

CENTRO UNIVERSITÁRIO INTERNACIONAL UNINTER

ESCOLA SUPERIOR POLITÉCNICA

BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO

DISCIPLINA DE LINGUAGEM DE PROGRAMAÇÃO

Prova bimestral - MICROCONTROLADORES

ISAIAS ROBERTO DE LIMA E SILVA – RU: 2588009

JULHO CESAR RADICHESKI DA COSTA - RU: 973129

nome professor: Winston Sen Lun Fung

CURITIBA - PR

2020

**ENUNCIADO**

Desenvolver um braço robotico, utilizando como material principal foamboard, usando moldes propostos pelo professor e contendo 3 eixos de rotação, sendo estes operados por dois motores de passo e um servomotor.

O Software do braço deve ser construido em Assembly e utilizando o microcontrolador PIC16F877A e a placa de aprendizagem HJ-5G.

**MONTAGEM**

A montagem iniciou-se pelo software, utilizando os botoes do teclado matricial da placa, e conectando as entradas VCC e GND pela protoboard. Foi feito com sucesso o código, no qual utiliza o torque maximo do motor de passo e uma aplicação totalmente digital no servomotor.

CODIGO:

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Definição do processador \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include p16F877A.inc

\_\_config \_HS\_OSC & \_WDT\_OFF & \_LVP\_OFF & \_PWRTE\_ON

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Memória de programa \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

DELAY EQU 0x24

VEZES EQU 0x25

loop EQU 0x26

ORG 0

RESET nop

goto START

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Interrupção \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ORG 4

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Inicio do programa \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

START

bsf STATUS, RP0

movlw b'00000000'

movwf TRISC

movlw b'00000000'

movwf TRISD

movlw b'00001111'

movwf TRISB

movlw b'00000000'

movwf TRISA

bcf STATUS, RP0

INICIO

movlw b'11101111'

movwf PORTB

BTFSS PORTB, 0

GOTO BOTAO\_0

BTFSS PORTB, 1

GOTO BOTAO\_1

BTFSS PORTB, 2

GOTO BOTAO\_2

BTFSS PORTB, 3

GOTO BOTAO\_3

movlw b'11011111'

movwf PORTB

BTFSS PORTB, 0

GOTO BOTAO\_4

BTFSS PORTB, 1

GOTO BOTAO\_5

goto INICIO

BOTAO\_0

movlw b'11111111'

movwf PORTC

CALL PERDE\_TEMPO

CLRF PORTC

CALL PERDE\_TEMPO\_2

BTFSS PORTB, 1

GOTO BOTAO\_1

CALL PERDE\_TEMPO\_MOTOR

GOTO INICIO

BOTAO\_1

movlw b'11111111'

movwf PORTC

CALL PERDE\_TEMPO\_3

CLRF PORTC

CALL PERDE\_TEMPO\_4

BTFSS PORTB, 0

GOTO BOTAO\_0

CALL PERDE\_TEMPO\_MOTOR

GOTO INICIO

BOTAO\_2

movlw b'00001100'

movwf PORTD

call PERDE\_TEMPO\_MOTOR

movlw b'00000110'

movwf PORTD

call PERDE\_TEMPO\_MOTOR

movlw b'00000011'

movwf PORTD

call PERDE\_TEMPO\_MOTOR

movlw b'00001001'

movwf PORTD

call PERDE\_TEMPO\_MOTOR

BTFSS PORTB, 2

GOTO BOTAO\_2

CALL PERDE\_TEMPO\_MOTOR

GOTO INICIO

BOTAO\_3

movlw b'00001001'

movwf PORTD

call PERDE\_TEMPO\_MOTOR

movlw b'00000011'

movwf PORTD

call PERDE\_TEMPO\_MOTOR

movlw b'00000110'

movwf PORTD

call PERDE\_TEMPO\_MOTOR

movlw b'00001100'

movwf PORTD

call PERDE\_TEMPO\_MOTOR

BTFSS PORTB, 3

GOTO BOTAO\_3

CALL PERDE\_TEMPO\_MOTOR

GOTO INICIO

BOTAO\_4

movlw b'00001100'

movwf PORTA

call PERDE\_TEMPO\_MOTOR

movlw b'00000110'

movwf PORTA

call PERDE\_TEMPO\_MOTOR

movlw b'00000011'

movwf PORTA

call PERDE\_TEMPO\_MOTOR

movlw b'00001001'

movwf PORTA

call PERDE\_TEMPO\_MOTOR

BTFSS PORTB, 0

GOTO BOTAO\_4

CALL PERDE\_TEMPO\_MOTOR

GOTO INICIO

BOTAO\_5

movlw b'00001001'

movwf PORTA

call PERDE\_TEMPO\_MOTOR

movlw b'00000011'

movwf PORTA

call PERDE\_TEMPO\_MOTOR

movlw b'00000110'

movwf PORTA

call PERDE\_TEMPO\_MOTOR

movlw b'00001100'

movwf PORTA

call PERDE\_TEMPO\_MOTOR

BTFSS PORTB, 1

GOTO BOTAO\_5

CALL PERDE\_TEMPO\_MOTOR

GOTO INICIO

PERDE\_TEMPO

MOVLW d'6' ;Função que 'perde' tempo, aguardando um intervalo de tempo sem nenhuma instrução (necessário para evitar alguns bugs)

MOVWF VEZES

LOOP\_VEZES

MOVLW d'250'

MOVWF DELAY

CALL DELAY\_US

DECFSZ VEZES,1

GOTO LOOP\_VEZES

RETURN

DELAY\_US

NOP

NOP

DECFSZ DELAY,1

GOTO DELAY\_US

RETURN

PERDE\_TEMPO\_2

MOVLW d'74' ;Função que 'perde' tempo, aguardando um intervalo de tempo sem nenhuma instrução (necessário para evitar alguns bugs)

MOVWF VEZES

LOOP\_VEZES\_2

MOVLW d'250'

MOVWF DELAY

CALL DELAY\_US\_2

DECFSZ VEZES,1

GOTO LOOP\_VEZES\_2

RETURN

DELAY\_US\_2

NOP

NOP

DECFSZ DELAY,1

GOTO DELAY\_US\_2

RETURN

PERDE\_TEMPO\_3

MOVLW d'10' ;Função que 'perde' tempo, aguardando um intervalo de tempo sem nenhuma instrução (necessário para evitar alguns bugs)

MOVWF VEZES

LOOP\_VEZES\_3

MOVLW d'250'

MOVWF DELAY

CALL DELAY\_US\_3

DECFSZ VEZES,1

GOTO LOOP\_VEZES\_3

RETURN

DELAY\_US\_3

NOP

NOP

DECFSZ DELAY,1

GOTO DELAY\_US\_3

RETURN

PERDE\_TEMPO\_4

MOVLW d'70' ;Função que 'perde' tempo, aguardando um intervalo de tempo sem nenhuma instrução (necessário para evitar alguns bugs)

MOVWF VEZES

LOOP\_VEZES\_4

MOVLW d'250'

MOVWF DELAY

CALL DELAY\_US\_4

DECFSZ VEZES,1

GOTO LOOP\_VEZES\_4

RETURN

DELAY\_US\_4

NOP

NOP

DECFSZ DELAY,1

GOTO DELAY\_US\_4

RETURN

PERDE\_TEMPO\_MOTOR

MOVLW d'30' ;Função que 'perde' tempo, aguardando um intervalo de tempo sem nenhuma instrução (necessário para evitar alguns bugs)

MOVWF VEZES

LOOP\_VEZES\_MOTOR

MOVLW d'30'

MOVWF DELAY

CALL DELAY\_US\_MOTOR

DECFSZ VEZES,1

GOTO LOOP\_VEZES\_MOTOR

RETURN

DELAY\_US\_MOTOR

NOP

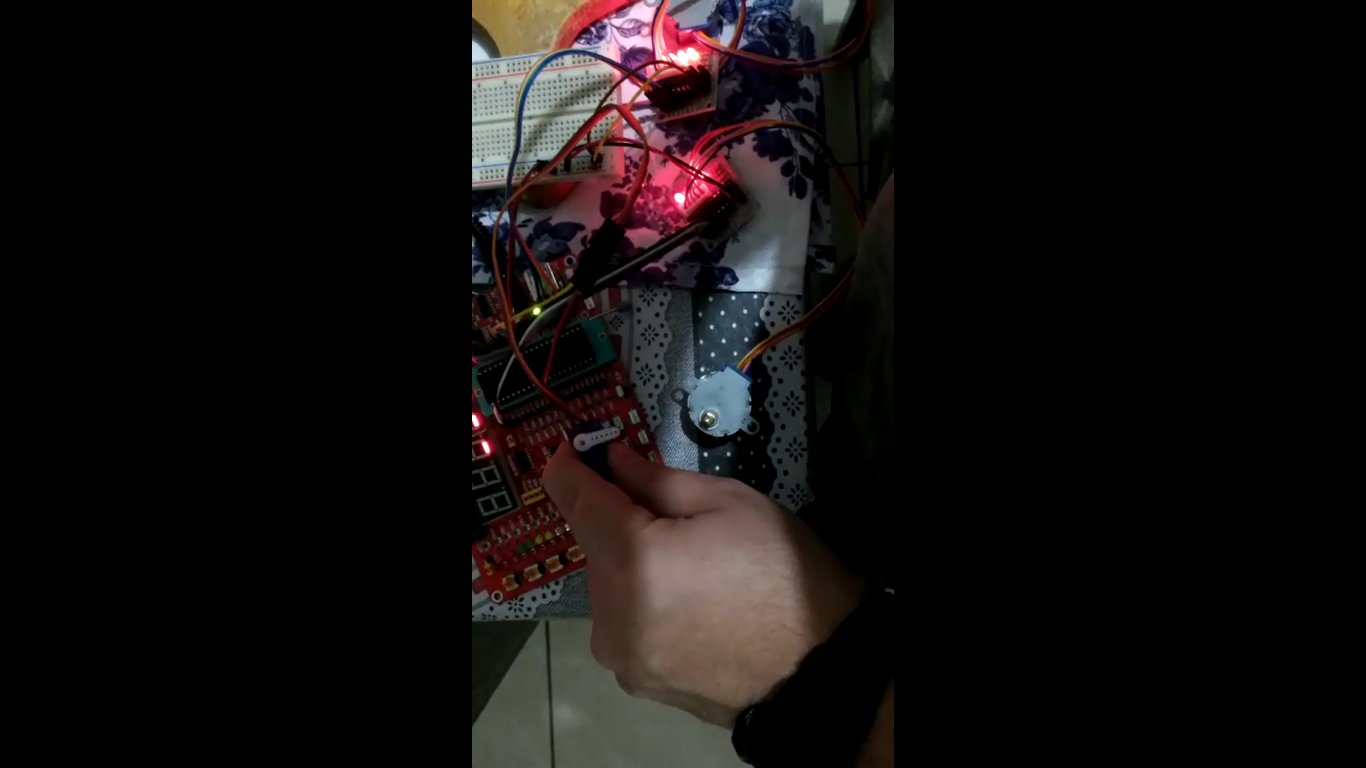
NOP

DECFSZ DELAY,1

GOTO DELAY\_US\_MOTOR

RETURN

END



*IMAGEM 1 – testagem do software nos motores.*

Após exito no codigo, foi realizado de fato a montagem do braço. Como esta montagem é uma adaptação de um projeto que utiliza 6 servomotores, algumas alterações não descritas no manual original tiveram de ser realizadas, com isso infelizmente os motores de passo não tiveram exito em ter força mecânica suficiente para mexer o braço (apesar do codigo utilizar o torque maximo do motor), sendo apenas o servomotor tendo exito em mexer a garra.



*IMAGEM 2 – garra do braço aberta pelo servomotor.*



*IMAGEM 3 – garra do braço fechada pelo servomotor.*