores suportando circuitos lógicos ainda mais complexos.

Assim surgi como sugestão para desenvolvimentos futuros pesquisas e projetos de controladores lógicos programáveis para automação industrial de grande porte utilizando dispositivos com tecnologia FPGA. O otimismo em relação a essa tecnologia é quase uma unanimidade e fica estampada em citações de alguns especialistas, como essa:

“Um ASIC é como ter que utilizar um velho suéter azul da década de 70 para sempre. Com um FPGA, você tem um hardware, mas pode modificá-lo sempre que os estilos mudarem.” Shackelford (HP Labs), 2000.

Talvez esse trabalho isoladamente não tenha um grande significado, porém mostra com clareza o potencial dessa tecnologia para fins de automatizar um processo industrial, criando uma nova vertente no ramo e viabilizando aos técnicos em automação industrial uma nova possibilidade para seus projetos, confirmando a tendência de uso da tecnologia PLD que a cada dia se torna mais real.

## SIGLAS

ASICs	(Application Specific IC)
CPLD	(Complex Programmable Logic Device)
CLP	(Controlador Lógico Programável)
EDA	(Electronic Design Automation)
EPLDs	(Erasable PLDs)
EEPROM	(Electrical Erasable PROM)
FPGA	(Field Programmable Gate Array)
HDL	(Hardware Description Language)
I/O	(INPUT / OUTPUT)
ISP	(In System Programmability)
LAB	(Logic Arrays Block)
MPGAs	(Mask Programmable Gate Arrays)
PAL	(Programmable Array Logic)
PLAs	(Programmable Logic Arrays )
PLDs	(Programmable Logic Device)
PROM	(Programmable ROM)
ROM	(Read Only Memory)
SPLD	(Simple PLDs)
VLSI	(Very Large Scale Integration)

# **CPLD (Complex Programmable Logic Device) applied in manufacturing automation**

## **ABSTRACT**

Frequently used in several areas as Telecommunications and Computer Science, the PLDs (Programmable Logic Device) technology has characteristics that favor its use in industrial automation. First of all, we analyzed the evolutions of this technology and its devices, in order to find a programmable logical controller more versatile, more efficient and cheaper that could be used in didactic automation plants that simulate an industrial process.

Assisting the needs of the project we chose for the controller's development a CPLD (Complex Programmable Logic Device), of MAX 3000A, manufactured by Altera Corp. These devices are ideal for our didactic automation process.

This job also shows the powerful programming tool called EDA (Electronic Design Automation) of Quartus II®. In this software we could develop, compile and simulate the logical circuits, in a friendly way, and them recorded it in the chip, using the kit – PLD, release Altera MAX 3000A, built in the CNZ platform.

Consequently, we conclude that the PLD devices are logical controllers as reliable and effective as the most logical controllers already existent, and cheaper than the technology more used nowadays. As a result, the PLD devices are a technology that appears as a new option for industrial automation, bringing new ways to the automation technicians to work, become themselves more and more versatile and qualified for market demand.

**Orientação: Prof. Jeferson André Bigheti  
LENÇÓIS PAULISTA – CT 7.92  
Agosto / 2005**