

Revista da Graduação

Vol. 4

No. 1

2011

16

Seção: FACULDADE DE ENGENHARIA

**Título: Gerenciamento de filas de atendimento
usando ZIGBEE**

**Autores: Mayara Ferreira da Silva e Jorge Ferreira da Silva
Filho**

Este trabalho está publicado na Revista da Graduação.

ISSN 1983-1374

<http://revistaseletronicas.pucrs.br/ojs/index.php/graduacao/article/view/8659/6120>

GERENCIAMENTO DE FILAS DE ATENDIMENTO USANDO ZIGBEE

Mayara Ferreira da Silva*

Me. Jorge Ferreira da Silva Filho**

RESUMO

A Teoria de Filas de Atendimento engloba técnicas e soluções matemáticas, através de probabilidade, para gerenciar pontos de atendimento de forma mais ágil e eficaz possível. Por meio dos modelos matemáticos consegue-se encontrar um ponto ótimo, tanto em relação a número de servidores, como em questões como tempo de espera do cliente na fila e tempo de conclusão do serviço, quando o cliente é atendido.

Com esses dados em mãos, pode-se criar um sistema que gerencie essas filas e os guichês de atendimento, com o uso do microprocessador MSP430. Este pode fazer controle das senhas que serão retiradas pelos clientes (e que devem ser impressas), do painel eletrônico, onde aparecerá o número do próximo cliente a ser atendido, e também dos terminais que estão livres e quais estão sendo liberados para entrada de um novo cliente.

Para fazer a comunicação entre os terminais de atendimento com a impressora de senhas e o monitor (que exibe as senhas chamadas), será utilizado o módulo ZIGBEE, possibilitando uma comunicação sem utilização de fios entre eles, facilitando a instalação dos mesmos a certa distância uns dos outros.

Dentre outras soluções, pretende-se instalar este sistema na Secretaria da Faculdade de Engenharia da Pontifícia Universidade Católica do Rio Grande do Sul, conduzindo o atendimento aos alunos de maneira mais organizada e com agilidade nos processos, para que os mesmos estejam mais satisfeitos com o atendimento da unidade.

Palavras-chave: Teoria de Filas de Atendimento. Microprocessador MSP430. Módulo de Comunicação sem Fio ZIGBEE. Gerenciamento de Sistema de Filas.

* Acadêmico da Faculdade de Engenharia Elétrica da PUCRS. E-mail: may.ferreira@gmail.com

** Professor da Faculdade de Engenharia Elétrica da PUCRS. E-mail: jorgef@pucrs.br

1 INTRODUÇÃO

Atualmente vivemos no mundo cuja principal característica é a explosão de informações, que traz como consequência uma busca de soluções rápidas e inovadoras, sobretudo a agilidade nos processos. Uma sociedade que avança em níveis tecnológicos e econômicos necessita uma contínua melhora em questões de organização e gerenciamento, possibilitando uma potencialização dos esforços para o atendimento destes requisitos.

Em decorrência disto, cada vez mais se busca novas formas de gerenciamento de fluxos, tanto de informações como de pessoas, aliado ao uso intenso de novas tecnologias e sistemas de comunicação. Neste contexto, elabora-se este trabalho a fim de desenvolver um Gerenciador Eletrônico de Filas de Atendimento, o qual se utiliza para automatizar os diferentes sistemas de atendimento ao público.

Este projeto é implementado com o intuito de melhorar a performance no atendimento na secretaria da Faculdade de Engenharia da Pontifícia Universidade Católica do Rio Grande do Sul, podendo torná-lo mais ágil e eficaz, principalmente em épocas de maior concentração de pessoas, como período de matrículas, semana da engenharia, entre outros eventos.

O objetivo principal do trabalho é o gerenciamento da fila de espera para atendimento na secretaria geral do curso de Engenharia da PUCRS, procurando separar seus diferentes tipos e encaminhar os alunos de forma mais focada e objetiva a um profissional que mais aplicadamente o ajudará. Além disto, procura-se também gerenciar os tempos de espera e de atendimento, com o propósito de aumentar a eficácia dos profissionais que ali se encontram, e também deixar os alunos, ou outros indivíduos que procurarem atendimento, mais satisfeitos com o atendimento.

2 REFERENCIAL TEÓRICO

Este trabalho baseia-se na teoria de filas de espera, visto que no dia-a-dia todos enfrentam este dilema, seja no banco, supermercado, ou até mesmo no trânsito, quando ocorrem engarrafamentos. A Teoria das filas propõe que através de expressões matemáticas e propriedades mensuráveis das filas consiga-se obter um ponto ótimo, satisfazendo o cliente, em termos de tempo de espera e solução, e também sendo economicamente viável ao provedor dos serviços.

2.1 DESCRIÇÃO DO CONTEXTO

O sistema de filas pode ser descrito como um sistema em que os clientes, os quais não são necessariamente pessoas (como por exemplo, processos de uma máquina), chegam e ficam aguardando sua vez de serem atendidos; caso a fila esteja vazia, os mesmos são atendidos imediatamente e após a utilização dos serviços os clientes saem do sistema. Como na maioria das vezes os servidores encontram-se ocupados, os clientes devem esperar em uma fila (FCFS). A Figura 1 mostra um resumo simplificado do modelo de filas de espera.

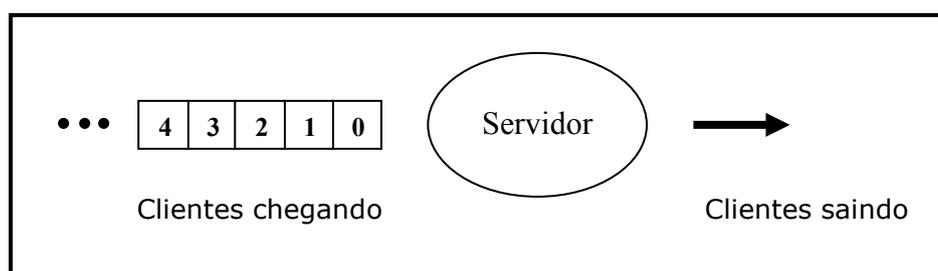


Figura 1. Modelo de Filas de Espera

2.2 CONCEITOS BÁSICOS

Para que se entenda melhor a teoria das filas de espera utilizam-se alguns parâmetros que as caracterizam. Desta forma, descrevem-se a seguir algumas delas, para maior compreensão do que é proposto nesta seção.

- **Centro de serviço** - Representa os recursos do sistema, compreendendo um ou mais servidores e um conjunto de clientes que esperam pelo serviço.
- **Fila** - Representa os clientes que estão esperando pelo serviço, juntamente com os que estão sendo atendidos pelos servidores.
- **Fila de espera** - Somente os clientes que estão aguardando pelo serviço.
- **F.I.F.O.** - First In First Out: o primeiro a entrar é o primeiro a sair.
- **F.C.F.S.** - First Come First Served: o primeiro a chegar é o primeiro a ser atendido.
- **LCFS** - (Last Come, First Served): Último a chegar, Primeiro a ser Atendido
- **LIFO** - (Last In, First Out): Último a Chegar, Primeiro a Sair. Aplicável em sistemas em que o item mais recente é mais fácil de ser recuperado, como por exemplo em sistemas de controle de estoque.

- **Padrão de Chegada dos Clientes** - É preciso conhecer a distribuição de probabilidade de chegada dos clientes, dado os tempos entre as sucessivas chegadas dos mesmos, ou até se chegam simultaneamente.
 - **Estacionário** – Que não muda com o tempo.
 - **Não-Estacionário** – Muda em diferentes tempos.
- **Padrões de Serviços** – O processo de serviço pode depender do número de clientes esperando pelo serviço, ou seja, o servidor pode trabalhar mais rápido se a fila estiver aumentando, ou, caso contrário, pode ser tornar confuso e ficar mais lento. Isto se chama serviço dependente do estado.
- **Disciplina de Filas** - A disciplina de filas refere-se ao modo pelo qual são escolhidos os clientes para entrar em serviço assim que a fila estiver formada. A mais comum é F.I.F.O, porém existem outros tipos. Verificamos duas situações gerais de prioridade: preemptiva, onde o cliente com mais alta prioridade entra em serviço independente de outro com menor prioridade estar sendo servido ou não-preemptiva, onde os clientes com mais alta prioridade vão para o início da fila, mas só entram em serviço quando o cliente sendo atendido deixa o sistema, mesmo que ele tenha baixa prioridade.
- **Número de Clientes (M)** - Número de clientes que chegam na fila e que pode ser infinito.
- **Capacidade do Sistema (K)** - Limitação física da quantidade de espaço na fila de atendimento de um cliente e a conseqüente diminuição do tamanho da fila.
- **Número de Canais de Serviço** - Número de estações de serviços paralelos que podem servir os clientes ao mesmo tempo. Temos dois tipos de sistema multicanais: um que possui uma única fila (Figura 2), e o outro que possui filas específicas (Figura 3) para os servidores (diferentes tipos de serviços). É geralmente assumido que os mecanismos de canais paralelos operam independentemente um do outro.
- **Utilização** - Fatia de tempo que o sistema permanece ocupado, atendendo a clientes.
- **População** - Quantidade de atendimentos a serem feitos em um determinado instante.
- **Tempo de resposta** - Intervalo de tempo entre o pedido e o início/conclusão do serviço.

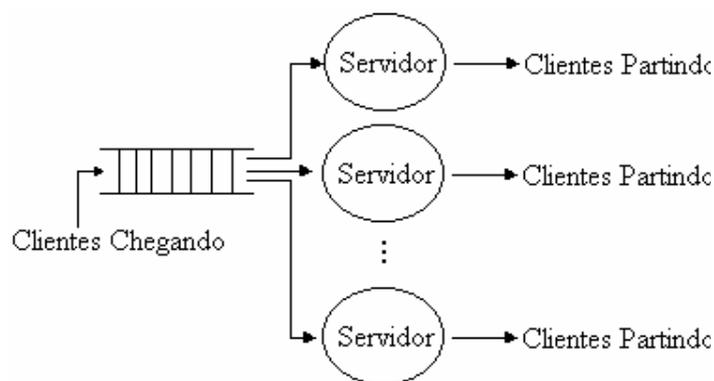


Figura 2. Sistemas Multicanais – Fila Única

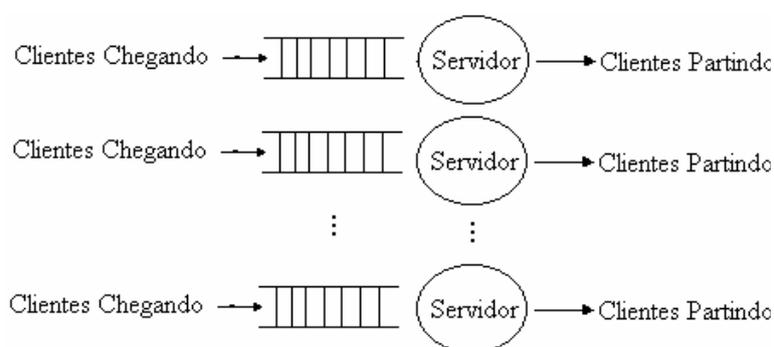


Figura 3. Sistemas Multicanais – Filas Independentes

- **Estágios de Serviço** – Um sistema de filas pode ter um único estágio de serviço ou pode ter vários estágios.
- **Round-robin (algoritmo)** - cada cliente recebe uma fatia de tempo do servidor (quantum), dentro da qual é atendido. Após o término do quantum, se a atividade não foi completada, o cliente é retirado e outro passa a ser atendido. Posteriormente, o cliente que foi interrompido retorna ao servidor e continua a sua atividade.

2.3 PARAMETRIZAÇÃO DE SISTEMAS

Sendo λ a taxa média de chegadas ao sistema (clientes/seg) e μ a taxa média de serviços (serviços/seg), o tempo de chegada entre clientes é, em média, $\frac{1}{\lambda}$, e o tempo de serviço de cada cliente é, em média, $\frac{1}{\mu}$ (ver Figura 5).

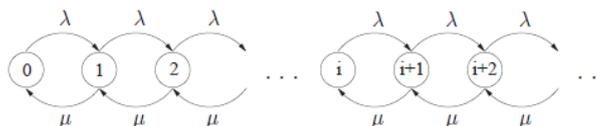


Figura 4. Parametrização de Sistemas

Logo, é possível observar-se a razão $\rho = \frac{\lambda}{\mu}$ (intensidade de tráfego), mostrando que:

- Se $\rho > 1$, há mais chegadas do que saídas de clientes. O número de clientes no sistema é ilimitado, o sistema é instável;
- Se $\rho < 1$, há mais saídas do que chegadas de clientes. Existe uma solução estacionária para o sistema;
- Se $\rho = 1$, significa que chega, em média, o mesmo número de clientes que saem do sistema. Qualquer número de clientes no sistema é equiprovável, o sistema é instável

Um Processo de Chegada e Saída de Clientes é caracterizado pela dupla (M, T) , onde:

M é o conjunto de estados; $T(x, y)$ é taxa de ocorrência da transição do estado x para o estado y . T é a função de transição a qual associa uma taxa de ocorrência de uma transição de um estado para outro. A função T possui domínio em $M \times M$ e contradomínio nos \mathfrak{R}^+ .

2.3.1 NOTAÇÕES

Supondo uma disciplina de serviços FCFS, temos a seguinte notação:

A – Função de Probabilidade Tempo de Chegada

B - Função de Probabilidade Tempo de Serviço

m – número de servidores

K – capacidade do sistema

M – número de clientes

Sendo (K) e (M) infinitos, pode-se utilizar apenas as notações restantes.

Os valores A e B têm valores que representam a distribuição a utilizar. O valor M representa uma distribuição exponencial negativa dos tempos conhecida por Markov ou Poisson. No gráfico seguinte (Figura 4) está representada a distribuição de probabilidades de chegarem n clientes num intervalo de 1 segundo quando a média é de 4 clientes por segundo.

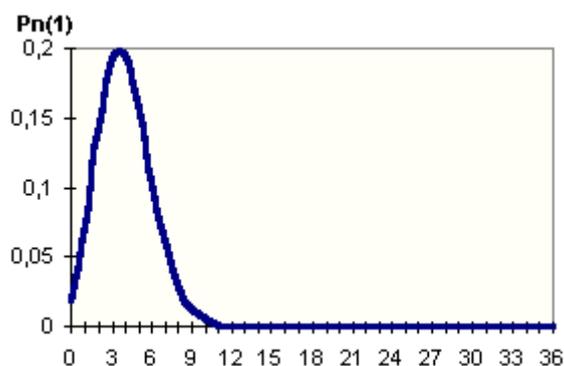


Figura 5. Distribuição de Markov de Média 4 Clientes por Segundo

Já pela notação de Kendall (Tabela 1) temos o seguinte:

A – Função de Probabilidade Tempo de Chegada

B - Função de Probabilidade Tempo de Serviço

m – número de servidores

K – capacidade do sistema

M – disciplina de serviço

Da mesma forma, sendo (M) do tipo FIFO e (K) infinita, pode-se utilizar apenas as notações restantes.

Características	Símbolo	Explicação
Distribuição de Tempo de Interchegada (A) e Distribuição de Tempo de Serviço (B)	M	Exponencial
	D	Determinístico
	E_k	Tipo k-Erlang ($k = 1, 2, \dots$)
	H_k	Mistura de k exponenciais
	PH	Tipo Fase
	G	Geral
Número Paralelo de Servidores (m)	$1, 2, \dots, \infty$	
Restrição na capacidade do sistema (k)	$1, 2, \dots, \infty$	
Disciplina da fila (M)	FCFS	First Come First Served
	LCFS	Last Come First Served
	RSS	Seleção Aleatória por Serviço
	PR	Prioridade
	GD	Disciplina Geral

Tabela 1. Notação Kendall

2.3.2 SISTEMA DO TIPO M M 1

Quando o sistema possui capacidade infinita e um servidor, e distribuição de tempo de chegada e de serviço do tipo Exponencial ($M/M/\infty/1$), sendo também chamado independente de carga, a probabilidade de estar no estado i depende da probabilidade de estar no estado $i-1$ e das taxas de saída de $i-1$ para i e de retorno de i para $i-1$. Como o somatório de

todas as probabilidades tem que ser 1, a probabilidade do primeiro estado (estado 0) é determinada sabendo-se a soma de todas outras probabilidades. Desta maneira, para $i > 0$ temos que a probabilidade de se encontrar no estado $i (P_i)$ é:

$$\lambda P_K = \mu P_{K+1} \text{ estando o número de chegadas e saídas do sistema em equilíbrio}$$

$$P_{K+1} = \rho \cdot P_K \quad P_K = \rho^K \cdot P_0$$

$$P_i = P_0 * \left(\frac{\lambda}{\mu}\right)^i \text{ e } P_0 = 1 - \frac{\lambda}{\mu}$$

2.3.3 SISTEMA DO TIPO M M C K

Quando o sistema possui capacidade finita, podendo ter vários servidores (M/M/C/K), sendo também chamado dependente de carga, pode ser solucionado (obter as probabilidades para cada estado) da seguinte forma:

- ✓ primeiramente atribui-se um peso qualquer para o estado $i = 0 (W_0)$. Para facilidade de cálculos pode-se utilizar $W_0 = 1$;
- ✓ para obter os pesos dos estados onde $i > 0$ utiliza-se a fórmula para o cálculo de probabilidades:

$$W_i = W_{i-1} * \left(\frac{\lambda}{\mu}\right) = \left(\frac{\lambda}{\mu}\right)^i * W_0$$

- ✓ assim obtém-se $W_0..W_K$ pesos para os estados 0 a K. Estes pesos mantêm, entre eles, as mesmas proporções que as probabilidades $P_0..P_K$. Assim, devem-se normalizar os resultados obtidos para que a soma das probabilidades seja 1, fazendo:

$$P_i = \frac{W_i}{\sum_{j=0}^K W_j}$$

Quando $C > 1$, o modo de resolução é bastante similar, porém, após atribuir um peso ao estado $i = 0$ (por exemplo, $W_0 = 1$), utiliza-se a seguinte fórmula para obter pesos para os demais estados:

$$W_i = W_{i-1} * \left(\frac{\lambda}{\min(i, C) * \mu}\right)$$

Onde $\min(x,y) = x$, se $x < y$ ou $\min(x,y) = y$ em caso contrário. A partir disso, os resultados são normalizados como no caso anterior para obter as probabilidades.

2.4 OCUPAÇÃO MÉDIA DA FILA DE ESPERA

O cálculo da ocupação média da fila de espera pode ser um parâmetro importante para o dimensionamento da mesma, visto que num sistema real não se tem uma fila com capacidade infinita.

Sendo N o número de elementos na fila, escreve-se que:

$$N = \sum_{K=0}^{\infty} K \cdot P_K = \sum_{K=0}^{\infty} K \cdot P_0 \cdot \rho^K = P_0 \sum_{K=0}^{\infty} K \cdot \rho^K$$

Efetua-se a mudança de variável $K = n - 1$, onde:

$$N = P_0 \cdot \left(\frac{1}{(1-\rho)^2} - \sum_{n=1}^{\infty} \rho^{n-1} \right), \text{ e repondo a variável } K, N = P_0 \cdot \left(\frac{1}{(1-\rho)^2} - \sum_{K=0}^{\infty} \rho^K \right)$$

$$\text{E por fim, } N = \frac{\lambda}{\mu - \lambda}.$$

2.5 ATRASO MÉDIO NO SISTEMA

O atraso médio no sistema (T) é o tempo que decorre desde que um cliente entra no sistema até que sai do mesmo (servido). É possível decompor o atraso médio no sistema numa componente de espera na fila (T_w) e uma componente de tempo de serviço (T_s), sendo

$$T_w = \frac{N}{\mu} \text{ e } T_s = \frac{1}{\mu}.$$

$$T = T_w + T_s = \frac{N+1}{\mu} = \frac{\left(\frac{\rho}{1-\rho}\right) + 1}{\mu} = \frac{1}{(1-\rho)\mu} \therefore \lambda \cdot T = N$$

Conseqüentemente chegamos a expressão de Little: $N = \lambda \cdot T$

$$\text{Por outro lado pode-se obter também } T = \frac{1}{(1-\rho) \cdot \frac{\lambda}{\rho}} = \frac{1}{(\mu - \lambda)}.$$

E então chegar a equações para o número médio de clientes em espera (N_w) e para o número médio de clientes sendo atendidos (N_s):

$$T = T_W + T_S \quad \therefore \quad N = \lambda \cdot T_W + \lambda \cdot T_S$$

$$N_W = \lambda \cdot T_W = \lambda \cdot \frac{N}{\mu} = \rho \cdot N \quad N_S = \lambda \cdot T_S = \lambda \cdot \frac{1}{\mu} = \rho$$

2.6 EXPRESSÃO DE POLLAZECK-KHINCHINE

A Expressão de Pollazeck-Khinchine permite determinar a ocupação média do sistema em função do desvio padrão (σ) e média do tempo de serviço (T_S):

$$N = N_S + N_W = \rho + \frac{\rho^2 \cdot \left[1 + \left(\frac{\sigma}{T_S} \right)^2 \right]}{2 \cdot (1 - \rho)}$$

Para uma fila M/M/1, o desvio padrão é igual a T_S , então $N = \rho + \frac{\rho^2}{(1 - \rho)}$.

Já para uma fila M/D/1, o desvio padrão é igual a zero, e $N = \rho + \frac{\rho^2}{2 \cdot (1 - \rho)}$.

Logo, percebemos que nas componentes N_W , $N_W(M/M/1) = 2 \times N_W(M/D/1)$.

E, usando a expressão Little, verificamos o atraso médio do sistema:

$$T = \frac{\rho}{\lambda} + \frac{\rho^2 \cdot \left[1 + \left(\frac{\sigma}{T_S} \right)^2 \right]}{2\lambda \cdot (1 - \rho)} = \frac{1}{\mu} + \frac{\rho \cdot \left[1 + \left(\frac{\sigma}{T_S} \right)^2 \right]}{2\mu \cdot (1 - \rho)}$$

2.7 FILAS FINITAS

Na prática as filas não podem crescer livremente, supondo uma capacidade finita, temos:

$$\sum_{K=0}^{N_0} P_K = 1, \quad P_K \cdot \lambda = P_{K+1} \cdot \mu \quad P_K = P_0 \cdot \rho^K$$

$$P_0 = \frac{1 - \rho}{1 - \rho^{N_0+1}}$$

Temos a probabilidade de bloqueio P_{N_0} : $P_B = P_{N_0} = P_0 \cdot \rho^{N_0} = \frac{1 - \rho}{1 - \rho^{N_0+1}} \cdot \rho^{N_0}$.

2.8 MÉTODO OPERACIONAL

2.8.1 QUANTIDADES OPERACIONAIS

São quantidades que podem ser medidas diretamente durante um período finito de observação.

- Período de observação: T
- Número de chegadas (arrivals): A_i
- Número de términos (completions): C_i
- Tempo ocupado (busy time): B_i
- Taxa de chegada: $\lambda_i = \frac{A_i}{T}$
- Throughput: $X_i = \frac{C_i}{T}$
- Utilização: $U_i = \frac{B_i}{T}$
- Tempo médio de serviço: $S_i = \frac{B_i}{C_i}$

2.8.2 LEI DA UTILIZAÇÃO

$$U_i = \frac{B_i}{T} = \frac{C_i}{T} \times \frac{B_i}{C_i} \quad \text{e} \quad U_i = X_i \cdot S_i$$

2.8.3 LEI DE LITTLE

$$Q_i = \lambda_i \cdot R_i$$

Q_i : Número médio de clientes; R_i : Tempo médio de resposta

$$R_i = S_i + W_i$$

S_i : Tempo médio de serviço; W_i : Tempo médio de espera

Se o sistema está em equilíbrio, a taxa de chegada é igual ao throughput, portanto:

$$Q_i = X_i \cdot R_i$$

3 METODOLOGIA

A metodologia aplicada permeia um estudo mais aprofundado da teoria de filas de espera, bem como suas soluções matemáticas, e no desenvolvimento de um módulo capaz de fazer o gerenciamento das informações, interligando todos os equipamentos, além de operacionalizar uma comunicação sem fio entre os mesmos.

Para tal, será pesquisado inicialmente as teorias quanto ao sistema de filas de espera e modelos de probabilidade, para que se possa obter algum conhecimento teórico do que será implementado.

Logo depois se atua na construção do módulo de controle e gerenciamento dos equipamentos. Um módulo irá controlar os tickets que serão impressos com o número de atendimento, conforme chegada dos clientes a fila e também a comunicação pelo painel eletrônico do número a ser chamado. O outro módulo vai monitorar os guichês de atendimento, informando quais estão liberados para o próximo cliente da fila.

Para comunicação entre estes dois módulos, será utilizado o módulo ZigBee, ZigBee Pro da RogerCom, que estabelecerá uma comunicação sem fios entre os dois módulos. No capítulo 4.5.2 será desenvolvido o protocolo para aplicação neste trabalho.

A seguir temos um quadro-resumo dos dois módulos, os quais irão se comunicar através do módulo ZigBee:

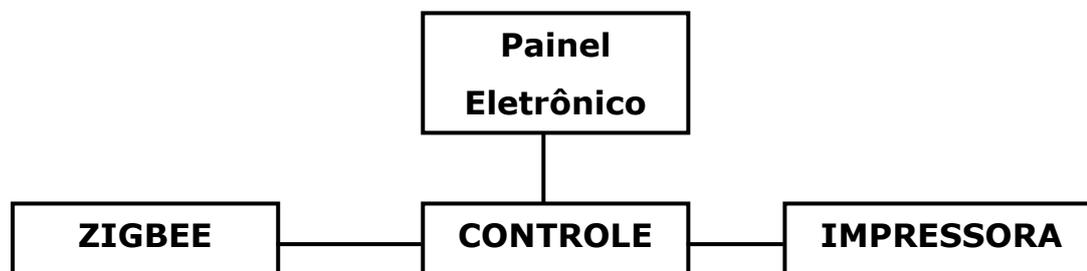


Figura 6. Módulo I – Impressão de Senha e Painel Eletrônico



Figura 7. Módulo II – Controle dos Guichês de Atendimento

Primeiramente será implementado o software de controle, utilizando a placa de processadores 2010/1, onde estará o microprocessador MSP430 que terá dois procedimentos para controlar. Um deles será a geração das senhas de atendimento, incrementada a cada cliente que chegar e sendo zerada ao final do dia, para que reinicie no próximo, e enviando a

numeração para impressora que ejetará um comprovante para o cliente. Nesta etapa o cliente deverá escolher entre algumas opções a que melhor atende a sua requisição, para que possa ser encaminhado ao terminal correto para solucionar seu problema. O outro será enviar a senha e o guichê, que será recebido através do módulo Zigbee, para serem exibidos no monitor; o número indicado corresponderá ao próximo cliente a ser chamado.

Logo após, será criada uma segunda placa, também contendo o microprocessador MSP430, que então irá controlar os guichês de atendimento, monitorando quais estão livres e quais vêm sendo liberados para atender os clientes que aguardam na fila. Ele deverá encaminhar o cliente para um terminal que possa atender a sua necessidade, devendo avaliar também se o terminal liberado corresponde a um serviço válido para o mesmo.

Até este momento os dois módulos estarão prontos, porém será necessário que os mesmos se comuniquem entre si. Esta comunicação se dará através do módulo ZigBee, que será a próxima etapa do projeto a ser cumprida, implementando um protocolo para que as duas unidades possam transmitir e receber informações. O Módulo 1 deverá enviar a quantidade de clientes aguardando na fila, e o Módulo 2 os guichês livres para atendimento do cliente que está na fila.

Por fim, todos os testes serão realizados, simulando diversas situações, com diferentes tipos de atendimento, até que seja comprovado que o sistema está operando corretamente.

4 APLICAÇÃO PRÁTICA

Este projeto pretende ser desenvolvido na secretaria da Faculdade de Engenharia Elétrica da Pontifícia Universidade Católica do Rio Grande do Sul. A PUCRS é uma instituição de ensino superior, de pesquisa e de extensão, constituída por um conjunto de unidades (faculdades, institutos, etc.), que promove a formação profissional e científica de pessoal de nível superior, a realização de pesquisa teórica e prática nas principais áreas do saber, o armazenamento e a divulgação de seus resultados e a promoção de atividades de extensão.

Diante do exposto, o primeiro passo foi consultar as secretárias da Faculdade de Engenharia, que convivem dia-a-dia no ambiente, verificando o tipo de atendimento e alguma particularidade que ali pudesse haver. As mesmas informaram que todas atendem a todos, não existindo divisão de funções de atendimento e que cada uma possui um guichê fixo.

Então se segue o trabalho através da implementação e programação de software e hardware.

4.1 O MICROCONTROLADOR MSP430F2619

Para implementação deste projeto foi utilizado um kit de desenvolvimento, elaborado para as disciplinas de Laboratório de Processadores da Faculdade de Engenharia; o mesmo utiliza como microcontrolador, o MSP430F2619 da Texas Instruments, que pertence a família do MSP430, os quais são muito usuais em projetos e possuem características como: reduzido consumo, rapidez e capacidade de processamento, além de possuir diversos periféricos que permitem um interfaceamento rápido e eficaz com diversos tipos de periféricos externos.

Algumas características do microprocessador MSP430F2619:

- Arquitetura RISC de 16bits;
- Frequência de trabalho de até 16MHz;
- Memória RAM de 4kB;
- Memória FLASH de 120kB;
- Alimentação facilmente fornecida de 1.8V a 3.3V;
- Protocolos internos para comunicação serial UART (Universal Asynchronous Receiver/Transmitter), SPI (Serial Peripheral Interface) e I2C (Inter-Integrated Circuit);
- Ferramenta de desenvolvimento MSPGCC.

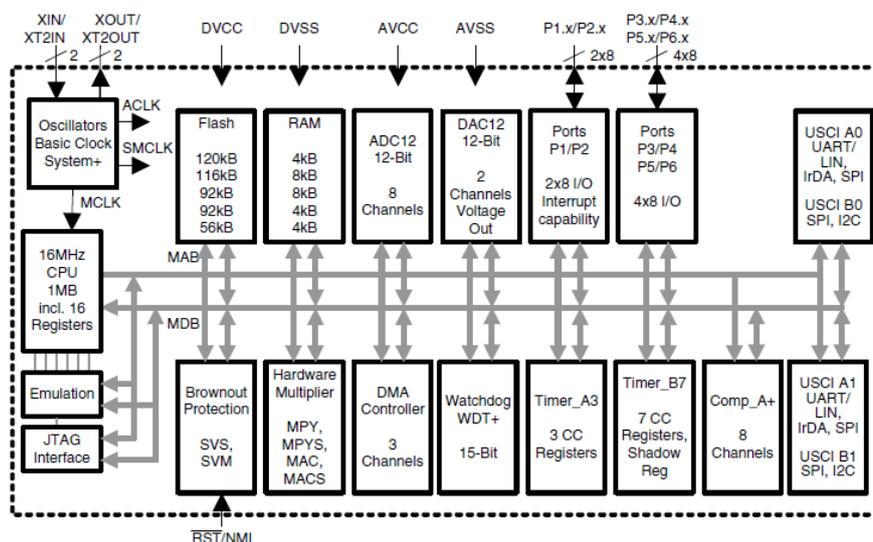


Figura 8. Diagrama de blocos funcional do MSP430F2619

O Diagrama de blocos funcional do MSP430F2619 mostra a arquitetura interna do MSP430F2619, através de um diagrama de blocos.

4.2 ESQUELETO DO PROGRAMA

Inicia-se a programação dos módulos através de um fluxograma, que indica todos os blocos necessários, com possibilidade de testes ao longo do mesmo. Utiliza-se a linguagem C, com auxílio do compilador MSPGCC.

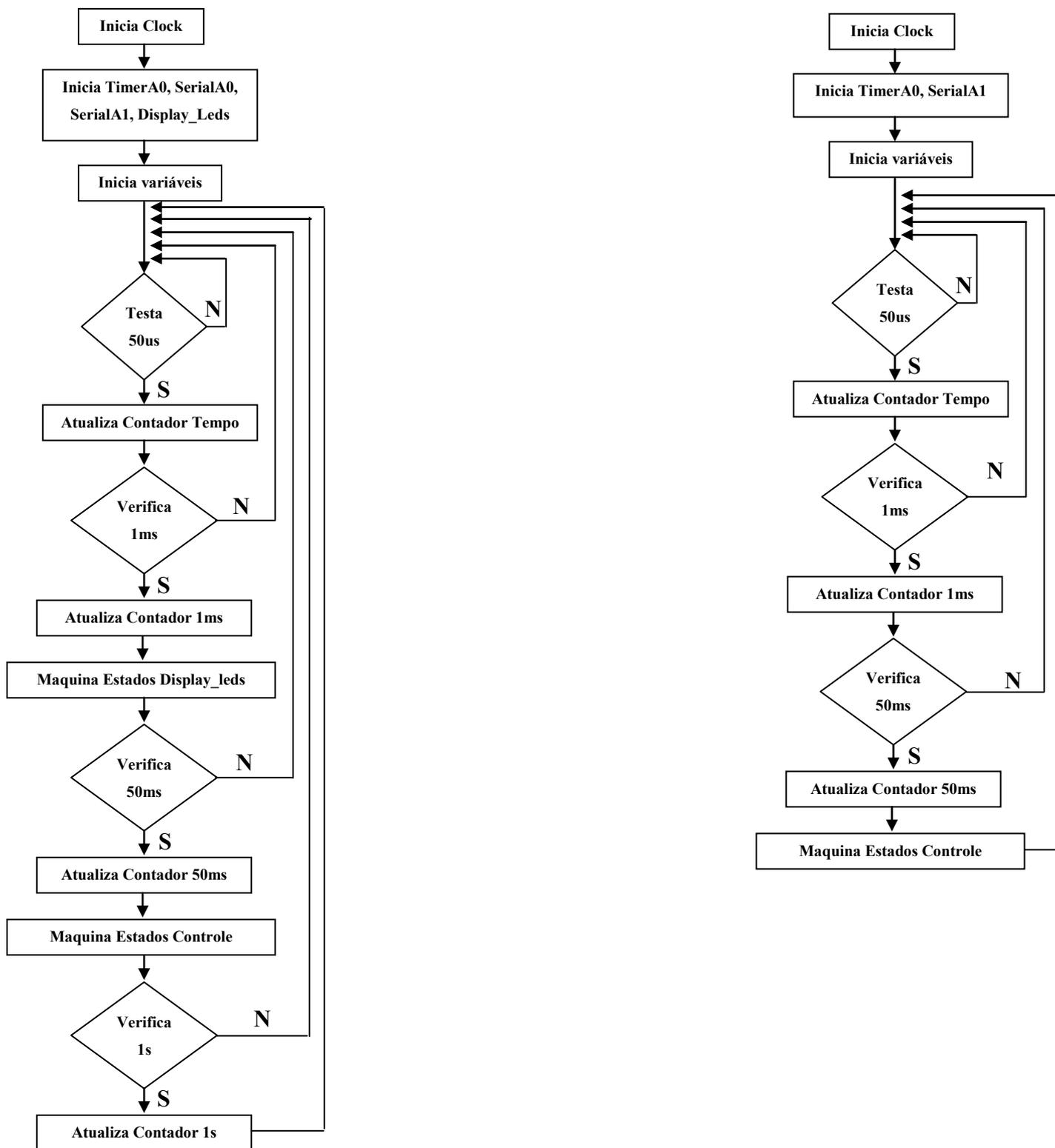


Figura 9. Fluxograma de base para o desenvolvimento do projeto.

Para o Módulo I, inicializa-se o Clock de 16MHz via cristal, e também as interfaces necessárias: TimerA0, SerialA0, SerialA1 e Display_Leds, assim como as variáveis utilizadas no programa. O delay base é fixado em 50us, dividindo-se o clock por 800, e a cada ciclo de interrupção incrementa-se uma variável, até atingir 20 contagens e chegar ao valor de 1ms, e a variável é resetada.

A partir de 1ms, iniciam-se dois novos contadores, um para 50ms, que contará até 50 e o outro para 1s, que contará até 1000. Nesta rotina de 1ms também é chamada a máquina de estados para o Display de Leds. A cada 50ms, iniciam-se as rotinas de controle da impressora e da escrita no display de leds, e a cada segundo, atualizam-se as funções do relógio. Todas essas rotinas serão detalhadas adiante.

Já o Módulo II é um pouco mais simples, devido apenas controlar os caixas livres e fazer o envio da informação via ZigBee. A inicialização do clock ocorre da mesma forma e as interfaces utilizadas são TimerA0 e SerialA1. Igualmente a indicação das variáveis do programa e o ajuste dos delays de 50us, 1ms e 50ms. As funções do programa pertencem todas ao delay de 50ms, e também serão detalhas no que segue.

4.3 MÓDULO 1 – DETALHAMENTO

4.3.1 DISPLAY DE LEDS

O display de leds utilizado no trabalho tem um acionamento serial por linha e por coluna, são 7 linhas e 140 colunas, das quais 70 de leds verdes e mais 70 de leds vermelhos, e caso acenda-se as duas cores, tem-se a cor amarela. Conforme Figura 10, estando o dado em 1 habilita o led, porém deve estar tanto na linha como na coluna para ser aceso.

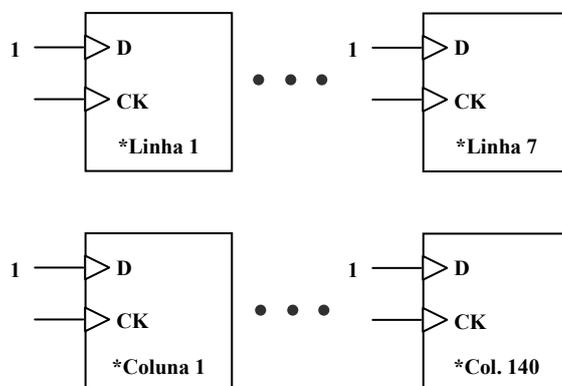


Figura 10. Flip-Flops de Linha e Coluna

Os leds são acesos por linha, ou seja, uma linha de cada vez, devido ao display ter uma fonte de corrente para cada coluna, garantindo assim o brilho máximo. Se for acesa mais de uma linha na mesma coluna, a corrente será a mesma, diminuindo o brilho dos leds.

4.3.2 CONTROLE BOTÃO E IMPRESSÃO

Quando um cliente chega na secretaria, aciona um botão e é impresso um ticket com a senha para atendimento. Este processo é controlado a partir de uma máquina de estados como será descrito a seguir.

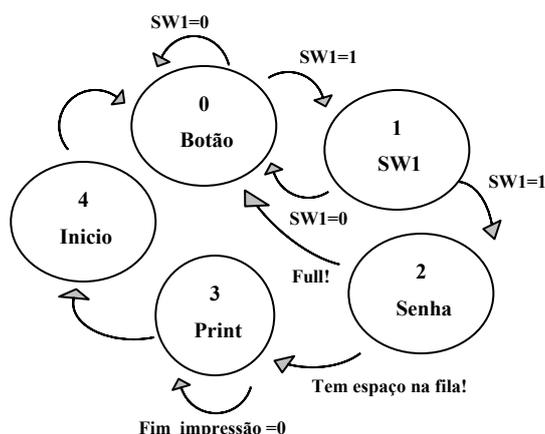


Figura 11. Máquina de Estados de Controle do Botão e Impressão

No estado 0, o botão SW1 fica sendo monitorado, aguardando a chegada de clientes. Ao ser pressionado, é feito um teste de debounce através do estado 1, e caso positivo, passando ao estado 2. Neste estado faz-se o incremento da senha e verificação do estado da fila: estando cheia volta-se ao estado 0, e se houver espaço, salta-se para o estado 3, onde será feita a impressão. Estando a impressora livre, será impressa a senha e incrementada a máquina para o estado 4, onde será alterada a variável de controle das posições na fila, e após retornará ao estado 0. Porém se já estiver em processo de impressão permanecerá no estado 3, aguardando liberação para impressão.

4.3.3 CONTROLE ESCRITA NO DISPLAY DE LEDS

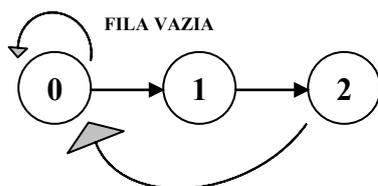


Figura 12. Máquina de Estados Controle Escrita no Display de Leds

O controle da escrita no display é uma rotina que é chamada apenas se o ZigBee receber uma mensagem, acionando automaticamente esta função.

No estado 0, verifica se existem clientes na fila, caso contrário, permanece no mesmo. Havendo clientes na fila, habilita estado 1, onde irá enviar o número da senha e do guichê do próximo atendimento, os quais serão escritos no display de leds. Então vai ao estado 2 e incrementa o final da fila, indicando próximo cliente a ser atendido.

4.4 MÓDULO 2 – DETALHAMENTO

4.4.1 CONTROLE DE GUICHÊS

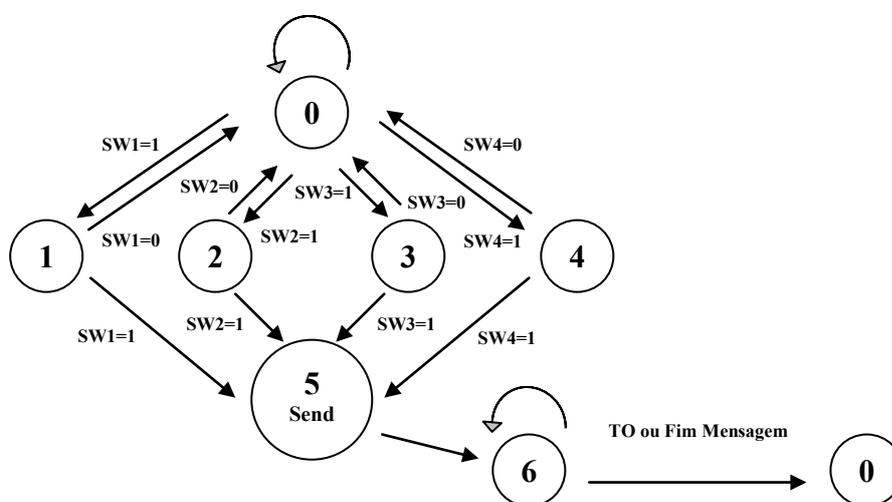


Figura 13. Máquina de Estados de Controle dos Guichês de Atendimento

Quando o terminal de atendimento está liberado, o atendente pressiona um botão. Cada atendente tem um botão diferente, que ao ser pressionado, indicará o próximo estado da máquina. Sendo SW1, passará ao estado 1, SW2, ao estado 2, SW3 ao estado 3 e SW4 ao estado 4. Em todos estes estados verifica-se o debounce, que sendo confirmado o pressionamento do botão, pulará para o estado 5, onde entrará na função de envio da mensagem, e irá ao estado 6, verificando o envio de toda a mensagem. Caso toda mensagem tenha sido enviada então voltará ao estado 0, ou ficará aguardando envio, e só retornará ao estado 0 se ocorrer Timer Out.

4.5 ZIGBEE

4.5.1 CONFIGURAÇÃO ZIGBEE

Para setar as configurações do módulo ZigBee adquirido, utilizou-se um adaptador USB, que através do software X-CTU, disponível no site do fornecedor, permite programar os parâmetros da comunicação via rádio, como: endereço dos módulos, determinação do mestre e dos escravos, programação de portas de entrada e saída, interface de comunicação com o processador.

Conforme ajustada a serial, seguiu-se os mesmo padrões para o ZigBee: 8 bits de dados, sem paridade, 1 stop bit e taxa de 9600bps. Na Figura 14 encontra-se a tela principal do programa X-CTU.



Figura 14. Tela do Programa X-CTU – Configuração ZigBee

4.5.2 PROTOCOLO ZIGBEE

Para comunicação entre os módulos do ZigBee Pro, foi estabelecido um protocolo específico, através de máquinas de estados, detalhadas abaixo.

O esquema abaixo representa a Máquina de Estados pertencente ao Módulo I:

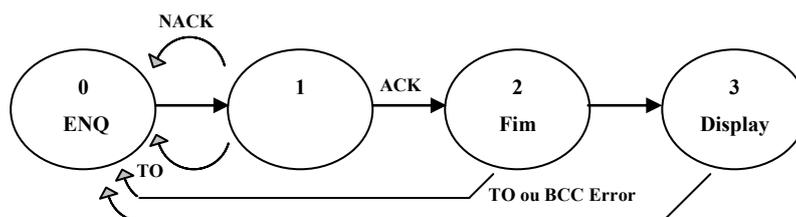


Figura 15. Máquina de Estados ZigBee Módulo I

No estado 0, o mestre envia um “enquire” ao escravo, perguntando se este tem algo para transmitir, e salta para o estado 1. Caso escravo responda com um ACK, vai para o estado 2, onde o mestre receberá a mensagem do escravo, verificando se foi recebida toda mensagem. Do contrário, recebendo um NACK, ou mesmo se exceder o limite de tempo (TO), voltará ao estado 0.

Estando no estado 2, havendo erro de BCC, ou esgotando limite de tempo, voltará ao estado 0. Se toda mensagem foi recebida sem erros, passará ao estado 3, onde processa informação recebida para sinalizar o Controle de Escrita no Display, qual caixa está livre para o próximo cliente, e depois retornando ao estado 0.

O esquema abaixo representa a Máquina de Estados pertencente ao Módulo II:

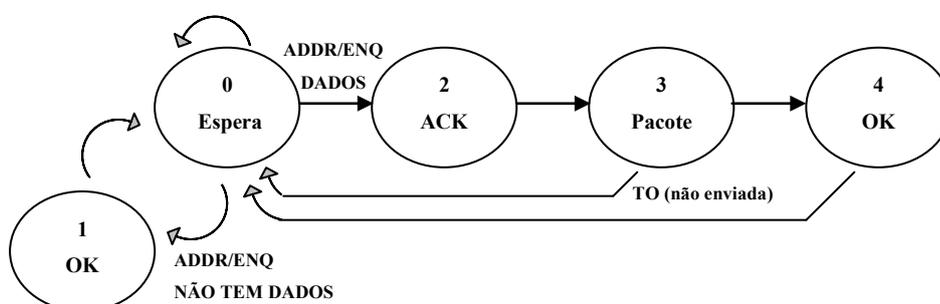


Figura 16. Máquina de Estados ZigBee Módulo II

No estado 0 fica aguardando, se receber um pacote verifica se é para o seu endereço e se é do tipo “enquire” e caso positivo vai para o estado 2 (se houver dados a serem enviados) ou para o estado 1 (se não houver guichês livres). Estando no estado 1, vai enviar um NACK e voltar ao estado 0. Porém, se estiver no estado 2, vai enviar um ACK ao mestre e pular para o estado 3. No estado 3 vai enviar a mensagem e incrementar para o estado 4. Então, se a mensagem chegar corretamente ao Módulo I, receberá um ACK do mestre e retornará ao estado 0, liberado para outra transmissão, ou do contrário, atingindo Time Out, irá ao estado 0 para retransmitir a mensagem.

Estrutura das Mensagens trocadas entre Mestre e Escravo, segundo protocolo adotado:

ADDR	STX	MENSAGEM	ETX	BCC
ADDR	ENQ	BCC		
ADDR	NACK	BCC		
ADDR	ACK	BCC		

ADDR – Endereço do Zigbee Escravo.

STX – Inicia buffer (02H).

MENSAGEM – Informação trocada entre os módulos

ETX – Termina buffer (03H).

BCC – Block Check Character (caracter de checagem do bloco).

ENQ – Pergunta se terminal tem mensagem (05H).

ACK – Tem mensagem para enviar.

NACK – Não tem mensagem a ser enviada.

4.6 SERIAL

A comunicação da porta serial foi criada de forma assíncrona, sem bit de paridade, com pacotes de 8 bits de dado e 1 stop bit, e a uma taxa de 9600bps. Esta configuração foi utilizada tanto na comunicação com a impressora, como com o ZigBee.

4.7 DESENVOLVIMENTO DE HARDWARE

O Kit de Laboratório de Processadores utilizado no trabalho não possui interface serial RS232 para a comunicação com a impressora, logo, foi necessária a implementação de uma placa contendo esta interface e também uma conexão para o adaptador do ZigBee.

Segue abaixo o esquemático da mesma:

SIPSOC-10

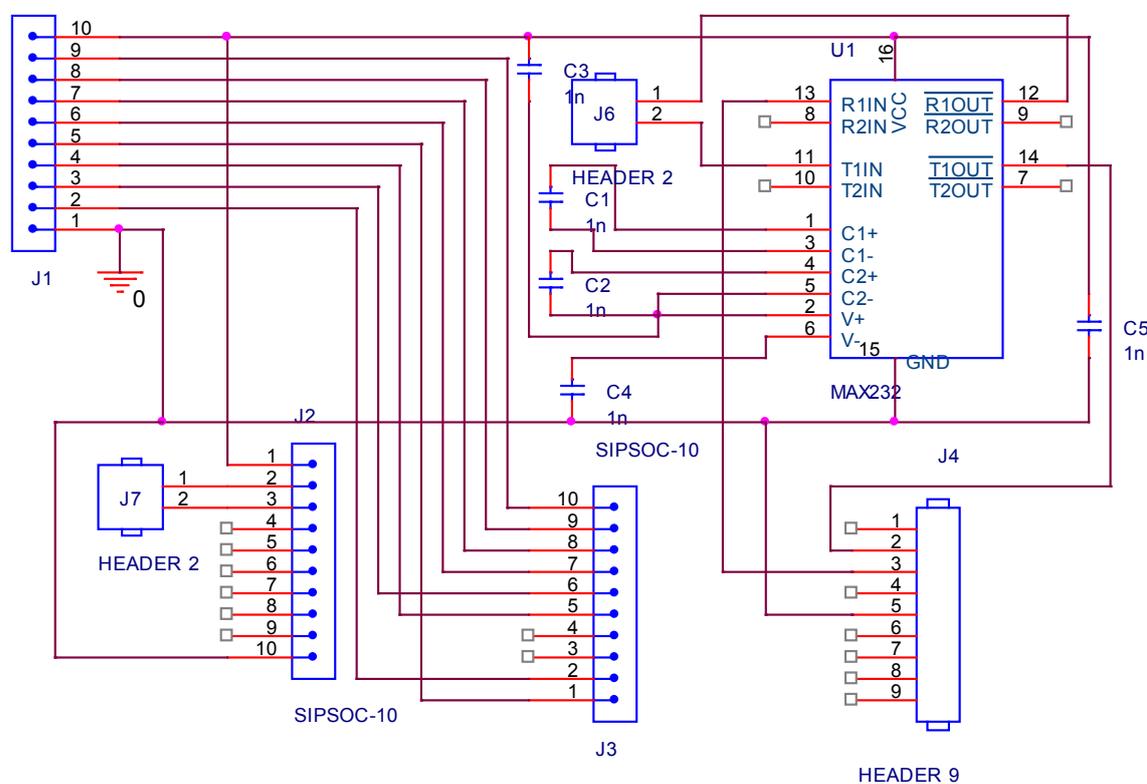


Figura 17. Esquemático Orcad

4.8 FUNCIONAMENTO DO SISTEMA

Para o funcionamento do sistema, tem-se, de um lado a impressora e o painel eletrônico, e de outro, os guichês de atendimento. Chegando um cliente na fila, o mesmo aperta o botão, acionando a impressão de uma senha, conforme Figura 18. Este cliente fica aguardando até que o painel eletrônico exiba sua senha e o guichê ao qual ele deve dirigir-se, vide Figura 19.



Figura 18. Impressora Bematech MP-20 MI



Figura 19. Display de Leds

4.9 DISCUSSÃO FINAL

O sistema de atendimento da secretaria da Faculdade de Engenharia da PUCRS não possui grandes complexidades, pois não tem um fluxo intenso de pessoas na maior parte do tempo, por isso foi dispensado um estudo mais aprofundado de reorganização do sistema físico. Porém, se aprovada a execução do projeto, deve-se fazer este estudo, a fim de analisar seu perfil, verificando as possibilidades de inclusão ou exclusão de terminais de atendimento.

De um modo geral todo o sistema funcionou conforme objetivos iniciais, exceto alguns detalhes descritos no começo do trabalho como: reset de senha ao final do dia e caixas

com diferentes tipos de atendimentos, o que ainda pode ser complementado de forma simples, com dedicação de mais um período de tempo para o trabalho.

5 CONCLUSÕES

O objetivo principal deste trabalho era implementar um sistema capaz de gerenciar filas de atendimento de clientes. Por ser um trabalho realizado dentro da PUCRS, deu-se preferência em aplicá-lo na própria universidade, mais especificamente na secretaria da Faculdade de Engenharia, local onde muito se convive.

Utilizando o Kit de Laboratório de Processadores desenvolvido para a cadeira de Laboratório de Processadores, que tem como microprocessador o MSP430F2619, foi possível desenvolver dois tipos de programas: um controlando a impressão de tickets com as senhas de atendimento e a recepção de instruções de guichês livres, e o outro controlando a liberação dos guichês e transmitindo tal informação a outro módulo.

O módulo ZigBee (XBee Pro) foi muito útil, pois permitiu que os equipamentos mantivessem uma certa distância entre si, sem a necessidade de fios interligando os mesmos, seguindo um protocolo definido especificamente para esta aplicação. O desenvolvimento do protocolo seguiu o esquema do protocolo poll/select, que é um protocolo para comunicação assíncrona muito utilizado.

Através deste trabalho foi possível reunir diferentes conceitos estudados durante o curso, permitindo uma aplicação prática real dos mesmos. Este trabalho possibilita que se desenvolva um projeto do começo ao fim, integrando tudo que se aprende, e também se deparando com todas as dificuldades que surgem ao longo dele, levando o aluno mais perto do que vai enfrentar fora da universidade, no seu dia-a-dia de trabalho.

Com o desenvolvimento do trabalho percebeu-se uma diferença entre este e os outros trabalhos aplicados pelos professores ao longo dos anos. Neste trabalho não havia um roteiro certo a cumprir, determinado pelo professor, mas sim o aluno que devia buscar os seus objetivos e traçar seu rumo, claro, com apoio do orientador.

Por fim, compreende-se que este trabalho deve ser continuado, visto que este assunto é muito amplo e pode ser aplicado em diversas situações. Sugerem-se dois seguimentos principais: um mais focado na parte teórica, e no uso das teorias de probabilidade, e o outro na prática, do desenvolvimento do software.

Para a questão teórica, pede-se aplicar as equações matemáticas destacadas neste artigo, o que exige um estudo muito mais aprofundado da situação do ambiente de execução

(como neste caso a secretaria), conhecendo os diferentes padrões de chegada dos clientes na fila, seus tempos mínimos e máximos de espera e atendimento, quantidade disponível de terminais nos diferentes horários, dentre outros.

Já na parte prática, pede-se o desenvolvimento de um sistema mais robusto, com filas de atendimento múltiplas, onde cada guichê atenda diferentes tipos de situações, escolha prévia do cliente pelo serviço requerido, facilitando o encaminhamento do cliente para o servidor que melhor lhe atenda, e também a aplicação de um sistema sonoro a cada chamada do próximo cliente.

6 REFERÊNCIAS

- [1] PRADO, Darci. Teoria das Filas e Simulação. 3ª edição, São Paulo, 2004.
- [2] M. Ajmone-Marsan, G. Balbo, and G. Conte. **Performance Models of Multiprocessor Systems**. The MIT Press, Cambridge, USA, 1986.
- [3] **Teoria de Filas de Espera**. Disponível em <http://www.dei.isep.ipp.pt/~andre/documentos/tfe.html>. Acessado dia 30/08/2010.
- [4] **NETSGAP. Soluções para controle de filas**. Disponível em <http://www.fcatec.com/netsgap>. Acessado em 01/09/2010.
- [5] **SISTEMATH – Sistemas e Soluções**. Disponível em <http://www.sistemath.com.br/rientadoresSenha.html>. Acessado em 01/09/2010.
- [6] Costa, Luciano Cajado. **Teoria das filas**. Apostila desenvolvida para disciplina de Teoria de filas e simulação. Disponível em http://www.deinf.ufma.br/~mario/grad/filas/TeoriaFilas_Cajado.pdf. Acessado em 01/09/2010.
- [7] **Datasheet MSP430F2619TPM**. Disponível em <http://focus.ti.com/lit/ds/symlink/msp430f2619.pdf>. Acessado em 13/10/2010.
- [8] **Manual da Impressora MP-20 MI**. Disponível em <http://www.bematech.com.br>. Acessado em 15/10/2010.
- [9] **Xbee Pro**. Disponível em <http://www.rogercom.com>. Acessado em 30/10/2010.