

# Lendo ângulos com um potenciômetro através de um PIC

Por João Lucas de Oliveira Torres  
09 de Abril de 2011

## Sobre o Autor

Aluno do segundo ano do curso de Engenharia Elétrica da Universidade Federal do Ceará, bolsista de Inovação Tecnológica do CNPq, foi medalhista de ouro por duas vezes e uma vez prata da Olimpíada Brasileira de Astronomia. Têm interesse por diversas áreas do conhecimento, como física, matemática, astronomia, cosmologia, eletrônica, programação, mecânica e outras. Atualmente está envolvido com uma pesquisa em sensoriamento remoto para agricultura utilizando redes de sensores sem fio.

Contato: [joao.lucas.torres@gmail.com](mailto:joao.lucas.torres@gmail.com)

## Sobre este trabalho

Este trabalho tem como objetivo ilustrar de maneira didática a utilização de um potenciômetro para ler ângulos, fazendo assim uso do conversor analógico digital e da biblioteca de controle para LCDs disponível para o microcontrolador no compilador CCS.

## Materiais

O microcontrolador utilizado nesse projeto foi o 18f2550 da Microchip, esse microcontrolador pode ser substituído por outro que possua conversor analógico para digital e uma porta B, bastando fazer algumas modificações no *firmware* apresentado adiante. O potenciômetro é do tipo genérico rotacional e linear (10K ohm), o fato de o potenciômetro ser linear será bastante útil, pois facilitará alguns cálculos. O LCD utilizado é um tipo comum 16x2, a referência do usado aqui é MGD1602B-FL-YBS.

Alguns itens diversos também são necessários, com um cristal oscilador de 20MHz, regulador de tensão e componentes passivos, porém como esse projeto visa à explicação de alguns periféricos do microcontrolador é pré-suposto que o leitor já tenha o conhecimento prévio de como utilizar esse tipo de dispositivo, sendo assim, uma pequena plataforma de uso genérico desenvolvida pelo autor foi usada para a conexão dos diversos componentes.

## O potenciômetro

“Um potenciômetro é um componente eletrônico que possui resistência elétrica ajustável. Geralmente, é um resistor de três terminais onde a conexão central é deslizante e manipulável. Se todos os três terminais são usados, ele atua como um divisor de tensão.”, Definição de potenciômetro pela Wikipédia.

Em sua grande maioria, os potenciômetros se distribuem em duas categorias, deslizantes e rotacionais, e as categorias dividem-se em tipos, linear, logarítmica, anti-logarítmica e outras.

Quanto à categoria, deslizante (figura 1) quer dizer que o eixo de controle do potenciômetro desliza sobre o mesmo a fim de alterar a resistência entre os terminais. Rotacional (figura 2) nos diz que o eixo de controle gira em torno de si para modificar a resistências entre os terminais.

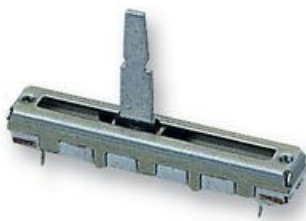


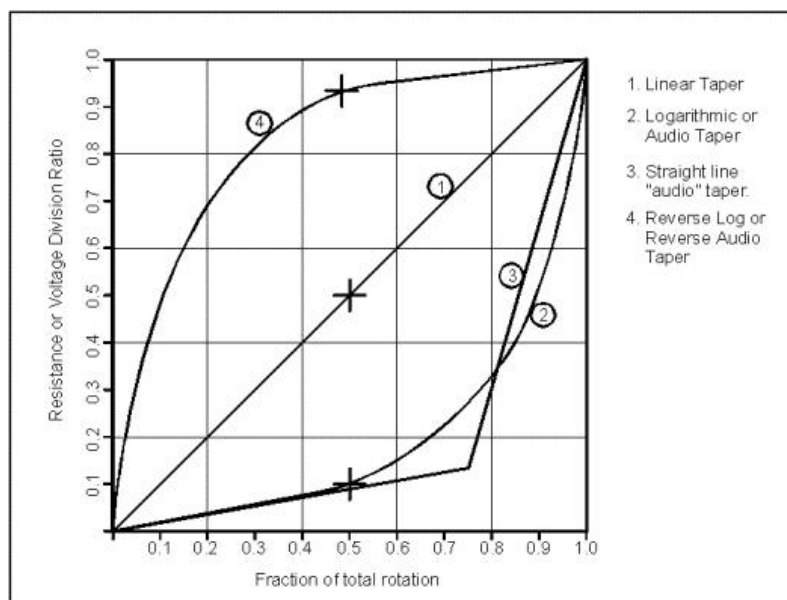
Figura 1.



Figura 2.

Quanto ao tipo, linear, logarítmica ou anti-logarítmica, refere-se à taxa de variação da resistência em função da variação do ângulo ou posição do eixo de controle.

Como podemos ver na figura abaixo.



A curva 1 representa o gráfico da variação da resistência em função do ângulo rotacionado em um potenciômetro linear, a curva 2 é o equivalente para o potenciômetro logarítmico e a curva 4 é o equivalente para o potenciômetro anti-logarítmico.

## Ângulo em função da leitura do ADC (Conversor Analógico Digital)

$$(I) \frac{R-0}{a-0} = \frac{dR}{da} = K$$

$$(II) da = \frac{1}{K} dR$$

$$(III) a = \int da = \int \frac{1}{K} dR$$

$$(IV) a = \frac{1}{K} R$$

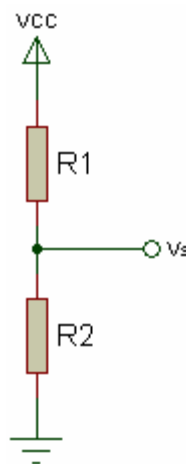
$$(V) V_{cc} = (R1 + R2) \cdot i$$

$$(VI) V_{cc} = R_{pot} \cdot i$$

$$(VII) V_s = R2 \cdot i$$

$$(VIII) R = R2$$

Lista de equações.



Potenciômetro visto como um divisor de tensão.  
Figura 3.

**R=R2:** Resistência inferior do potenciômetro.

**a:** Ângulo de rotação do potenciômetro tendo como referência a posição onde R2 é igual a zero.

**K:** Taxa de variação da resistência em função da rotação, melhor explicado adiante.

**Vcc:** Tensão de alimentação do potenciômetro, 5 volts nesse caso.

**Vs:** Tensão de saída no terminal do meio do potenciômetro.

**I:** Corrente que flui pelo potenciômetro.

**ValADC:** Valor lido pelo Conversor Analógico Digital.

**n:** Número de bits usado pelo ADC.

Sendo constante a taxa de variação da resistência em função da rotação do eixo de controle de um potenciômetro linear, podemos escrever (I); logo o diferencial do ângulo é (II); integrando em ambos os lados como visto em (III) resulta em (IV), que é uma

relação entre o ângulo rotacionado e a resistência inferior do potenciômetro.

Porém, há um problema com (IV), em (IV), R é um valor que não temos como medir diretamente, mas, como o potenciômetro funciona como um divisor de tensão, podemos achar uma relação entre as tensões e R.

Como visto na figura 3, o potenciômetro pode ser visto como um divisor de tensão, aplicando a lei de Ohm ( $U=R.I$ ), podemos relacionar as tensões  $V_{cc}$  e  $V_s$  às resistências  $R_{pot}$  (resistência nominal do potenciômetro) e  $R$  (resistência inferior do potenciômetro).

Aplicando a lei de Ohm a todo o potenciômetro obtemos (V) e aplicando ao ramo inferior obtemos (VI). Dividindo (VI) por (V) obtemos a seguinte razão:

$$\frac{V_s=R2.i}{V_{cc}=R_{pot}.i}$$

O que resulta na seguinte relação:

$$\frac{V_s}{V_{cc}} = \frac{R2}{R_{pot}} = \frac{R}{R_{pot}} = \frac{ValADC}{[(2^n)-1]}$$

Onde,  $\frac{ValADC}{[(2^n)-1]}$  é a relação entre o valor lido no ADC ( $ValADC$ ) e o máximo valor que o ADC pode retornar ( $[(2^n)-1]$ ). A relação obtida anteriormente resulta na seguinte equação:

$$R = \frac{V_s}{V_{cc}} \cdot R_{pot}$$

Onde podemos facilmente chegar a:

$$(IX) R = \frac{ValADC}{[(2^n)-1]} \cdot R_{pot}$$

Substituindo (IX) em (IV), temos que:

$$(X) a = \frac{R_{pot}}{K[(2^n)-1]} \cdot ValADC$$

Que relaciona o ângulo desejado ao valor lido pelo ADC, fazendo,

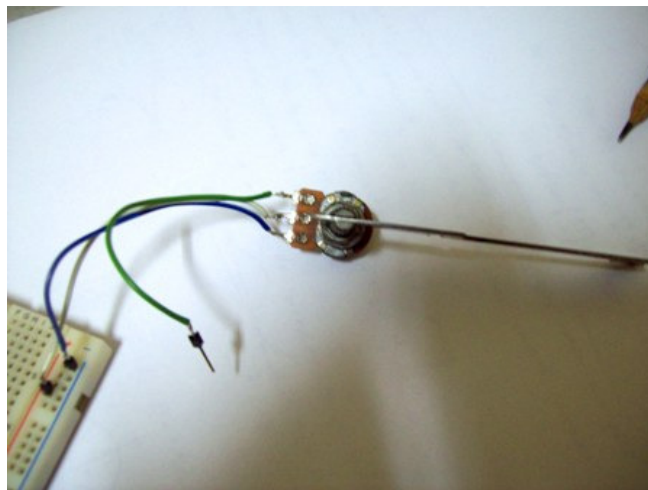
$$\frac{R_{pot}}{K[(2^n)-1]} = P$$

, temos que (X) se reduz a:

$$(XI) a = P \cdot Val_{ADC}$$

Onde P será uma característica do sistema montado. Para solucionar P temos que achar Rpot, K, e n. Rpot é a resistência total ou nominal do potenciômetro, que pode ser facilmente medida com um multímetro, no caso do sistema montado Rpot = 10,89K ohm, n é o número de bits usado pelo ADC, que é definido pelo projetista dentro do *firmware*, que nessa situação foi 10 bits, e K é a Taxa de variação da resistência em função da rotação, que foi obtida como demonstrado adiante.

Para resolver o valor de K, faz-se preciso realizar um pequeno experimento com o potenciômetro que será utilizado, fixando uma das pontas de prova do multímetro ao terminal inferior (que será ligado ao terra) e a outra ao terminal central, pode-se medir como a resistência irá variar. Fixando uma haste rígida ao eixo do potenciômetro poderemos marcar pontos para medir o ângulo percorrido mais facilmente.



Com o potenciômetro sobre uma folha em branco devemos marcar a reta sobre a qual a haste está e tomar nota da resistência lida no multímetro.

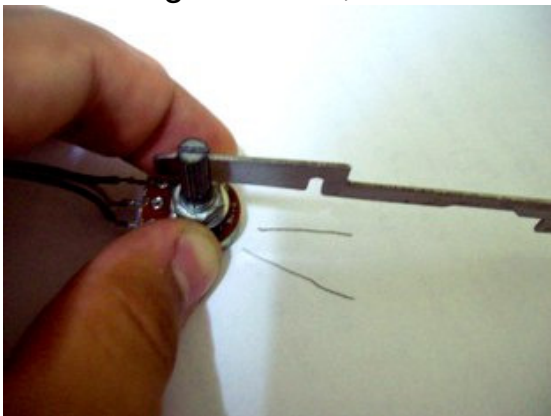


Essa primeira linha será a linha de referencia para o ângulo a ser medido. E nessa situação o valor medido no multímetro foi de 1,00K ohm.



Podemos, então, repetir o processo de girar a haste e desenhar a reta abaixo e anotar a leitura do multímetro.

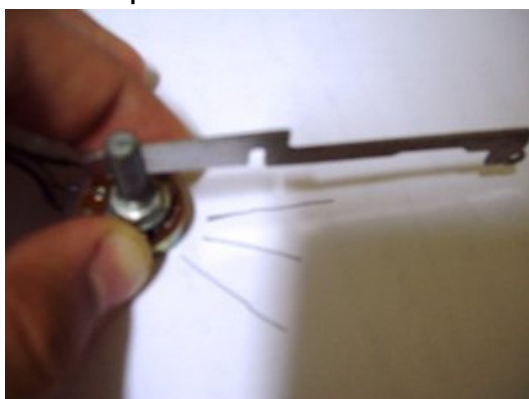
Para a segunda reta, foi feito.



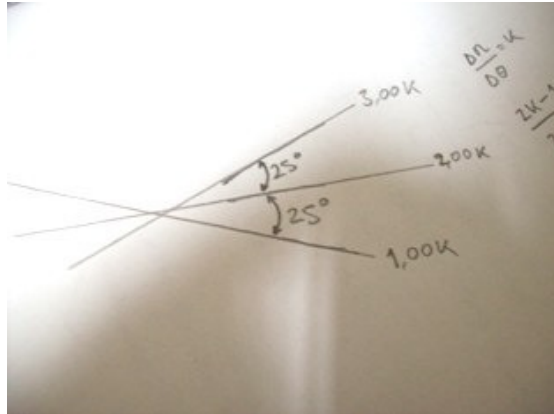
Como é visto na imagem, a leitura para o novo posicionamento foi de 2,00K ohm.

Um terceiro posicionamento pode ser feito e tomado também um terceiro valor. Essa ultima medição deve ser feita para aumentar a certeza de que a variação é constante.

Terceiro posicionamento e terceiro valor.



Feito isso, as retas desenhadas devem ser corrigidas com uma régua e também alongadas, para que o centro de rotação seja encontrado. Depois basta medir o ângulo entre as retas desenhadas.



Agora, facilmente podemos encontrar a razão entre a variação da resistência pela a variação do ângulo.

$$\frac{\Delta R}{\Delta \theta} = K$$

$$\frac{2,00k - 1,00k}{25^\circ - 0} = \frac{1000}{25}$$

$$K = 40 \text{ ohm}/^\circ$$

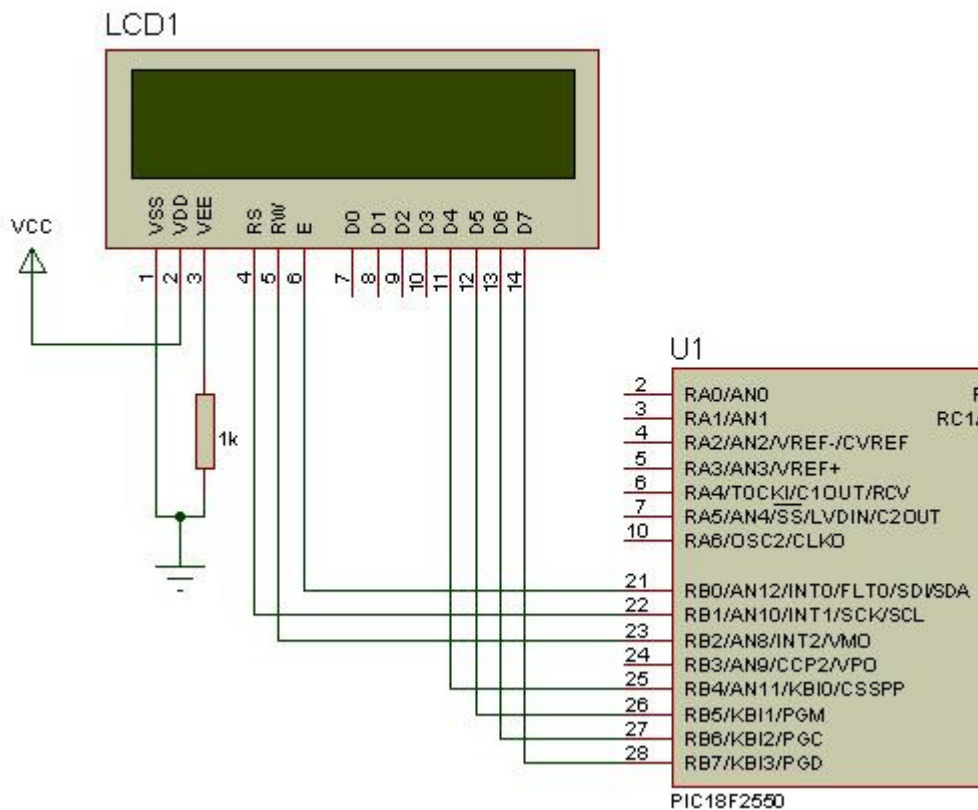
Assim, K foi determinado, e todo esse processo deve ser feito para todos os potenciômetros que possam ser utilizados com esse propósito. De posse desses valores podemos achar P, e a equação (XI) torna-se:

$$(XII) a = 0,26613 \cdot \text{ValADC}$$

O que facilitará bastante o cálculo que o processador do microcontrolador terá que fazer, lembrando que a equação (XI) só é válida para potenciômetros lineares.

## O LCD

Para conectar um display de LCD 16x2 ao PIC basta seguir o seguinte layout.



Note que o pino RB3 não está conectado, isso ocorre porque nessa configuração ele não é utilizado.

Alguns modelos de LCD possuem iluminação de fundo, o que acrescentaria mais dois pinos com polarização, ou seja, um que deve ser ligado aos 5 volts através de um resistor de 330 ohm e outro que deve ser ligado ao terra. Lembrando que o Vcc nesse caso são os mesmos 5 volts que alimentam o PIC.

Sendo o CCS o compilador utilizado para escrever o *firmware* do sistema podemos escrever algumas funções para controle do LCD utilizando a biblioteca `lcd.c`, que acompanha o CCS.

A primeira função `lcd_init()` deve ser chamada antes de ser escrever ou ler algo no LCD, isso se dá ao fato que é essa função quem “acorda” o LCD deixando-o pronto para os outros processos.

Para se enviar uma string para o LCD deve se chamar a função `lcd_putc(sua_string)`, sendo que combinar essa função com a

função printf torna a manipulação de strings mais fácil. Sendo assim, o seguinte exemplo é válido:

```
printf(lcd_putc,"Sua string, isso %d eh um inteiro",meu_inteiro);
```

Note que no exemplo existem 32 caracteres a serem exibidos no LCD(o %d representa a variável meu\_inteiro), se a variável meu\_inteiro for um número entre 0 e 9, logo essa string não caberia nas 16 posições de uma linha do LCD, porem o LCD utilizado possui duas linhas, basta dividir a string de modo que a mensagem caiba nas 32 posições totais, para fazer isso devem ser utilizados caracteres de controle. Existem 3 caracteres de controle do LCD, esse caracteres são:

\n : “quebra” a linha, serve para ir da primeira linha para a o começo da segunda.

\f : Limpa a tela do LCD, normalmente utilizado no começo da string para limpar o que havia anteriormente escrito.

\b : volta uma posição.

Assim, a string do exemplo acima poderia ser reescrita da seguinte maneira:

```
printf(lcd_putc,"\fSua string, isso\n%d eh um inteiro.",meu_inteiro);
```

note que foram introduzidos os caracteres \f e \n, de modo que o resultado, se o valor de meu\_inteiro fosse zero, seria o seguinte:

```
Sua string, isso
0 eh um inteiro.
```

Note também que foi escrito “eh” em vez de “é”, isso foi feito porque caracteres com acento não estão definidos nas rotinas do LCD.

## O ADC

“O conversor analógico-digital (frequentemente abreviado por conversor A/D ou ADC) é um dispositivo eletrônico capaz de gerar uma representação digital a partir de uma grandeza analógica, normalmente um sinal representado por um nível de tensão ou intensidade de corrente elétrica.” Definição de ADC pela Wikipédia.

Em outras palavras o que um ADC faz é discretizar uma grandeza contínua. Para ilustrar isso vamos pensar num ADC de tamanho 1, logo ele poderia representar  $2^1$  valores, ou seja, poderia representar um sinal qualquer através de dois valores discretos, o 0 e o 1. Nesse caso quando o valor do sinal fosse zero o ADC retornaria 0 e quando o valor do sinal fosse 5 volts o ADC retornaria 1. E se o tamanho do conversor fosse 2? Então ele seria capaz de representar  $2^2$  valores entre 0 e 5 volts, ou seja, o sinal poderia ser “quebrado” em quatro partes o que significa um intervalo de 1,25 volts, assim, tudo entre 0 volts e 1,25 volts retornaria 0, tudo entre 1,25 volts e 2,5 volts retornaria 1, tudo entre 2,5 volts e 3,75 volts retornaria 2 e tudo entre 3,75 volts e 5 volts retornaria 3. Desse modo é possível concluir-se o seguinte, quanto maior o tamanho do ADC, maior será a precisão da representação da grandeza analógica, pois menor serão os intervalos de tensão medidos. E é fácil perceber também, que o maior valor que o conversor irá retornar é  $(2^n)-1$ , no caso do tamanho 2, temos,  $2^2-1=4-1=3$ .

## O *firmware*

O *firmware* ou software embarcado do sistema foi escrito em C e compilado no compilador CCS. Abaixo segue o código completo que é totalmente aberto e livre para distribuição sem fins lucrativos.

```
#include<18f2550.h>//microcontrolador utilizado
#device adc=10//tamanho do ADC
#use delay(clock=20000000)//20MHz, clock do processador
#fuses hs, nolvp, nowdt, noput//fusíveis de configuração
#define use_portb_lcd TRUE//bit de config. para ligar o LCD na port B
#include <lcd.c>//driver de controle do LCD

#define p 0.26613//constante de proporção entre o ang e val
/*

    p = Rpot/{K. [(2^n)-1]}

    k[graus/byte]=R'(ang)
    n[Natural] = num. bits ADC
    Rpot[ohm] = Valor Pot.
    val[Natural] = valor lido no ADC

    ang[graus]=(p*Val)

*/

void main()//função principal
{

    int16 val;
    float ang;

    lcd_init();//inicializa o LCD

    delay_ms(10);

    //configura ADC
    setup_adc(ADC_CLOCK_INTERNAL);
    setup_adc_ports(AN0);
```

```

set_adc_channel(0);

delay_ms(10);

//menssagens de saudação exibidas no LCD
printf(lcd_putc, "\fUniv. Fed. Ceara\n");
delay_ms(3000);
printf(lcd_putc, "J. Lucas Torres");
delay_ms(3000);

while(1)//loop principal
{

    val=read_adc();//ler valor do ADC

    ang=alfa*val;//resolve o angulo para o valor lido

    //exibe a informação no LCD
    printf(lcd_putc, "\fValor do angulo\n%f graus", ang);

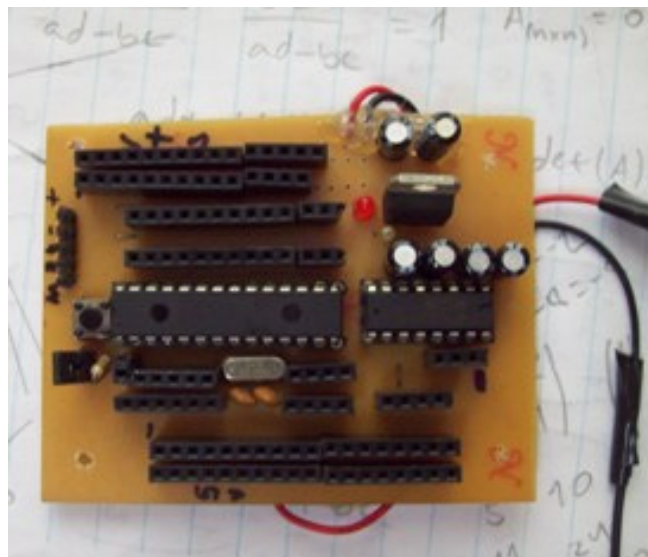
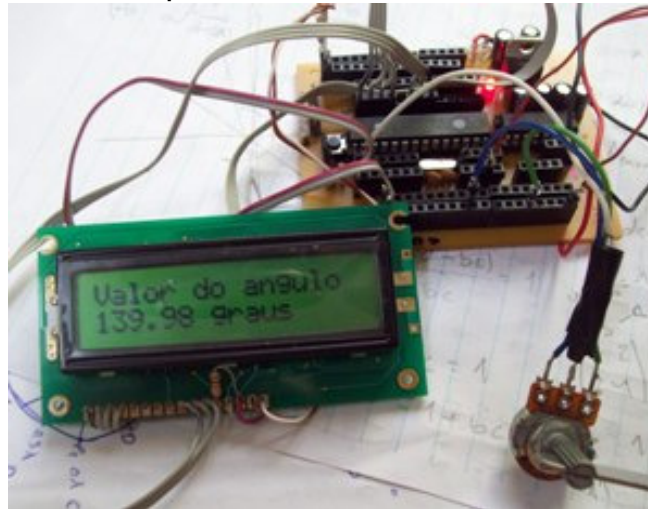
    delay_ms(500);//espera 0,5 segundos para fazer nova leitura
}
}

```

O cabeçalho do programa é auto explicativo pois está todo comentado, caso um outro microcontrolador seja utilizado no projeto de ser feita a devida alteração na linha `#include<18f2550.h>`, afim de se selecionar o componente correto, a linha `#define p 0.26613` é o valor de proporcionalidade encontrado na página 11, a linha `setup_adc_ports(AN0)` seta o pino RA0 como entrada do Conversor Analógico Digital, o restante do código também é muito auto explicativo um vez que se encontra comentado. Uma leitura no Help do CCS sobre as funções citadas com certeza resolverá qualquer dúvida sobre a estrutura do programa.

## Conclusão

Por fim, depois de compilado o arquivo .hex será gerado, após a gravação do PIC e a montagem do projeto o sistema já estará pronto para rodar. Depois de montado os teste realizados demonstraram uma grande precisão do sistema, para evitar flutuação nas medidas foi adicionado um resistor de 200K entre o terminal do meio do potenciômetro e a entrada do ADC(pino RA0). Abaixo segue uma foto do sistema funcionando e uma foto da plataforma utilizada, respectivamente.



Caso algum leitor decida por alterar algo descrito aqui, por favor enviar um email ao autor contando o que foi feito e os resultados a fim de haver um *feedback* de informação.

## Referencias

[http://www.geofex.com/article\\_folders/potsecrets/potscret.htm](http://www.geofex.com/article_folders/potsecrets/potscret.htm)

<http://pt.wikipedia.org/wiki/Potenci%C3%B4metro>

[http://pt.wikipedia.org/wiki/Conversor\\_anal%C3%B3gico-digital](http://pt.wikipedia.org/wiki/Conversor_anal%C3%B3gico-digital)