

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
CURSO DE ENGENHARIA DE COMPUTAÇÃO
MICROPROCESSADORES II

PROJETO INTEGRADO
Prótese Inteligente

CURITIBA
2009

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA
CURSO DE ENGENHARIA DE COMPUTAÇÃO
MICROPROCESSADORES II**

GIOVANNI ZANTEDESCHI

MARCELO FERNANDES

TIAGO PERETTO

VALQUIRE DE JESUS

PROJETO INTEGRADO

Prótese Inteligente

Projeto entregue ao professor Afonso Ferreira Miguel, da disciplina de Microprocessadores II, do curso de Engenharia de Computação da Pontifícia Universidade Católica do Paraná, com o objetivo de demonstrar os conhecimentos adquiridos durante o semestre e agregar a eles novos, de maneira a aplicá-los de maneira prática através de um projeto desenvolvido pelos alunos.

CURITIBA

2009

SUMÁRIO

Fundamentação Teórica.....	4
1. Introdução.....	4
1.1 Justificativas.....	5
1.2 Metodologia.....	6
1.3 As responsabilidades.....	6
2. Objetivos.....	7
3. Não está Incluso no Escopo deste Projeto.....	7
4. O Projeto.....	8
5. Os resultados esperados.....	8
6. Resultados obtidos.....	9
7. Materiais.....	9
8. Esquemático do Projeto.....	10
9. Conclusões.....	10
10. Anexos.....	11

FUNDAMENTAÇÃO TEÓRICA

Microcontroladores

Um microcontrolador é um componente que tem num único chip, além de uma CPU, elementos como memória ROM e RAM, temporizadores, contadores, canais de comunicação e conversores analógico-digitais. Esse sistema diferencia os sistemas baseados em microcontroladores daqueles baseados em microprocessadores, onde normalmente se utilizam vários componentes para implementar essas funções. Em contrapartida, as CPUs dos microcontroladores são, em geral, menos poderosas do que a dos microprocessadores. Seu clock é mais baixo e o espaço de memória endereçável costuma ser bem menor. Com isso vê-se que a área de aplicação dos dois é um pouco distinta, o microcontrolador será usado em sistemas de menor complexidade e menor custo do que um sistema que exija a capacidade de processamento de um microprocessador.

A programação dos microcontroladores é, em geral, mais simples do que a dos microprocessadores, ao menos no que diz respeito às exigências de conhecimento dos componentes periféricos. Isto acontece porque os periféricos *on-chip* dos microcontroladores são acessados de uma forma padronizada e integrada na própria linguagem de programação.

Nos dias atuais os microcontroladores são elementos eletrônicos básicos para todos os engenheiros eletricitas, isso em função do seu grande número de aplicações. Com o avanço da tecnologia e a utilização da eletrônica digital por grande parte das empresas, o emprego dos microcontroladores vêm sendo muito requisitado para um melhor desenvolvimento da produção, diminuindo os custos e trazendo benefícios para as empresas que utilizam esse sistema. Considerando a relação custo/benefício, o uso dos microcontroladores não fica restrito somente às empresas de grande/médio porte, eles podem ser usados também em projetos eletrônicos, na substituição de vários componentes digitais, obtendo-se assim no final do projeto um melhor acabamento, pois o microcontrolador ocuparia um menor espaço físico, além de uma maior eficiência e praticidade.

Alguns exemplos de sistemas onde os microcontroladores são aplicados são:

- Controle de semáforos;
- Balanças eletrônicas;
- Controle de acesso;
- Telefones públicos;
- Controle de carregadores de baterias;
- Inversores;
- Taxímetros;
- Eletrodomésticos em geral.

1. Introdução

O projeto Prótese Inteligente surgiu de um artigo lido pelo colaborador Tiago Augusto Peretto no site www.inovacaotecnologica.com.br. A informação consistia que uma empresa colocara no mercado “a primeira prótese disponível a se

aproximar dos conceitos de biônica até agora somente vistos nos filmes de ficção científica”.

O interesse do grupo na área de biotecnologia fez com que a idéia fosse aprovada e, a partir daí, as metas para a criação do conceito de prótese inteligente para fins acadêmicos começou a ser elaborado.

Os participantes do grupo serão Giovanni Zantedeschi, Marcelo Fernandes, Tiago Augusto e Valquire de Jesus. As experiências de projetos anteriores com estes participantes pode ser utilizada a fim de promover uma maior interação entre os mesmos.

1.1 Justificativas

Do ponto de vista acadêmico, a idéia de participar de um projeto que envolva biotecnologia é muito atrativa e, *a priori*, este foi o fator motivante para a elaboração do projeto.

Por outro lado, pensando a nível comercial, vê-se como um projeto deste pode ter envolvimento com diversas áreas da sociedade.

Mesmo o Brasil não sendo um país que sofra com os males da distribuição de minas terrestres pelo seu território - como ocorre em diversos países que passaram por um processo de guerra civil ou outro tipo de conflito igualmente sangrento que produz vítimas inocentes até os dias de hoje - ainda assim existem inúmeros outros fatores que causam a perda de membros, superiores e inferiores.

Entre os principais causadores estão problemas vasculares, diabetes, infecções e traumatismos provenientes de acidentes de trânsito.

Nestes casos, para as pessoas que optam pela utilização de próteses a opção mais comum é o chamado pilão, que nada mais é que um complemento para o coto e que possui diversos modelos e materiais, desde gesso até plástico ou titânio.

Das próteses mais elaboradas, estão aquelas que possuem articulações que respondem a estímulos nervosos, mas o custo destas ainda encontra-se fora da realidade da maioria das pessoas que possuem membros amputados.

O modelo de prótese de membro inferior inteligente citado anteriormente é chamado de Power Knee e é produzido pela empresa Ossus. Este modelo foi feito para pessoas que tiveram a perna amputada acima do joelho, incorporando os ligamentos e através da musculatura biônica garante um andar natural para o usuário. A prótese agrega alimentação própria, sensores, atuadores e um computador rodando um programa de inteligência artificial que permite que os pacientes caminhem naturalmente e em segurança. Assim que a prótese detecta o contato com o solo, o que é feito automaticamente por meio de sensores, ela libera o movimento em qualquer ângulo de flexão, transmitindo a sensação de total estabilidade. Sensores adicionais de torque garantem que o movimento não ficará travado ou que seja exagerado, permitindo um ritmo natural de andar.

1.2 Metodologia

A construção do protótipo foi feita com material leve, já que os motores que representarão os ligamentos devem ser capazes de mover toda a estrutura.

O “cérebro” da prótese é um microcontrolador PIC16F877A, que interpreta os sinais recebidos pelo sensor de distância, calcula a variação do valor entre o solo e a prótese e envia aos motores os ângulos de movimento necessários. A ferramenta utilizada para a confecção do *firmware* foi o MPLAB, que é o *software* mais utilizado durante esta fase do curso.

foram necessários dois servo-motores para a construção do projeto, um para fazer a ligação do joelho e outro para o tornozelo.

Além disso, foi necessário implementar um sensor de distância que fica constantemente enviando os dados referentes à variação de distância entre a prótese e o solo. Como base nestes dados o μC envia os comandos para os motores.

1.3 As responsabilidades

Na primeira fase do projeto, todos foram aplicados para a pesquisa sobre biotecnologia e construção de próteses, já que é essencial o conhecimento de todos sobre o conteúdo teórico do projeto e suas implicações.

Em um segundo momento, os colaboradores Marcelo e Giovanni foram responsáveis pela pesquisa de mercado e aquisição dos componentes e materiais.

Tendo em mãos os equipamentos necessários, a divisão de tarefas ficou: Tiago e Valquire projetaram o *firmware* do μC , Giovanni tratou do sensor e dos motores e Marcelo cuidou da estrutura em si.

Tivemos, também, o auxílio do prof^o Afonso Ferreira Miguel para tratamento das dúvidas recorrentes à programação do *firmware*, com demais professores para possíveis dúvidas a respeito da parte elétrica e mecânica e com a participação da Instituição PUCPR para utilização das instalações e de equipamentos, como computadores, *softwares* e outras necessidades.

2. Objetivos

O objetivo deste projeto foi confeccionar uma prótese de perna capaz de flexionar-se automaticamente de acordo com a distância do solo e da posição que se encontra.

Academicamente, serão desenvolvidos os estudos na área de biotecnologia, produção de próteses, aprimoramento no tratamento de servo-motores e aplicação prática de microcontroladores.

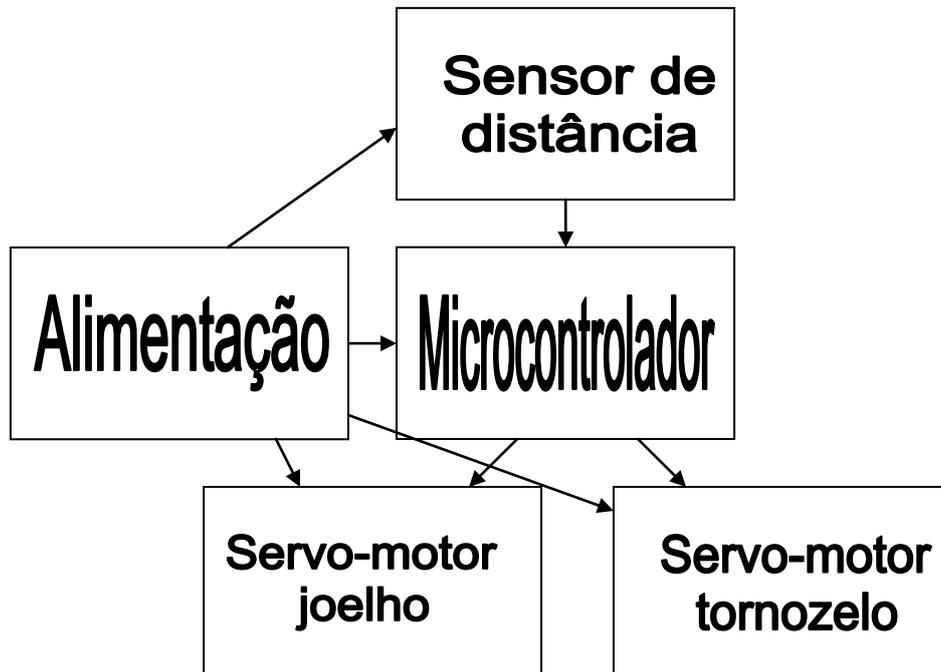
Os objetivos foram alcançados porém devido a instabilidade do sensor o projeto não funcionou como esperado, o projeto é viável e apenas algumas alterações no *firmware* ou no sensor seriam necessárias para o funcionamento correto da perna.

3. Não esta incluso no escopo deste projeto

- O projeto não foi feito de material capaz de suportar o peso de uma pessoa. A construção foi um protótipo em miniatura para demonstrar a tecnologia envolvida e não para fins comerciais.

- A prótese não foi feita para atividades complexas. O projeto foi construído visando exclusivamente a movimentação básica de uma pessoa, ou seja, caminhar. Não está no escopo atividades como correr, praticar esportes etc.

4. O projeto



O projeto consiste em um *loop*, onde o sensor de distância capta o valor corrente entre a sola da prótese e o solo. Este valor é transmitido ao μC que interpreta o valor, calcula a variação de distância – necessário para a obtenção da informação de que a passada é em direção ao corpo ou em direção ao solo – entre o valor previamente calculado e o corrente e, por último, obtem um resultado quanto a movimentação e transmite tal valor aos servo-motores para posicionar as junções de maneira apropriada.

Fechado o ciclo, um novo valor é detectado pelo sensor e o processo inicia-se novamente.

5. Os resultados esperados

Como resultados deste projeto, foram esperados:

- a) *Firmer* do projeto funcionando adequadamente em todas as suas rotinas – interpretação da informação recebida pelo sensor, cálculo da variação de movimento e transmissão do ângulo de flexão adequado aos servo-motores.
- b) Sensor de distância obtendo os dados corretamente e rotina de detecção sendo contínua.
- c) Servo-motores interpretando os ângulos de maneira confiável e executando-os de forma síncrona. Devem ser capazes de movimentar toda a estrutura que fazem parte.

6. Resultados Obtidos:

- a) *Firmer* do projeto funcionando adequadamente em todas as suas rotinas – interpretação da informação recebida pelo sensor, cálculo da variação de movimento e transmissão do ângulo de flexão adequado aos servo-motores.
- C) Servo-motores interpretando os ângulos de maneira confiável e executando-os de forma síncrona.

7. Materiais:

Dois Servo Motores Eurgle 16t02 com capacidade para 3,8 kgf/cm.

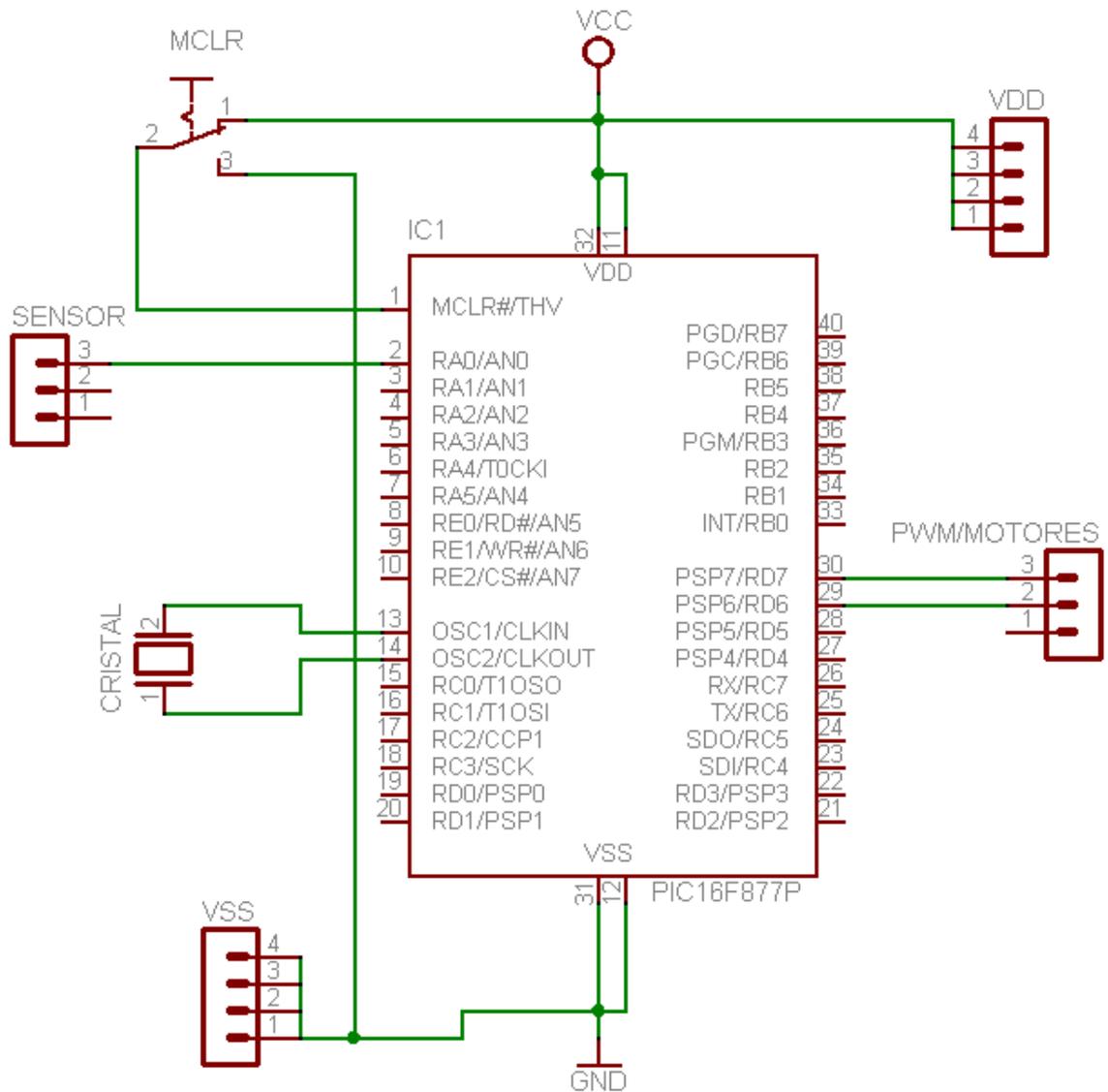
Um cristal oscilador de 3,6 MHz.

Um PIC16F877A.

Canaleta para cabos(de rede, energia, telefone etc.).

Um sensor de distância infra-vermelho GP2D12

8. Esquemático do projeto:



9. CONCLUSÕES:

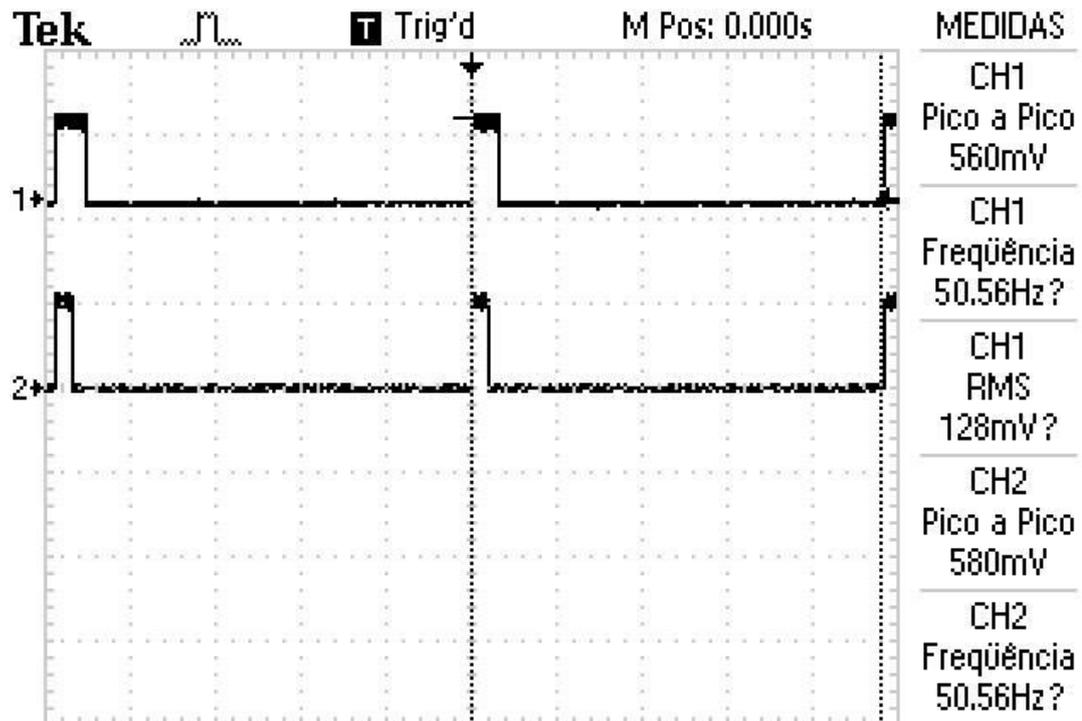
O presente projeto contribuiu para o enriquecimento intelectual dos integrantes através do estudo de referências e técnicas utilizadas no campo de microprocessadores, controle de servo e utilização do conversor A/D do PIC. Esse projeto pode ser utilizado por pessoas com deficiências físicas para facilitar a movimentação de uma maneira mais confortável.

O projeto é viável e apenas foram encontrados problemas com o sensor, que é muito instável (muda o valor muito rapidamente), fazendo com que os servos não saibam a posição real do pé e fiquem tentando corrigir a rota sem parar, fora isso o projeto funcionou bem e sem maiores dificuldades.

Inicialmente o projeto seria feito com MSP430, porém mudamos para o PIC16F877A devido a facilidade da conversão A/D já ter sido implementada em atividades do professor Afonso.

10. Anexos

Foto do Osciloscópio:



Código em linguagem Assembly do sistema projetado:

```
#include "p16F877a.inc"
```

```
__CONFIG 3F31
```

```
CBLOCK 0x20
```

```
WORK_TEMP
```

```
STATUS_TEMP
```

```
FSR_TEMP
```

PCLATH_TEMP

aux_timer ;contador de interrupções. Conta de 0 até 255.

ciclo_pwmJ ;largura do pulso do Servo que controla o joelho

ciclo_pwmT ;largura do pulso do Servo que controla o tornozelo

atual

parametro

ENDC

org 0

goto inicio

org 4

call push

BANKSEL PIR1

bcf PIR1,TMR2IF ;; zera a flag do Timer2 para poder
voltar da interrupção

BANKSEL TMR2

clrf TMR2 ;; zera o TMR2

BANKSEL aux_timer

movlw .254

subwf aux_timer, W

btfsc STATUS, C

```
goto estouro
```

```
incf aux_timer
```

rotina:

```
BANKSEL ciclo_pwmJ
```

```
movf ciclo_pwmJ, W
```

```
BANKSEL aux_timer
```

```
subwf aux_timer,W
```

```
btfss STATUS, C
```

```
goto um_joelho
```

```
goto zero_joelho
```

um_joelho:

```
BANKSEL PORTD
```

```
bsf PORTD,7
```

```
goto tornozelo
```

zero_joelho:

```
BANKSEL PORTD
```

```
bcf PORTD, 7
```

```
goto tornozelo
```

tornozelo:

```
BANKSEL ciclo_pwmT
```

```
movf ciclo_pwmT, W
BANKSEL aux_timer
subwf aux_timer, W
btfss STATUS, C
goto um_tornozelo
goto zero_tornozelo
```

um_tornozelo:

```
BANKSEL PORTD
bsf PORTD,6
goto CAD
```

zero_tornozelo:

```
BANKSEL PORTD
bcf PORTD, 6
goto CAD
```

CAD:

```
BANKSEL aux_timer
movlw 0
subwf aux_timer, W
btfss STATUS, Z
goto vaza
```

```
BANKSEL  ADCON0
```

```
btfsc  ADCON0, GO_DONE
```

```
goto  vaza
```

```
BANKSEL  ADRESH
```

```
movf  ADRESH,W
```

```
BANKSEL  atual
```

```
movwf  atual
```

```
BANKSEL  ADCON0
```

```
bsf      ADCON0, GO_DONE      ;inicia a conversão
```

```
goto  vaza
```

estouro:

```
BANKSEL  aux_timer
```

```
clrf  aux_timer
```

```
goto  rotina
```

vaza:

```
call  pop
```

```
BANKSEL  TMR2
```

```
clrf  TMR2      ;; zera o TMR2
```

```
retfie
```

inicio:

```
BANKSEL TRISD
movlw B'00000000'
movwf TRISD
BANKSEL TRISC
movlw B'00000000'
movwf TRISC
```

```
movlw 0
```

```
BANKSEL atual
```

```
movwfatual
```

```
BANKSEL parametro
```

```
movwfparametro
```

```
BANKSEL ciclo_pwmJ
```

```
movwfciclo_pwmJ
```

```
BANKSEL ciclo_pwmT
```

```
movwfciclo_pwmT
```

```
BANKSEL ADCON1
```

```
movlw B'01000000'
```

```
movwf ADCON1
```

```
BANKSEL ADCON0
```

```

movlw      B'00000101'
movwf     ADCON0

BANKSEL   PIE1
movlw     B'00000010'      ;; habilita a interrupção TMR2
movwf     PIE1

BANKSEL   PR2
movlw     .70              ;; 4MHz / 4 / 16 / 4 = 12500 *** 12500 / 256 = 48.8Hz!
movwf     PR2              ;; PR2 = 0

BANKSEL   INTCON
movlw     B'01000000'      ;; habilita todas as interrupções mascaradas
movwf     INTCON

BANKSEL   T2CON
movlw     B'00000000'      ;; prescaler 16, Postscale 4, Timer2 is on
movwf     T2CON

BANKSEL   aux_timer
movlw     .0              ;; contador pra definir o ciclo do pulso
movwf     aux_timer

BANKSEL   ciclo_pwmJ

```

```
    movlw .19      ;; este parametro define a largura do pulso, logo a
movimentação do servo
```

```
    movwfciclo_pwmJ
```

```
    BANKSEL  ciclo_pwmT
```

```
    movlw .11      ;; este parametro define a largura do pulso, logo a
movimentação do servo
```

```
    movwfciclo_pwmT
```

```
    BANKSEL  INTCON
```

```
    bsf      INTCON,GIE    ;; habilita as interrupções
```

```
    BANKSEL PIR1
```

```
    bcf      PIR1,TMR2IF   ;; zera a flag do Timer2 para poder
voltar da interrupção
```

```
    BANKSEL  T2CON
```

```
    bsf      T2CON, TMR2ON ;; liga a interrupção Timer2
```

```
    goto  ratimum
```

subindo:

```
    call  desceJoelho
```

```
    call  sobeTornozelo
```

```
    BANKSEL atual
```

```
    movf  atual, W
```

```
    BANKSEL parametro
```

```
    movwfp parametro
```

```
    goto  ratimum
```

descendo:

```
call  desceTornozelo
```

```
call  sobeJoelho
```

```
BANKSEL atual
```

```
movf  atual, W
```

```
BANKSEL parametro
```

```
movwf parametro
```

```
goto  ratibum
```

push:

```
BANKSEL WORK_TEMP
```

```
movwf WORK_TEMP
```

```
swapf STATUS, W
```

```
BANKSEL STATUS_TEMP
```

```
movwf STATUS_TEMP
```

```
BANKSEL FSR
```

```
movf  FSR, W
```

```
BANKSEL FSR_TEMP
```

```
movwf FSR_TEMP
```

```
BANKSEL PCLATH_TEMP
```

```
movf  PCLATH_TEMP, W
```

```
clrf  PCLATH
```

```
clrf STATUS
```

```
return
```

```
pop:
```

```
BANKSEL PCLATH_TEMP
```

```
movf PCLATH_TEMP,W
```

```
movwfPCLATH
```

```
BANKSEL FSR_TEMP
```

```
movf FSR_TEMP,W
```

```
BANKSEL FSR
```

```
movwfFSR
```

```
BANKSEL STATUS_TEMP
```

```
swapf STATUS_TEMP,W
```

```
movwfSTATUS
```

```
BANKSEL WORK_TEMP
```

```
swapf WORK_TEMP,F
```

```
swapf WORK_TEMP,W
```

```
return
```

```
desceJoelho:
```

```
movlw.9
```

```
BANKSEL ciclo_pwmJ
```

```
subwf ciclo_pwmJ, W  
btfss STATUS, Z  
decf ciclo_pwmJ  
return
```

desceTornozelo:

```
movlw .11  
BANKSEL ciclo_pwmT  
subwf ciclo_pwmT, W  
btfss STATUS, Z  
decf ciclo_pwmT  
return
```

sobeJoelho:

```
movlw .19  
BANKSEL ciclo_pwmJ  
subwf ciclo_pwmJ, W  
btfss STATUS, Z  
incf ciclo_pwmJ  
return
```

sobeTornozelo:

```
movlw .21
```

```
BANKSEL ciclo_pwmT  
subwf ciclo_pwmT, W  
btfss STATUS, Z  
incf ciclo_pwmT  
return
```

ratibum:

```
BANKSEL atual  
movf atual, W  
BANKSEL PORTC  
movwf PORTC  
BANKSEL parametro  
subwf parametro, W  
BANKSEL STATUS  
btfsc STATUS, Z  
goto ratibum  
btfsc STATUS, C  
goto subindo  
goto descendo
```

END