

**ERIC MAYOR SILVEIRA**

**SISTEMA DE AQUISIÇÃO DE DADOS  
PARA PALMILHA INSTRUMENTADA  
COM TRANSMISSÃO SEM FIO E  
SOFTWARE DE VISUALIZAÇÃO**

Trabalho de Conclusão de Curso  
apresentado à Escola de  
Engenharia de São Carlos, da  
Universidade de São Paulo

Curso de Engenharia Elétrica com  
ênfase em Eletrônica

ORIENTADOR: Prof. Dr. Alberto Cliquet Junior

São Carlos  
2007

## Resumo

Este trabalho de graduação tem por objetivo desenvolver um sistema capaz de coletar dados de baixa freqüência fornecidos por sensores diversos. Os dados, após serem digitalizados e multiplexados, são transmitidos por radiofreqüência até um receptor conectado à porta serial de um computador. Através do software LabVIEW, foi desenvolvido um instrumento virtual responsável por coletar os dados da porta serial, demultiplexá-los e exibir, através de gráficos, o comportamento de cada sensor simultaneamente e em tempo real. Além disso, os dados podem ser armazenados em arquivos para posterior análise.

Para exemplificar uma possível aplicação do sistema, ele foi adaptado para coletar dados de sensores de força tipo FSR (*force sensing resistors*) presentes em uma palmilha – a palmilha instrumentada em si não faz parte do escopo do projeto, embora, para que o exemplo fosse tratado, as características do FSR tenham sido estudadas e alguns testes feitos.

A principal motivação deste projeto é produzir um sistema de baixíssimo custo capaz de prover maior liberdade de movimento e de distância que pode ser percorrida em um teste qualquer com sensores remotos.

Para adaptar o sistema à aplicação desejada basta modificar o painel frontal da interface em LabVIEW e condicionar a saída dos sensores para a faixa de 0 a 5 V.

Palavras chave: Aquisição de dados, transmissão sem fio, baropodômetro, palmilha instrumentada.

## Abstract

This graduation project's objective is to develop a system capable of acquiring low frequency data from several kinds of sensors. After being digitalized and multiplexed, the data is transmitted using a radio transmitter to a receptor connected in a computer's serial port. Through the software LabVIEW, a virtual instrument was developed to collect this data from the serial port, demultiplex and show, using graphics, each sensor's behavior concurrently and in real time. Besides, the data can be stored in files for further analysis.

To exemplify a possible application of the system, it was adapted to collect data from Force Sensing Resistors (FSR) present on an insole – the instrumented insole itself is not included in the project's scope, although, to develop this example, FSR's characteristics were studied and some tests made.

This project's main motivation is to produce a low-cost system capable of improving mobility freedom and the distance that can be performed in any given test with remote sensors.

In order to adapt the system to the desired application, it's just necessary to modify the front panel of the LabVIEW interface and to have the sensor out properly conditioned to range of 0 to 5 V.

**Keywords:** Data acquiring, wireless transmission, baropodometer, instrumented insole.

# Sumário

Sumário.....	1
Índice de Figuras.....	4
Índice de Tabelas .....	5
1    Introdução .....	6
2    Contexto da Pesquisa .....	8
3    A Primeira Etapa .....	9
4    Estrutura da Segunda Etapa .....	11
5    Sensores.....	12
5.1    Características .....	12
5.2    Implementação .....	14
6    Transmissor .....	16
6.1    Hardware .....	16
6.2    Software .....	17
6.2.1    Controle do Conversor .....	18
6.2.2    Controle da Comunicação Serial .....	19
7    Receptor .....	21
8    Interface com o Usuário em LabVIEW.....	22
9    Resultados .....	23
10    Conclusões .....	24
10.1    Sugestões para implementações futuras .....	24
Referência Bibliográfica.....	25
Apêndice I – Diagramas dos VI's da Primeira Etapa.....	27
Apêndice II – Programa do Microcontrolador .....	31
Apêndice III – Mudanças nos VI's na Segunda Etapa .....	34

# Índice de Figuras

Figura 1 - Painel Frontal - Interface com o usuário.....	9
Figura 2 - Representação simplificada do sistema .....	11
Figura 3 - Estrutura do FSR, retirado de SENSOR WIKI, (2006). .....	12
Figura 4 – Resposta do FSR – Massa x Resistência .....	14
Figura 5 - Condicionamento de sinal dos sensores .....	15
Figura 6 - Representação esquemática do módulo transmissor .....	16
Figura 7 - Módulo transmissor .....	17
Figura 8 - Seqüência de operação do conversor A/D, retirado de TEXAS INSTRUMENTS (1988) .....	18
Figura 9 - Representação esquemática do módulo receptor .....	21
Figura 10 - Módulo receptor.....	21
Figura 11 - Dados do pé esquerdo.....	23
Figura 12 - Dados do pé direito.....	23

## **Índice de Tabelas**

Tabela 1 – Levantamento da resposta do FSR .....	13
--	----

# 1 Introdução

A tecnologia de transmissão de dados sem fio vem ganhando importância fundamental em aplicações pequenas e de uso local. Esta tendência, já extremamente difundida, facilita muito o uso de tais aplicações trazendo liberdade de movimento.

Há muitas aplicações em Engenharia de Reabilitação nas quais se faz necessário o monitoramento dos esforços exercidos, ou mesmo da posição em que se encontram os membros superiores e inferiores de pacientes com deficiência de mobilidade (FARIA & CARVALHO, 2002). Um sistema responsável por esta tarefa deve coletar os dados de forma que não atrapalhe ou limite o movimento natural para que os dados representem de maneira mais fiel a situação a ser estudada.

Este projeto possuiu duas etapas de implementação. A primeira, iniciada a partir de um trabalho realizado na disciplina SEL396 - Introdução à Instrumentação Eletrônica Biomédica, e a segunda etapa com o objetivo de promover melhorias na funcionalidade do sistema. O trabalho em questão tinha a proposta de coletar e apresentar dados de uma palmilha formada por sensores FSR, daí surgiu o exemplo de aplicação utilizado neste projeto. Foi utilizado para a coleta de dados o hardware da National Instruments, PCI 6024E, juntamente com o software LabVIEW.

Este hardware é extremamente versátil e poderoso, mas, devido ao seu custo, não é tão difundido quanto se gostaria para possibilitar o uso desta aplicação em qualquer computador.

Outra limitação da etapa inicial era a necessidade de cabos que ligassem o usuário da palmilha ao hardware de aquisição de dados, restringindo a liberdade de movimento a poucos metros em volta do computador utilizado.

A segunda etapa apresenta e implementa uma alternativa para os dois problemas apresentados. Os requisitos são a aquisição em 8 canais com sensores FSR e freqüência de amostragem adequada para a análise de eventos na marcha, eliminação da necessidade do hardware da National Instruments sem perda significativa de qualidade na aquisição de dados e, finalmente, transmissão sem fio dos dados entre o objeto de estudo onde estão os sensores e o computador.

O trabalho apresenta no Capítulo 2 alguns conceitos e uma revisão de outros trabalhos que serviram de base para o desenvolvimento do exemplo de aplicação escolhido. Após, há a descrição da primeira etapa de desenvolvimento com o hardware da National Instruments. O Capítulo 4 faz a descrição inicial da segunda

etapa, sendo que os elementos que a formam (sensores, transmissor, receptor, interface utilizando LabVIEW) são descritos nos capítulos subseqüentes.

## 2 Contexto da Pesquisa

Sensores de pressão vêm sendo usados na área da saúde em inúmeras aplicações. Muitas soluções comerciais existem para análise de marcha, posicionamento de pacientes para evitar ulcerações, análise da distribuição de pressão no encaixe entre próteses e o corpo, entre outras (TEKSCAN, 2007).

Com relação à análise de marcha, o monitoramento das pressões plantares pode ser utilizado para acompanhar a evolução de tratamentos e determinar sua eficácia, sejam com órteses, fisioterapia, cirurgia, etc. Outra utilidade seria o diagnóstico de anormalidades de postura e de pisada. Estas aplicações, no entanto, demandam grande número de pontos de monitoramento e boa precisão nas medidas para que sejam obtidos resultados satisfatórios.

Um tipo de sensor muito utilizado em trabalhos acadêmicos relacionados é o FSR. Ele tem sido comumente aplicado no controle de estimulação elétrica durante a marcha em pacientes com lesão medular (SMITH et al. 2002). WERTSCH et al. (1992) utilizaram o que parecem ser FSRs (sensor de polímero condutivo) em uma palmilha com um sistema portátil de monitoramento de pressões plantares.

### 3 A Primeira Etapa

Como mencionado anteriormente, o projeto foi desenvolvido inicialmente utilizando como meio de aquisição de dados o hardware PCI 6024E da National Instruments. Ele é uma placa que fica interna no computador instalada em um barramento PCI. Um cabo liga a placa a um painel externo de conectores que serve de interface para as aplicações. Ele possui 16 entradas analógicas de 200 kS/s e 12 bits, 8 portas de entrada e saída digitais, 2 saídas analógicas de 12 bits e 2 contadores de 24 bits. Na aplicação foram utilizadas oito de suas entradas analógicas e o hardware atendeu perfeitamente à necessidade, além de ser muito fácil de operar.

O tratamento dos dados e a interface com o usuário foram feitos através do software LabVIEW. Programas desenvolvidos nesta linguagem são chamados de Instrumentos Virtuais, ou simplesmente VI's (do inglês Virtual Instrument). Um VI é dividido em duas partes: o Diagrama de Blocos (programa propriamente dito) e o Painel Frontal (interface gráfica com o usuário). O VI desenvolvido para este projeto era responsável por ler os dados, plotá-los, registrá-los em arquivo e reproduzir em tempo real nos gráficos uma aquisição anteriormente gravada.

Na Figura 1 vemos o painel frontal do VI.

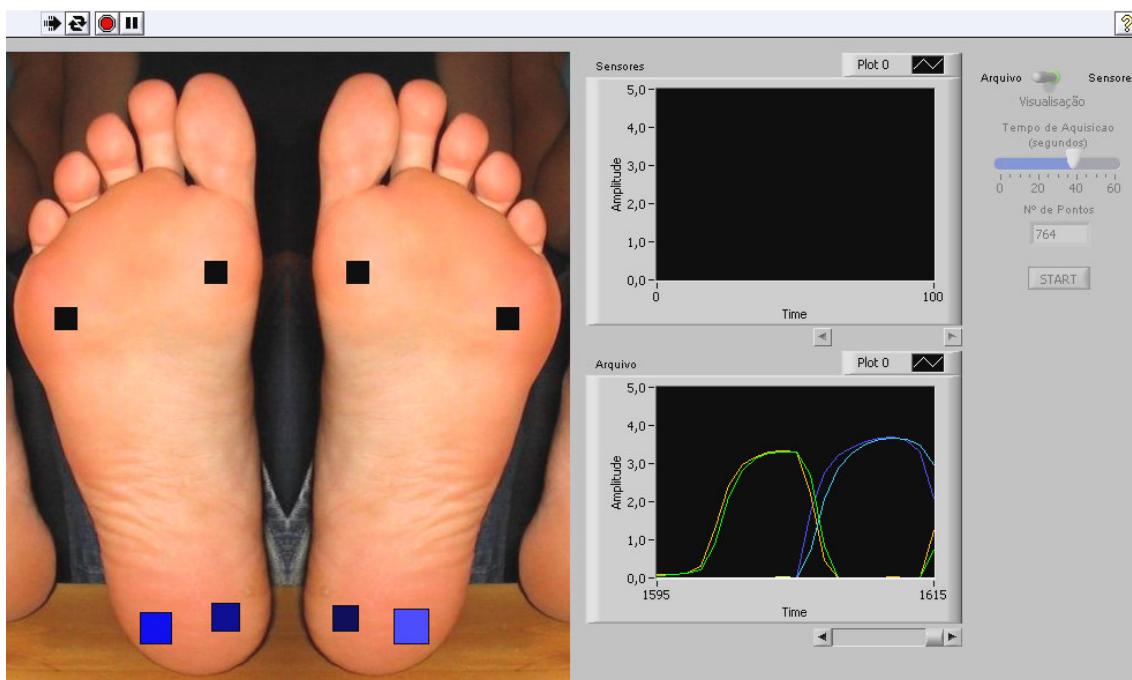


Figura 1 - Painel Frontal - Interface com o usuário

O painel possui à direita alguns controles. São eles: uma chave para mudar

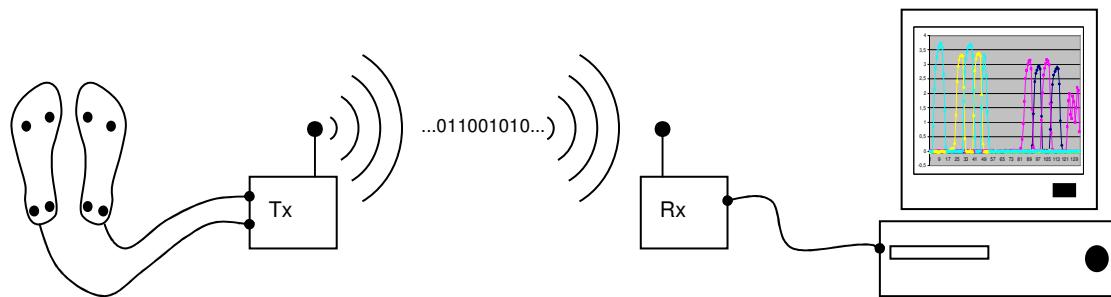
entre os modos “aquisição” e “leitura de arquivo”; um botão deslizante para selecionar o tempo de duração da aquisição; um botão que dá início à aquisição ou à leitura.

Os dados do sensor ou do arquivo são plotados nos gráficos em tempo real, enquanto isso, a figura à esquerda mostra, na posição dos sensores em cada pé, quadrados que variam seu tamanho e cor de acordo com a pressão em cada sensor.

As figuras que mostram os diagramas de bloco do VI principal, e ainda de dois sub-VI's que foram criados para agrupar partes repetitivas do diagrama, podem ser vistas no Apêndice I.

## 4 Estrutura da Segunda Etapa

O projeto da segunda etapa é formado por dois módulos de hardware e um software para PC. Um dos módulos digitaliza e transmite os sinais de 8 sensores. O outro módulo recebe e envia os sinais para a porta serial do PC. Desta forma eliminam-se a necessidade da placa de aquisição de dados e da ligação direta entre os sensores e o computador através de fios. Este conjunto é bem compacto e tem uma representação simplificada na Figura 2.



**Figura 2 - Representação simplificada do sistema**

O primeiro critério na escolha dos componentes foi a sua disponibilidade no laboratório. A lista de componentes que atende aos requisitos descritos na Introdução deste trabalho é vista abaixo:

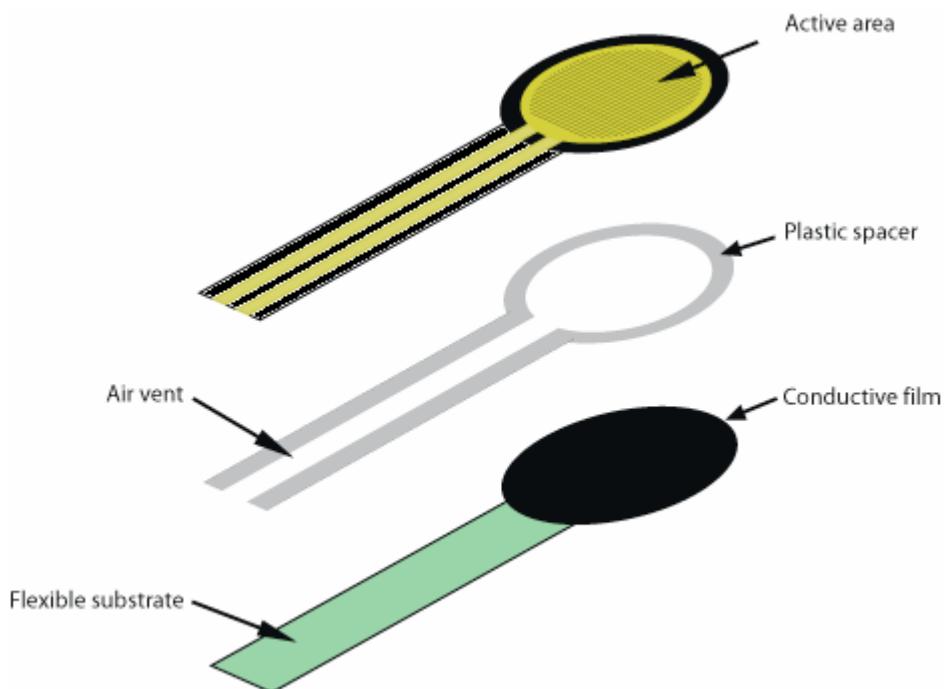
- 8 sensores FSR Interlink Electronics nº 402;
- 8 resistores de  $4,7\text{ k}\Omega$ ;
- Regulador de tensão LM7805;
- Digitalizador ADC0808;
- Microcontrolador AT89C52;
- Cristal de 16 MHz;
- 2 capacitores de  $33\text{ pF}$ ;
- Transmissor de rádio Telecontrolli modelo RT4 de 433,92 MHz;
- Bateria de 9 V;
- Receptor de rádio Telecontrolli modelo RR3 de 433,92 MHz;
- Driver/Receiver de RS-232, MAX-233;
- Cabo e conector DB9 fêmea.

## 5 Sensores

O FSR (force sensing resistor) é um elemento resistivo cuja resistência varia de acordo com a força aplicada sobre sua superfície. Como a sua saída depende da área de sua superfície em que a força é aplicada, seria mais correto chamá-lo de sensor de pressão. Quanto maior a pressão aplicada, menor a resistência entre os terminais do FSR (INTERLINK ELECTRONICS, 2002). Com isto em mente, conclui-se que quanto menor o atuador, menor a força necessária para saturar sua saída.

### 5.1 Características

O FSR é formado por um filme de polímero condutivo e eletrodos, sendo estas duas partes separadas por um espaçador. Sua estrutura é mostrada na Figura 3.



**Figura 3 - Estrutura do FSR, retirado de SENSOR WIKI, (2006).**

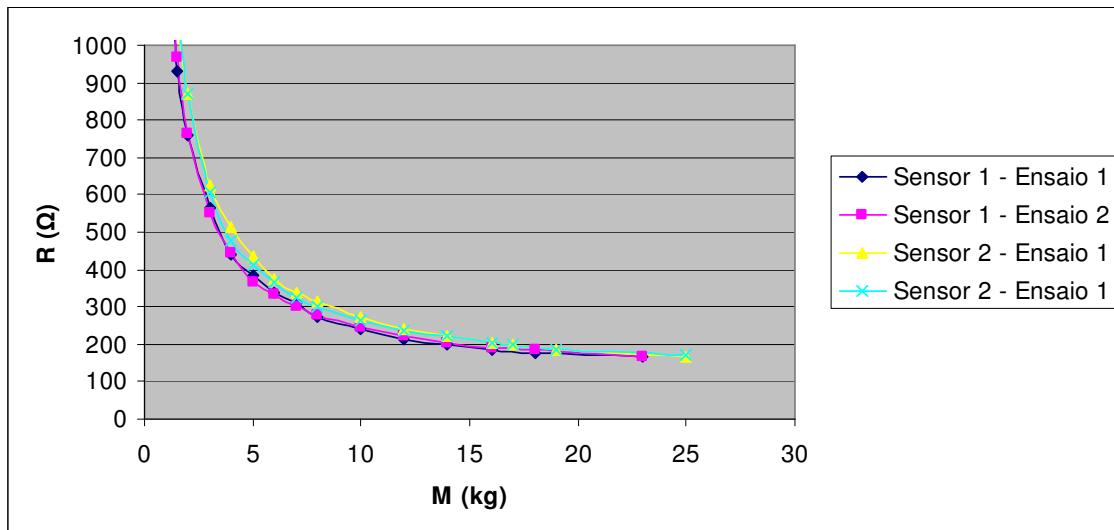
Com nenhuma força aplicada sobre ele, seu comportamento é de circuito aberto. Na medida em que sua superfície é pressionada e os eletrodos da área ativa entram em contato com o material condutivo, a resistência entre os dois terminais diminui até o ponto em que, mesmo aumentando a pressão aplicada, a resistência não diminui ou diminui muito pouco. Neste ponto o FSR está saturado. INTERLINK ELECTRONICS (2002) relata que a pressão de saturação de um FSR típico é da ordem de 100 a 200 psi. Considerando que a área ativa do sensor utilizado é de 1,27 cm<sup>2</sup>, isto teoricamente daria um máximo de 17,7 Kg até a saturação.

A resposta do sensor não é linear e Zehr et al. (1995) atribuem à característica de não-linearidade dos FSRs a dificuldade de se obter um método válido, confiável e calibrável de se registrar a variação de força com eles. Também sugerem um método de linearização. Porém, há ainda outras características dos FSRs que sugerem uma maior adequação de seu uso para analisar aspectos qualitativos, e não quantitativos, da variação de força. INTERLINK ELECTRONICS (2002) menciona que FSRs não são adequados para medições de precisão.

A Figura 4 mostra a resposta do FSR Interlink Electronics nº 402. Foi utilizado um dinamômetro da marca SHIMPO, modelo FGV300H, com um atuador de ponta de borracha do mesmo tamanho da área ativa do sensor. Dois sensores foram submetidos ao ensaio, que foi feito duas vezes em cada sensor. Os dados também estão na Tabela 1.

**Tabela 1 – Levantamento da resposta do FSR**

Sensor 1			Sensor 2		
Ensaio 1	Ensaio 2		Ensaio 1	Ensaio 2	
M (kg)	R (Ω)	R (Ω)	M (kg)	R (Ω)	R (Ω)
0,5	2177	2620	0,5	3600	4000
1,0	1311	1320	1,0	1653	2399
1,5	930	966	1,5	1098	1150
2,0	757	765	2,0	870	870
3,0	563	550	3,0	625	607
4,0	438	445	4,0	515	475
5,0	386	367	5,0	434	413
6,0	336	333	6,0	377	366
7,0	308	300	7,0	340	326
8,0	272	278	8,0	314	302
10,0	241	244	10,0	274	263
12,0	211	220	12,0	239	238
14,0	199	203	14,0	222	220
16,0	187	191	16,0	205	205
18,0	177	184	19,0	187	183
23,0	166	168	25,0	168	171

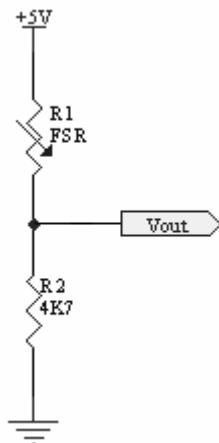


**Figura 4 – Resposta do FSR – Massa x Resistência**

Comparando a resposta de um mesmo sensor em diferentes momentos, vemos uma diferença média de aproximadamente 10 Ohms. Já, se compararmos a resposta de diferentes sensores, a diferença média é de mais de 40 Ohms. Isto demonstra que os processos de fabricação do sensor podem influenciar na repetitibilidade das respostas entre os sensores. Em geral esta tolerância está na faixa de  $\pm 15\%$  a  $\pm 25\%$  (INTERLINK ELECTRONICS, 2002). O tipo de atuador também influencia na resposta, por exemplo, num caso extremo se for utilizado um atuador plano e rígido, com área maior que a do sensor, o atuador ficará apoiado no espaçador do sensor. Isto impedirá que a área ativa toque o filme condutivo. Portanto, nestas condições, dificilmente o FSR sairá da condição de circuito aberto. Neste caso é recomendável uma interface não rígida, como no ensaio descrito acima.

## 5.2 Implementação

Quatro FSRs serão usados em cada pé. Devido ao hardware utilizado, é necessário extrair um sinal de tensão de 0 a 5 V. O condicionamento feito no FSR é simples. Apenas limita a corrente e permite uma sensibilidade maior para uma melhor visualização. A Figura 5 ilustra o divisor de tensão utilizado.



**Figura 5 - Condicionamento de sinal dos sensores**

Cada FSR foi colocado em série com um resistor de 4,7 kΩ. Considerando a resistência mínima que o FSR alcançou, o valor do resistor foi escolhido de forma que a tensão a ser lida não ficasse muito abaixo de 5 V e a corrente máxima no sensor por volta de 1 mA.

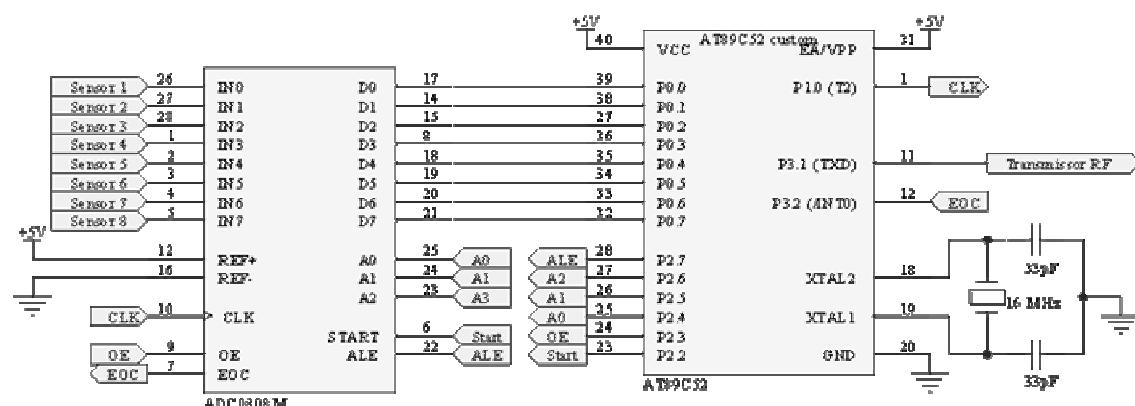
## 6 Transmissor

Este módulo do hardware é responsável por digitalizar os dados de 8 sensores, multiplexá-los e transmiti-los por rádio freqüência. Como este módulo será levado pelo usuário do sistema, ele é portátil e alimentado por bateria.

Os principais componentes deste módulo são o conversor A/D, o microcontrolador e o transmissor de rádio. Seguindo a premissa de utilizar preferencialmente componentes presentes no laboratório, optou-se por um digitalizador separado ao invés de um microcontrolador que já possuísse tal função.

## 6.1 *Hardware*

A Figura 6 mostra o esquemático do circuito deste módulo.



**Figura 6 - Representação esquemática do módulo transmissor**

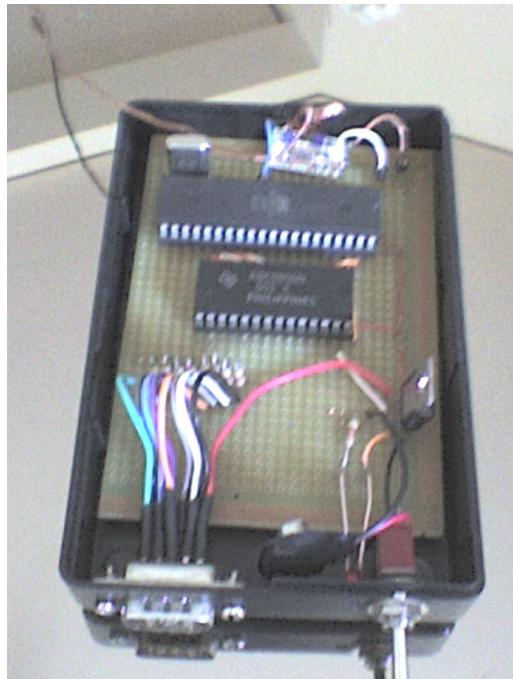
O conversor ADC0808 utilizado possui exatamente 8 canais, saída paralela – que facilita na transmissão dos dados para o microcontrolador –, baixo tempo de conversão – com o clock utilizado de 1 MHz, o tempo de conversão é de no máximo 74  $\mu$ s –, 8 bits de resolução – adequada para a faixa de 0 a 5 V em que se encontra os sinais, dando uma resolução de aproximadamente 20 mV – e baixo consumo.

As características interessantes do microcontrolador AT89c52 para esta aplicação são as 4 portas de entrada e saída, 3 timers de 16 bits e UART para a comunicação serial. A porta P0 foi usada para receber os dados do conversor. Da porta P1 só foi utilizado o bit 0, configurado como uma saída de clock através do Timer 2. Este sinal de clock é utilizado no conversor A/D. A porta P2 utiliza 6 bits no controle do conversor. Finalmente, da porta P3 foram utilizados 2 bits, um deles na recepção

do sinal EOC do conversor – que avisa quando a conversão terminou – e o outro bit é a saída serial, que tem sua temporização controlada pelo Timer 1.

A saída serial do microcontrolador é ligada diretamente ao transmissor de rádio. A taxa máxima de dados que o transmissor suporta é 4 kHz e sua freqüência de transmissão é de 433,92 MHz.

A Figura 7 mostra o módulo já montado e pronto para utilizar.



**Figura 7 - Módulo transmissor**

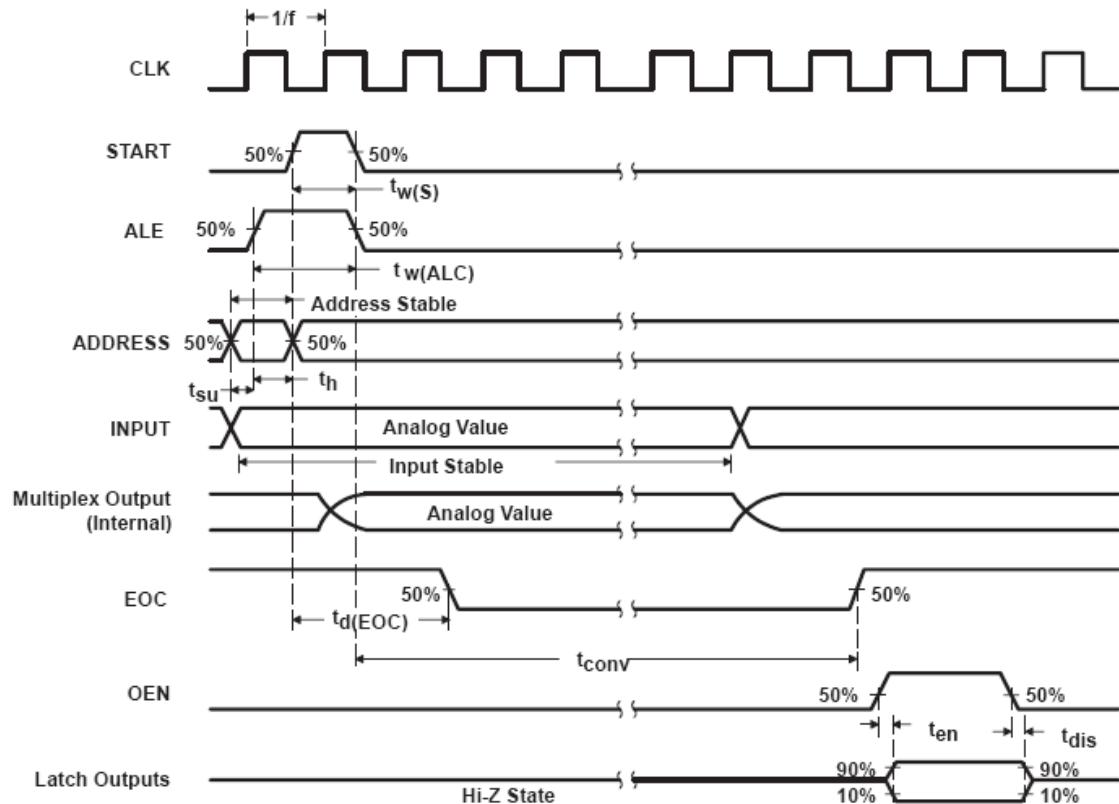
Como comentado anteriormente, a alimentação deste módulo é por uma bateria de 9 V sendo que um regulador de tensão é utilizado para conseguir os 5 V necessários para alimentar os componentes. Um conector DB9 foi utilizado para receber os dados dos sensores. Um de seus pinos fornece 5 V e os outros oito pinos recebem os dados dos sensores. Os resistores em série, utilizados no condicionamento mostrado na Figura 5, podem ser vistos nos terminais do conector DB9. Para que este sistema de aquisição de dados seja usado em outra aplicação, este condicionamento deve ser alterado para atender aos requisitos do tipo de sensor utilizado.

## **6.2 Software**

O software que roda no microcontrolador (ver Apêndice II) é responsável por controlar o conversor A/D e por enviar, através de comunicação serial, os dados recebidos dele.

### 6.2.1 Controle do Conversor

A folha de dados do conversor ADC0808 apresenta a seguinte característica de operação mostrada na Figura 8.



**Figura 8 - Seqüência de operação do conversor A/D, retirado de TEXAS INSTRUMENTS (1988)**

O clock máximo que o componente suporta é de 1,28 MHz. A freqüência escolhida para esta aplicação foi de 1 MHz por simplicidade. Este clock é gerado pelo Timer 2 do microcontrolador. Para que esse timer funcionasse como uma saída de clock programável, os registradores de funções especiais que controlam o timer receberam os seguintes valores (em hexadecimal): T2MOD = 02 e T2CON = 04.

A freqüência é determinada pela equação (1)

$$f_{clock-out} = \frac{f_{oscilador}}{4 \times [65536 - (RCAP2H, RCAP2L)]} \quad (1)$$

Como a freqüência do oscilador é de 16 MHz, para obter um clock de 1 MHz, o denominador em (1) deve ser 16. Ou seja, os registradores de 8 bits devem juntos formar o número 65532 em decimal. Portanto, em hexadecimal temos, RCAP2H = FF e RCAP2L = FC.

A seqüência de operações do conversor, vista na figura anterior, é implementada no programa pela função “converte\_sensor”. Esta função tem como parâmetro de entrada o número da entrada que deve ser convertida, e retorna um byte com o valor de tensão já digitalizado.

### 6.2.2 Controle da Comunicação Serial

A velocidade da comunicação serial foi escolhida levando-se em conta a taxa de dados que o receptor de rádio suporta. Nominalmente, são apenas 2 kHz, mas, experimentalmente, verificou-se que a taxa de dados excede 3 kHz.

Além disso, a velocidade deve ser suficiente para garantir uma visualização satisfatória dos dados. Segundo ZHU et al. (1991), ao monitorar eventos de marcha, aquisições a 20 amostras por segundo são suficientes, já que não mostram sinais muito diferentes daquelas feitas a 200 amostras por segundo.

O modo de operação da serial é controlado pelo registrador SCON, que recebeu o valor de 40 hexadecimal. Este é o Modo 1 de operação no qual cada transmissão tem 10 bits: 1 start bit, 8 bits de dados e 1 stop bit.

O Baud Rate da serial é determinado pelo Timer 1 e obedece à equação (2).

$$BaudRate = \frac{2^{SMOD}}{32} \cdot \frac{f_{oscilador}}{12(256 - TH1)} \quad (2)$$

Os registradores do Timer 1 receberam os seguintes valores em hexadecimal: TH1 = F2 e TL1 = F2. O registrador SMOD da serial recebeu Zero. Desta forma, temos Baud Rate = 2976 Bds.

O quadro a ser transmitido é formado por 9 bytes. Oito deles contêm os dados de cada canal. O nono byte é um valor fixo (foi escolhido FEh pois, da forma como foram condicionados os sensores, não é possível que sua leitura alcance este valor). Este byte é usado no LabVIEW para determinar quando começa e termina cada quadro.

A transmissão dos dados dos sensores é seqüencial. Para determinar quanto tempo é gasto para transmitir um quadro com os dados dos 8 sensores e mais o byte de controle, foi utilizado o bit zero da porta P2. A cada transmissão de um quadro, o estado do bit era invertido. Portanto, a freqüência de amostragem dos sensores é o

dobro da freqüência deste bit de teste. Com o Baud Rate de 2976 Bds, a freqüência de amostragem dos sensores foi de 33,33 Hz, portanto, quase 70% acima do necessário para uma boa visualização dos dados.

## 7 Receptor

Este módulo é bem simples. Ele consiste de um driver/receiver MAX-233 fazendo a interface entre o receptor de rádio e a porta serial do computador. A Figura 9 ilustra o esquemático deste módulo.

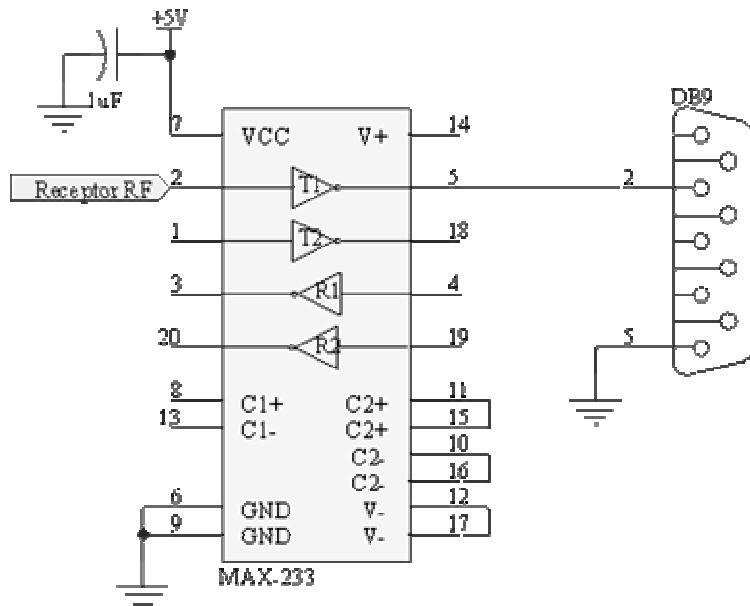


Figura 9 - Representação esquemática do módulo receptor

A Figura 10 mostra o módulo montado e pronto. É possível notar a escolha da alimentação por uma fonte regulada de 5 V, além do cabo para conectar na porta serial do PC e a antena feita com malha de cobre.

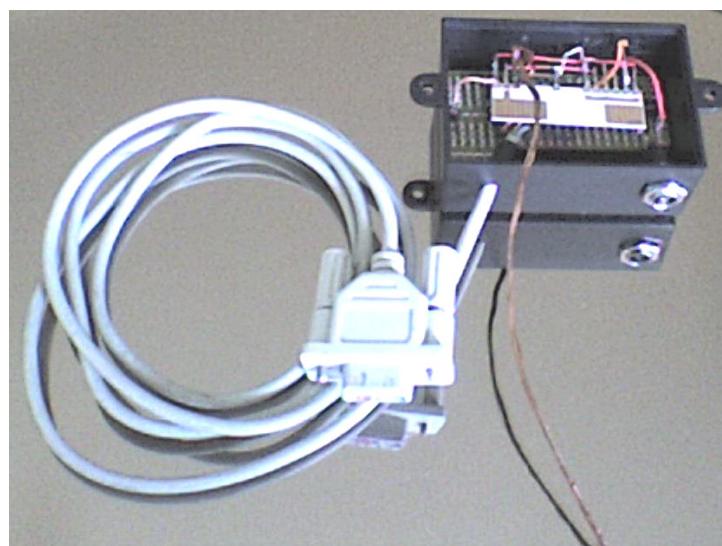


Figura 10 - Módulo receptor

## 8 Interface com o Usuário em LabVIEW

Nesta segunda etapa foram feitas adaptações no programa inicial, já que a entrada de dados agora é a porta serial. Ao painel frontal do VI foi acrescentado um controle para escolher o número da porta serial.

O diagrama de blocos principal também mudou pouco. A mudança maior foi no Sub-VI que tratava da leitura dos sensores, que agora trata da leitura da porta serial.

As mudanças feitas podem ser vistas nas figuras do Apêndice III.

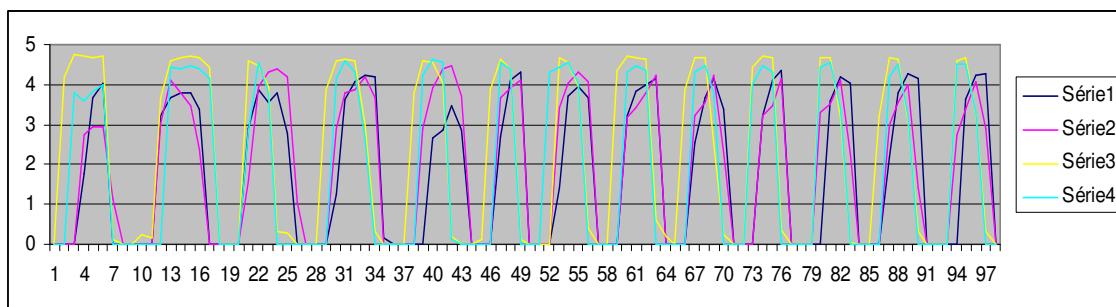
Um recurso interessante do LabVIEW que foi utilizado é a possibilidade de criar programas instaláveis. Basicamente estes programas incluem o VI desenvolvido em formato executável e os componentes do LabVIEW necessários para que o programa rode em qualquer computador mesmo sem o LabVIEW instalado.

## 9 Resultados

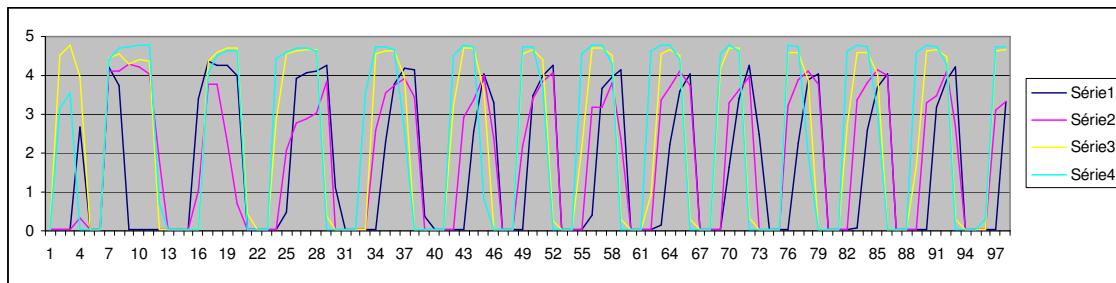
Após posicionar os sensores, ligá-los no módulo transmissor, ligar o receptor no PC, alimentar o conjunto e rodar o VI o sistema está pronto para fazer uma aquisição de dados. Basta determinar no painel frontal do VI quantos pontos de cada sensor vão ser coletados.

A distância de transmissão que o conjunto conseguiu alcançar foi de cerca de 20 m.

Nas Figuras 11 e 12 vemos o resultado de uma aquisição:



**Figura 11 - Dados do pé esquerdo**



**Figura 12 - Dados do pé direito**

Tendo em vista o posicionamento de sensores como visto na Figura 1, dos resultados acima é possível ver claramente o calcanhar tocando o chão primeiro e com mais força (os gráficos mostram a tensão lida ao longo dos pontos de coletados). Comparando os dois pés também é possível ver que eles tocam o chão em momentos alternados. Este tipo de informação sobre a posição em que se encontra o usuário, pode ser muito útil por exemplo no controle de estimulação elétrica durante a marcha.

# 10 Conclusões

Ao longo do texto os resultados foram sendo comentados, mas um resumo será apresentado.

O FSR é um sensor muito prático de trabalhar, não exigindo condicionamentos de sinal complicados. Sua aplicação, porém, não é para medidas precisas em um dado ponto da sola do pé, e sim para, por exemplo, determinar o momento em que aquele ponto toca o chão, ou ainda, comparar se um ponto sofre mais pressão do que outro.

O par transmissor/receptor de rádio, transmitindo dados digitalmente com freqüência menor do que 4kHz, foi um limitante neste projeto, apesar disto, os resultados foram satisfatórios para o exemplo de aplicação escolhido, dada a baixa freqüência que se observa nos eventos de marcha.

Para evitar interferências durante a transmissão por radiofreqüência, geralmente são utilizados codificadores que asseguram que um dado receptor só receba dados do transmissor escolhido. Este elemento foi suprimido deste projeto pois ficaria impraticável a transmissão de dados no volume que era necessário, já que os codificadores precisam transmitir alguns bits também. Portanto este sistema é suscetível a interferências, apesar de isto não ter sido observado durante os testes.

A distância máxima alcançada na transmissão de dados foi de 20 m, o que é mais do suficiente num ambiente laboratorial, ou mesmo num uso doméstico.

A interface com o usuário em LabVIEW apresenta os dados de maneira intuitiva, armazenando para posterior análise em arquivos de texto. Sendo que tais arquivos podem ser reproduzidos pela própria interface ou então importados pelo software Microsoft Excel.

## 10.1 Sugestões para implementações futuras

Dada a simplicidade do projeto e as possibilidades de melhorias, seria uma boa opção para ser aplicado como projeto didático em laboratório.

Um novo par transmissor/receptor com maior taxa de dados poderia ser aliado a uma otimização do VI em LabVIEW de forma que a taxa de amostragem dos sensores fosse maior, isto aumentaria a gama de aplicações que podem se utilizar deste sistema como acessório.

## Referência Bibliográfica

FARIA, U. C.; CARVALHO, A. A. (2002). *Implementação de transdutores com extensômetros para monitorar forças exercidas pelos membros superiores e inferiores de pacientes*. Revista Brasileira de Engenharia Biomédica, v. 18, n. 3, p. 163-172, set/dez 2002.

INTERLINK ELECTRONICS (2002). *Force Sensing Resistor Integration Guide and Evaluation Parts Catalog*. Disponível em <<http://www.interlinkelectronics.com/library/media/papers/pdf/fsrguide.pdf>>, acessado em agosto de 2006.

SMITH, B. T.; COIRO, D. J.; FINSON, R.; BETZ, R. R.; MCCARTHY, J. (2002). *Evaluation of Force-Sensing Resistors for Gait Event Detection to Trigger Electrical Stimulation to Improve Walking in the Child With Cerebral Palsy*. IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 10, no. 1, March 2002.

SENSOR WIKI (2006). *Force-sensitive resistor (FSR)*. Disponível em <[http://www.sensorwiki.org/index.php/Force-sensitive\\_resistor\\_\(FSR\)](http://www.sensorwiki.org/index.php/Force-sensitive_resistor_(FSR))>, acessado em outubro de 2007.

TEKSCAN, INC. (2007). *Pressure & Force Sensing*. Disponível em <<http://www.tekscan.com/pdfs/Brochure-Medical.pdf>>, acessado em outubro de 2007.

TEXAS INSTRUMENTS (1988). *ADC0808M, CMOS analog-to-digital converter with 8-channel multiplexer*. Disponível em <<http://www.ti.com/lit/gpn/adc0808>>, acessado em janeiro de 2007.

WERTSCH, J. J.; WEBSTER, J. G.; TOMPKINS W. J. (1992). *A Portable Insole Plantar Pressure Measurement System*. Journal of Rehabilitation Research and Development, v. 29, n. 1, p. 13-18.

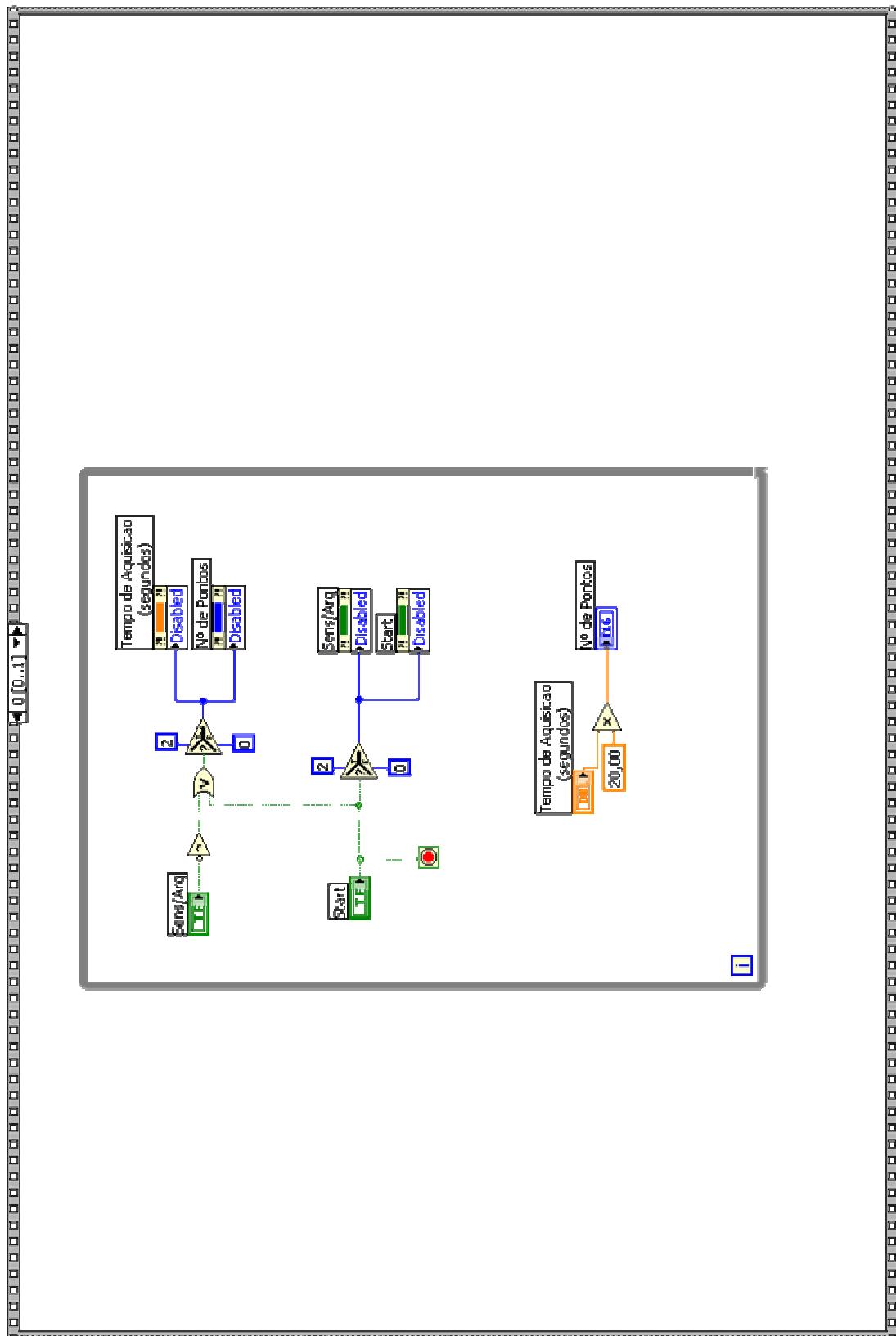
ZEHR, E. P.; STEIN, R. B.; KOMIYAMA, T.; KENWELL, Z. (1995). *Linearization of force sensing resistors (FSR's) for force measurement during gait*. Engineering in

Medicine and Biology Society, IEEE 17th Annual Conference, vol. 2, p 1571-1572, Sep. 1995.

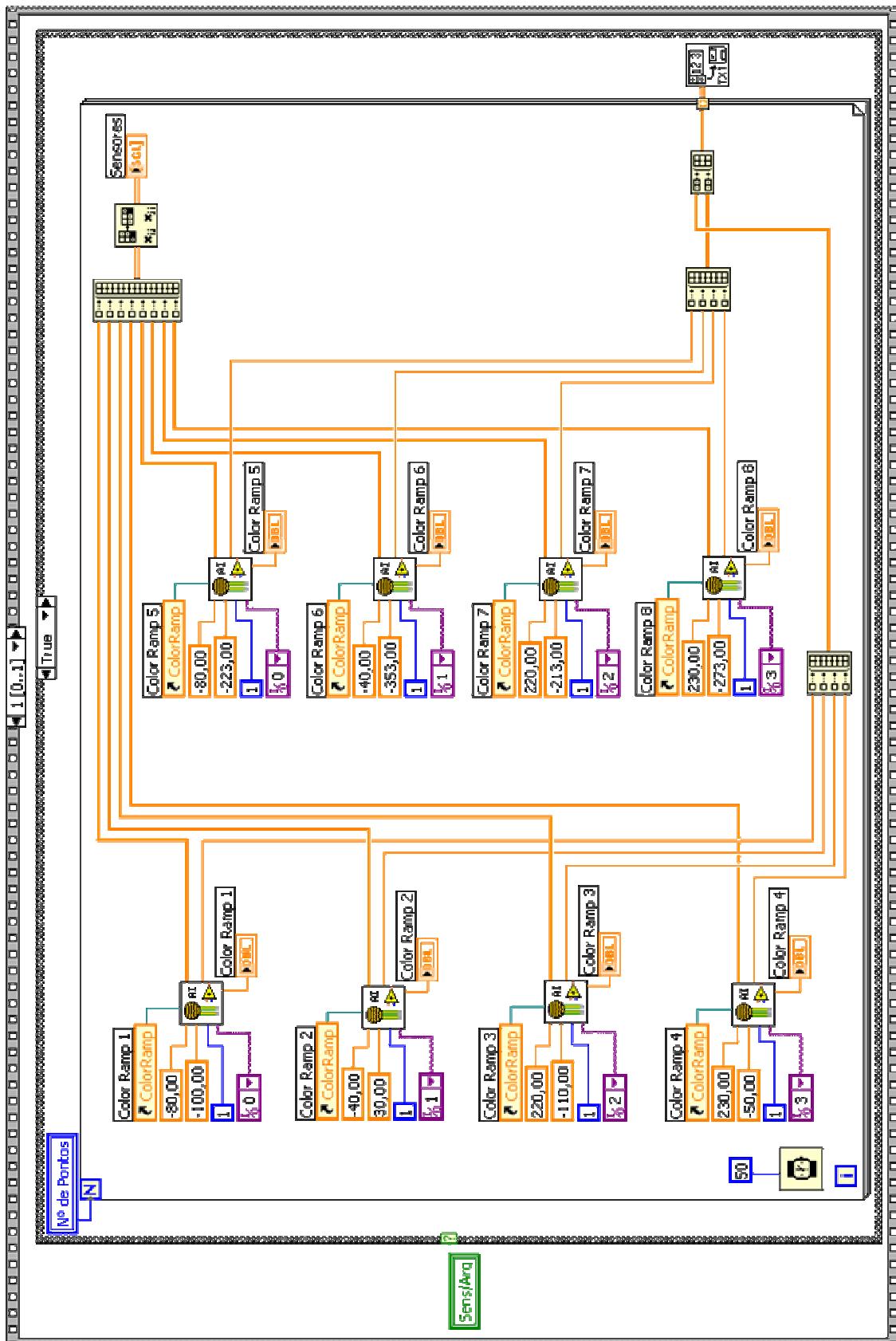
ZHU, H.; HARRIS, G. F.; WERTSCH, J. J.; TOMPKINS, W. J.; WEBSTER, J. G. (1991). *A Microprocessor-Based Data-Acquisition System for Measuring Plantar Pressures From Ambulatory Subjects*. IEEE Transactions on Biomedical Engineering, v. 38, n. 7, p. 710-714.

## Apêndice I – Diagramas dos VI's da Primeira Etapa

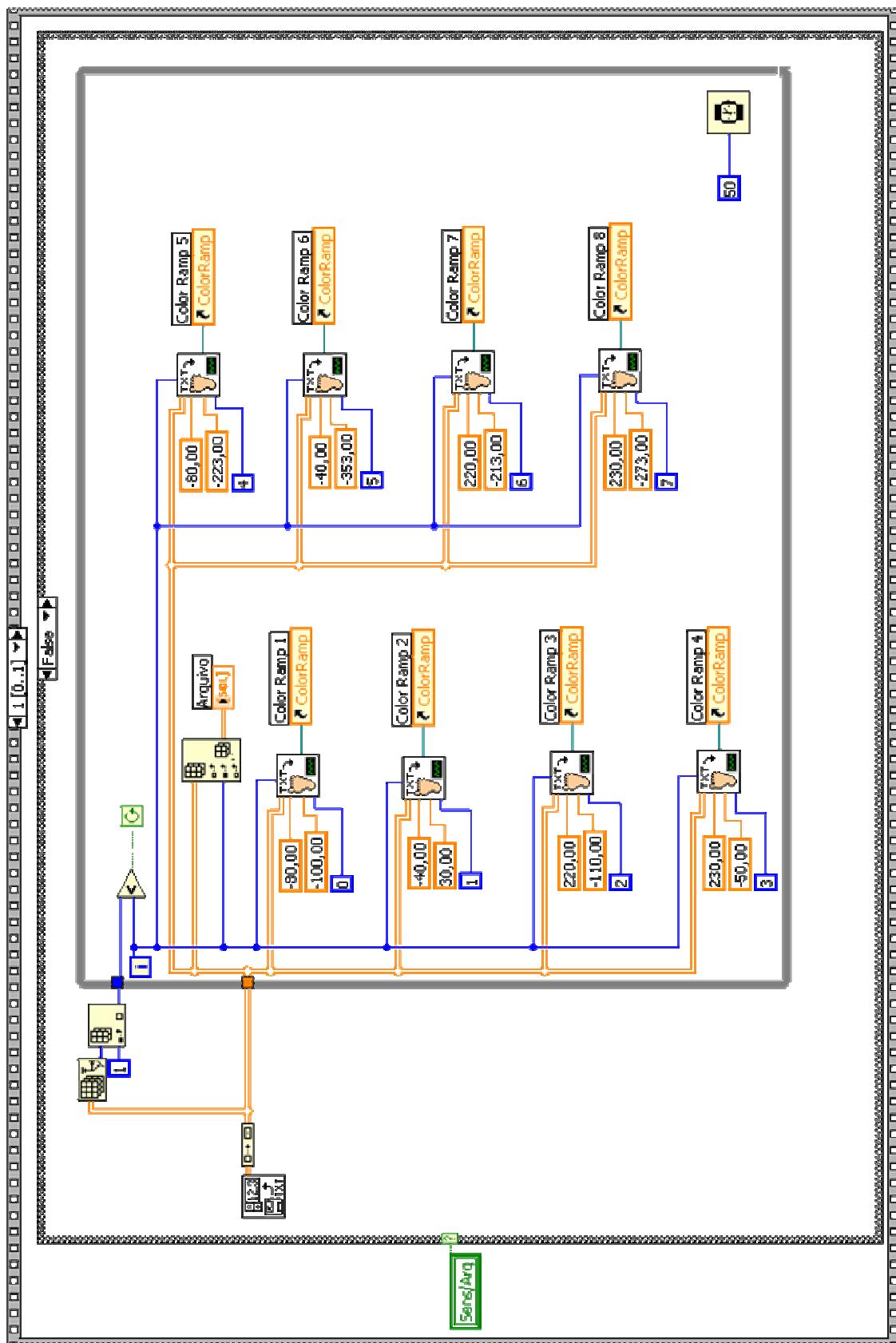
Rotina de controle dos botões do painel.



Rotina de controle da leitura dos sensores.



Rotina de controle da leitura de arquivo.



Sub-VI's: (a) Leitura de um sensor pela entrada analógica; (b) Tratamento dos dados de um sensor lidos de um arquivo.



Diagrama do sub-VI de leitura a partir da entrada analógica.

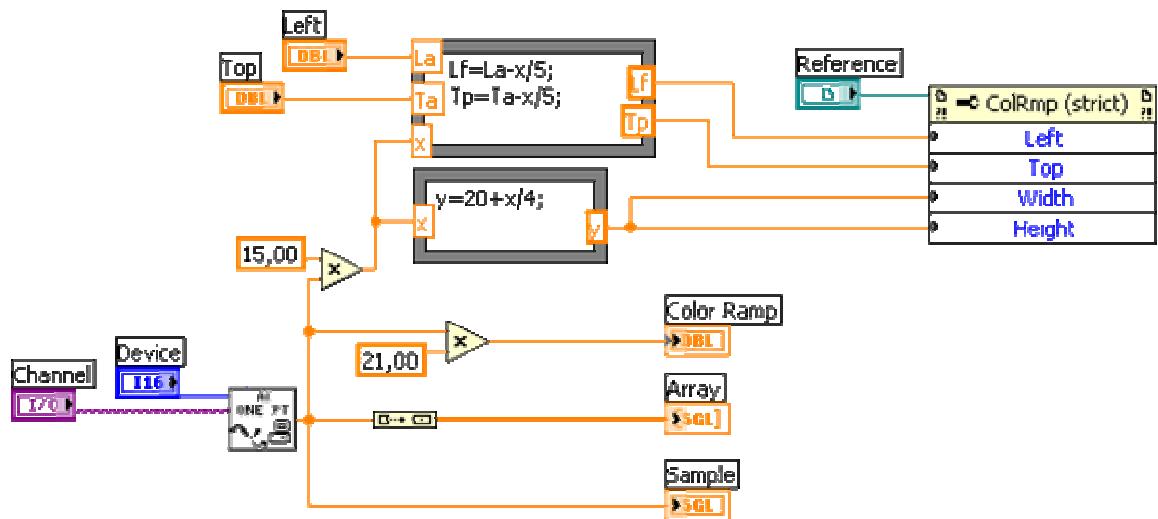
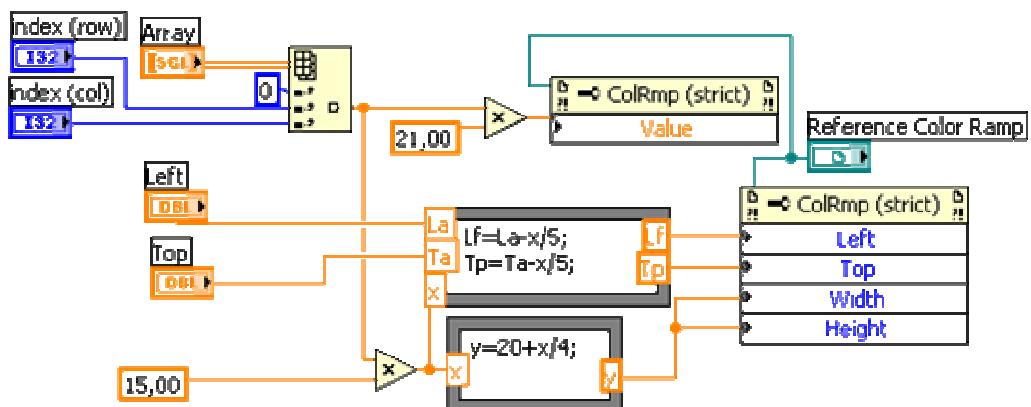


Diagrama do sub-VI de tratamento dos dados lidos a partir de um arquivo.



## Apêndice II – Programa do Microcontrolador

```
#include <ioAT89S52.h>

//*****Controles do ADC0808
#define sensor_data P0
#define START P2_bit.P2_2
#define OE P2_bit.P2_3
#define A0 P2_bit.P2_4
#define A1 P2_bit.P2_5
#define A2 P2_bit.P2_6
#define ALE P2_bit.P2_7
#define EOC P3_bit.P3_2

#define TESTPIN P2_bit.P2_0 // Para determinar taxa de amostragem

//*****Declaracao de Funcoes
void delay (void);
void config (void);
unsigned char converte_sensor (unsigned char n_sensor);

//*****Definição de Funcoes
void delay (void)
{
    unsigned char x, y;
    for (x=0; x<255; x++);
        for (y=0; y<255; y++);
}

void config (void)
{
    // Baud-rate da serial
    TMOD = 0x20; // Mode 2 no Timer 1. Baud rate generator da serial
    TH1 = 0xF2; // 2976 Bds @ 16 MHz
    TL1 = 0xF2;
    TCON = 0x40; // Liga Timer 1
    PCON_bit.SM0 = 0; // usado no calculo do Baud rate da serial

    // Clock-out
    T2MOD = 0x02; // Configura T2 como clock generator
    RCAP2L = 0xFC; // Saída do Timer 2 f=1MHz para oscilador de 16MHz
    RCAP2H = 0xFF;
    T2CON = 0x04; // Liga Timer 2

    // Modo de operacao da serial
    SCON = 0x40; // UART mode 1 (10 bits: start + 8bits + stop)

    delay ();
    TESTPIN = 0;
}

unsigned char converte_sensor (unsigned char n_sensor)
{
    unsigned char aux = 0;

    // Endereco do sensor
    if (n_sensor == 0)
    {
        A0=0;
        A1=0;
        A2=0;
    }
    else if (n_sensor == 1)
    {
        A0=1;
        A1=0;
        A2=0;
    }
    else if (n_sensor == 2)
    {
        A0=0;
        A1=1;
        A2=0;
    }
    else if (n_sensor == 3)
    {
        A0=1;
        A1=1;
        A2=0;
    }
    else if (n_sensor == 4)
    {
        A0=0;
        A1=0;
        A2=1;
    }
    else if (n_sensor == 5)
    {
        A0=1;
        A1=0;
        A2=1;
    }
    else if (n_sensor == 6)
    {
        A0=0;
        A1=1;
        A2=1;
    }
    else if (n_sensor == 7)
    {
        A0=1;
        A1=1;
        A2=1;
    }
}
```

```

}
else if (n_sensor == 1)
{
    A0=1;
    A1=0;
    A2=0;
}
else if (n_sensor == 2)
{
    A0=0;
    A1=1;
    A2=0;
}
else if (n_sensor == 3)
{
    A0=1;
    A1=1;
    A2=0;
}
else if (n_sensor == 4)
{
    A0=0;
    A1=0;
    A2=1;
}
else if (n_sensor == 5)
{
    A0=1;
    A1=0;
    A2=1;
}
else if (n_sensor == 6)
{
    A0=0;
    A1=1;
    A2=1;
}
else if (n_sensor == 7)
{
    A0=1;
    A1=1;
    A2=1;
}
else {}

asm ("nop");
asm ("nop");

ALE = 1;           // Apos endereco estavel, ALE=1

asm ("nop");
asm ("nop");

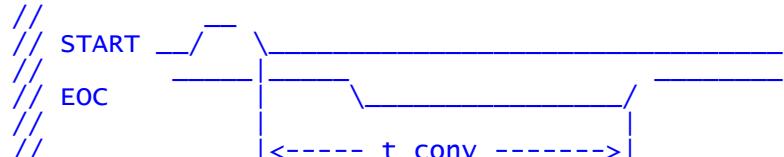
START = 1;

asm ("nop");

START = 0;          // Comeco da conversao (descida de borda)
ALE = 0;

// START _____
// EOC      _____
//           |----- t conv -----|
while (EOC); // Espera enqto o EOC ainda esta alto
while (!EOC); // Espera qnto o EOC esta baixo

```



```

OE = 1;           // OUTPUT ENABLE - disponibiliza os dados na saida

asm ("nop");
asm ("nop");
asm ("nop");
asm ("nop");

aux = sensor_data; // Armazena os dados

asm ("nop");
asm ("nop");
asm ("nop");
asm ("nop");

OE = 0;

return aux;
}

//*****Programa Principal
void main (void)
{
    unsigned char i;

config ();

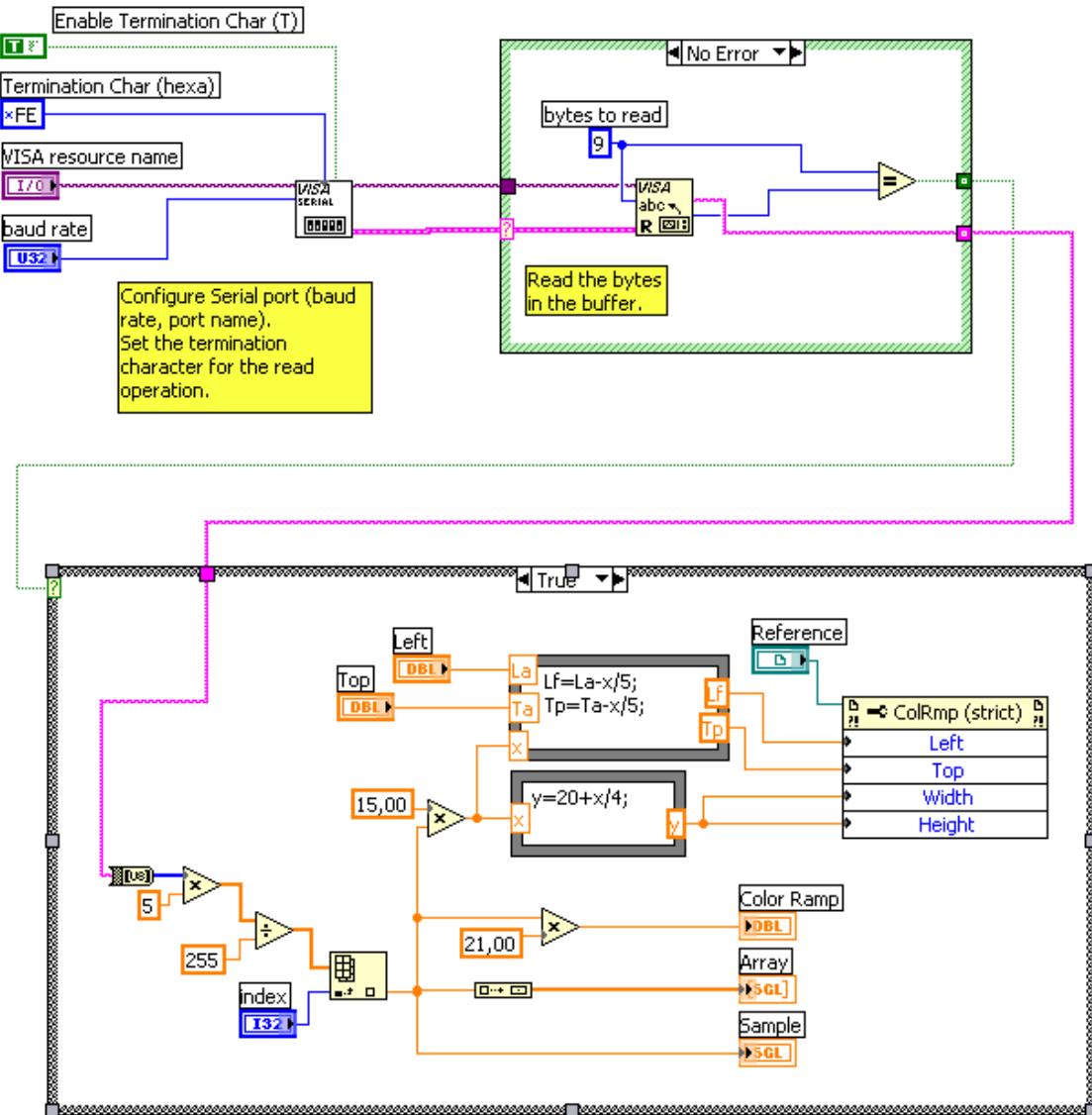
while (1)           // Loop infinito
{
    if (TESTPIN)           // A cada transmissao de um quadro
        TESTPIN = 0;        // inverte o pino de teste
    else TESTPIN = 1;

    for (i=0; i<8; i++)
    {
        SBUF = converte_sensor(i); // Joga na serial cada sensor
        while (!SCON_bit.TI)        // Limpa flag de escrita da serial
            asm ("nop");
            asm ("nop");
        SCON_bit.TI = 0;
    }
    SBUF = 0xFE;           // Joga na serial byte de controle
    while (!SCON_bit.TI)
        asm ("nop");
        asm ("nop");
    SCON_bit.TI = 0;
}
}

```

## Apêndice III – Mudanças nos VI's na Segunda Etapa

Diagrama do sub-VI de leitura a partir da porta serial.



Rotina de controle da leitura dos sensores.

