



UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE INFORMÁTICA
MICROCONTROLADORES

PROJETO INTEGRADO II

Conversor A/D
Medição de Tensão e Indicação no LCD

Aluno

Francisco Erberto de Sousa

Professor

Mardson Freitas de Amorin

Março, 2019

Francisco Erberto de Sousa

PROJETO INTEGRADO II

Projeto da Disciplina de Microntro-
ladores: Conversor A/D: Medição de
Tensão e Indicação no LCD.

Março, 2019

Sumário

1	Introdução	3
1.1	Contextualização	3
1.2	Apresentação do Tema	3
1.3	Fundamentação Teórica	3
1.4	Objetivos	5
2	Materiais e Métodos	6
2.1	Fluxograma	6
2.2	Especificações	6
3	Resultados	8
3.1	Configurações	8
3.2	Subrotinas	9
4	Conclusão	13
5	Referências bibliográficas	14

Lista de Figuras

1	Exemplo de um Conversor A/D	3
2	Barramento I ² C	4
3	Condição START e STOP	5
4	Fluxograma da Solução do Problema	6
5	Configuração do Endereço do Slave para Envio	8
6	Configuração do Endereço do Slave para Envio	8
7	Subrotina da Média	9
8	Subrotina do Protocolo I ² C	10
9	Subrotina do envio do Byte	11
10	Subrotina do Protocolo I ² C	12

1 Introdução

1.1 Contextualização

Este relatório tem o intuito de explicar como foi desenvolvido o segundo trabalho da disciplina de Microntroladores. Do qual abordou a conversão A/D, protocolo I²C, TIMR1, Display de LCD e entre outros conteúdos já visto no primeiro trabalho.

1.2 Apresentação do Tema

O projeto consiste na implementação de um conversor A/D capaz de se utilizar de 32 valores e posteriormente realizar a média. Para a exibição do resultado obtida na média é utilizado um KIT LCD (desenvolvido pelo LA-BEC), que para realizar a comunicação entre o microcontrolador e o display de LCD utiliza o protocolo I²C.

1.3 Fundamentação Teórica

Vivemos num mundo analógico, tudo que está a nossa volta é analógico. As grandezas como tensão, corrente, temperatura, pressão e etc são sinais analógicos. Para que um sistema digital possa interpretar um sinal analógico é necessário um conversor A/D, como pode ser visto na figura 1.

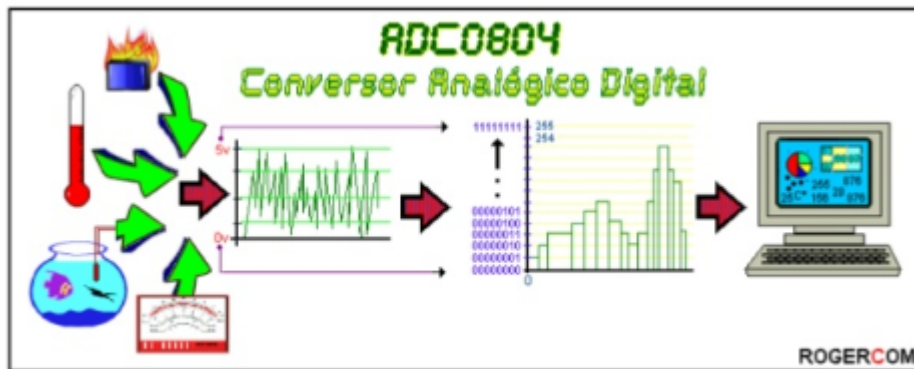


Figura 1: Exemplo de um Conversor A/D

O conversor A/D faz algumas amostragens e depois converte o sinal lido num sinal binário conforme sua precisão em bits. No caso de um conversor com 8 bits o valor de tensão num sinal digital assume um intervalo de 0 a 255. A resolução depende da quantidade de bits e pode ser calculada da seguinte forma: $\text{resolução} = V_{\text{ref}}/2^n$.

- V_{ref} - referência para conversão;
- n - Quantidade de bits do conversor;

O protocolo I²C possibilita a utilização de grande quantidade de componentes padronizados, os quais podem realizar diversas funções, além de possibilitar a troca eficaz de informações entre eles. O conceito do barramento I²C é facilitar a integração de circuitos de caráter final de aplicação como por exemplo sensores e conversores, com um sistema de controle, de modo que eles possam trabalhar com seus sinais de maneira direta.

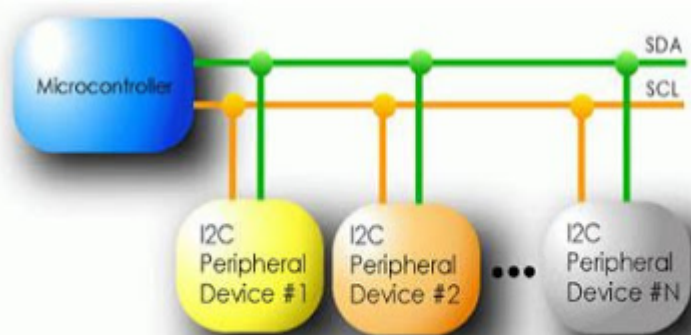


Figura 2: Barramento I²C

Uma característica bastante interessante deste barramento é a possibilidade de utilizar, em um mesmo sistema, componentes de tecnologias construtivas diferentes sem que haja incompatibilidade e nem conflitos na comunicação. No I²C a transmissão da informação entre os dispositivos é feita através de 2 fios (Serial Data SDA e Serial Clock SCL).

Condições de Start e Stop: a comunicação entre os dispositivos mestres e os escravos, conectados ao barramento I2C, é iniciada pela condição de start (início) e finalizada pela condição de stop (fim). Acompanhe a condição de start e stop na figura seguinte:

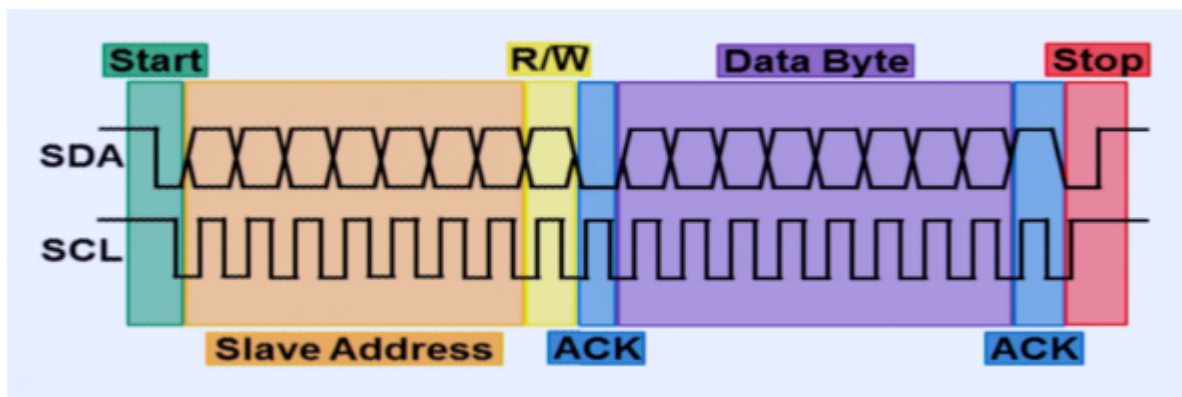


Figura 3: Condição START e STOP

Dentro da operação I2C, temos os procedimentos start e stop. Uma transição de nível alto para baixo na linha SDA, enquanto a linha SCL está no nível alto, é o indicativo da situação de START. Já uma transição do nível baixo para o nível alto da linha SDA enquanto a linha SCL se mantém no nível alto, define uma condição STOP. Sempre o mestre é o responsável pela geração dessas condições. Após uma condição de START o barramento é considerado ocupado, e apenas volta a ficar livre algum tempo depois da condição de STOP.

1.4 Objetivos

O objetivo deste trabalho é mostrar que aluno tem domínio dos conceitos apresentados no segundo módulo da disciplina e que sabe aplicar isso na prática. Para isso, foram desenvolvidos exercícios. Dos quais foram:

- Conversor A/D;
- Protocolo I²C;
- Display LCD;

2 Materiais e Métodos

2.1 Fluxograma

Para iniciar a solução foi implementando um fluxograma de maneira geral para visualizar uma solução para o problema proposto. A figura 4 mostra uma visão geral.

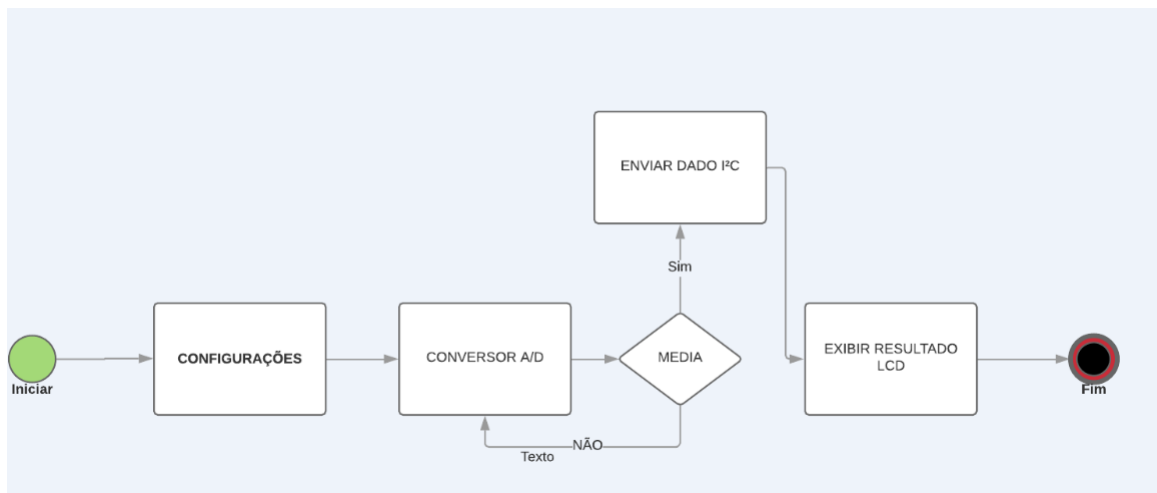


Figura 4: Fluxograma da Solução do Problema

O projeto contém algumas especificações que foram solicitadas conforme estão listadas abaixo:

- Conversor A/D;
- Protocolo I²C;
- Display LCD;

2.2 Especificações

- DEVE ser implementado com o PIC12F675;
- Utilizar o kit LCD/PIC2F675, disponível no LABEC2;

- A conversão A/D de 32 valores deve ser efetuada para calcular a média de cada medida;
- A conversão A/D deve ser efetuada, em modo cíclico e tão rápido quanto possível (limitado pela velocidade do microcontrolador);
- O valor da média conversão A/D deve ser transformado para o correspondente valor de tensão com, pelo menos, uma casa decimal de precisão;
- O valor da tensão, em notação de base decimal, deve ser enviado ao módulo LCD, identificando a grandeza e sua unidade;
- Para envio das informações, o protocolo I2C deve ser implementado no PIC no modo MASTER;
- A visualização dos dados deve ser feita com a utilização da placa LCD, que já está preparada para comunicação I2C (ver documentação específica);
- O endereço do módulo LCD é 03h (gravado na posição 00h da EEPROM do módulo LCD)

3 Resultados

3.1 Configurações

Configuração dos registradores

```
BANK1           ;ALTERA PARA O BANCO 1
MOVLW  B'00000111' ;CONFIGURA TODAS AS PORTAS DO GPIO (PINOS)
MOVWF  TRISIO     ;COMO SAÍDAS, MAS GP2 COMO ENTRADA
MOVLW  B'00100100'
MOVWF  ANSEL      ;DEFINE PORTAS COMO Digital I/O
MOVLW  B'00000100'
MOVWF  OPTION_REG ;DEFINE OPÇÕES DE OPERAÇÃO
MOVLW  B'01000000'
MOVWF  INTCON      ;HABILITA INTERRUPÇÃO E INT DE PERTIFERICOS
MOVLW  B'00000001'
MOVWF  PIE1        ;HABILITA INTERRUPÇÃO DO TMR1

BANK0           ;RETORNA PARA O BANCO
MOVLW  B'00100000' ;TMR1 PRESCALE 1:4
MOVWF  T1CON
MOVLW  B'00000111'
MOVWF  CMCON       ;DEFINE O MODO DE OPERAÇÃO DO COMPARADOR ANALÓGICO
MOVLW  B'00001001'
MOVWF  ADCON0
```

Figura 5: Configuração do Endereço do Slave para Envio

Configuração do endereço do SLAVE

```
;*****
;*                               CONSTANTES                               *
;*****
; DEFINIÇÃO DE TODAS AS CONSTANTES UTILIZADAS PELO SISTEMA
#define ENDEREÇO H'03' ;ENDEREÇO DO SLAVE
```

Figura 6: Configuração do Endereço do Slave para Envio

3.2 Subrotinas

Segue a implementação das subrotinas nas figuras abaixo; A primeira é o cálculo da média dos 32 valores.

```

;CALCULA MÉDIA DO VALORES LIDOS /// SOMA(VALORES)/32
MEAN
    CLRf    CONT1
    BSf     CONT1,0
    CLRf    AUX
    MOVLW   B'00110110'
    MOVWF   FSR
    MOVWF   INDF

MEAN_INNER
    INCf    FSR
    ADDWF   INDF,W
    BTFSC   STATUS,C
    INCf    AUX
    INCf    CONT1        ;ADICIONA OS VALORES LIDOS AO WORK
    BTFSS   CONT1,5      ;USANDO A VARIÁVEL AUX COMO PARTE ALTA
    GOTO    MEAN_INNER
    INCf    FSR
    MOVWF   INDF

    RRF     AUX           ;DIVIDINDO SOMA POR 32 (5 ROTATES PARA DIREITA)
    RRF     INDF          ;COMO TEM BITS MAIS SIGNIFICATIVOS NA VAR AUX
    ;USO O VALOR DO ROTATE RIGHT PARA PASSAR O BIT
    RRF     AUX           ;MENOS SIGNIFICATIVO DA PARTE ALTA PARA O C
    RRF     INDF          ;E PASSA ESSE BIT PARA A PARTE MAIS SIGNIFICATIVA
    ;DA PARTE BAIXA

    RRF     AUX
    RRF     INDF

    RRF     AUX
    RRF     INDF

    RRF     AUX
    RRF     INDF

```

Figura 7: Subrotina da Média

Em seguida foi implementada a subrotina para o protocolo I²C

```

ACK
BTFSC SCL ;ESPERA CLOCK ESTAR EM LOW
GOTO $-1
BANK1
BCF SDA_IO ;MUDA PORTA DO SDA PARA SAÍDA
BANK0
BCF SDA ;MANDA ACK, I.E SDA EM LOW

BTFSC SCL
GOTO $-1
BTFSS SCL ;ESPERA CLOCK IR PARA 1, I.E MASTER IRÁ LER ACK BIT
GOTO $-1

BTFSC SCL ;ESPERA CLOCK IR PARA ZERO
GOTO $-1

BCF SCL
BANK1
BCF SCL_IO ;MUDA PORTA DO SCL PARA SAÍDA
BSF SDA_IO ;LIBERA SDA
BANK0
BCF SCL ;JOGA CLOCK PARA LOW, MASTER ESPERAR

RETURN

;*****

```

Figura 8: Subrotina do Protocolo I²C

```

SEND_BYTE
    CLRF    ADDR
    MOVLW   .8
    MOVWF   RCONT
R_LOOP
    BTFSC   SDA           ;ESPERA BORDA DE SUBIDA DO SCL
    GOTO    $-1
    BTFSS   SDA
    GOTO    $-1
    BSF     STATUS,C      ;LIMPA CARRY
    BTFSC   SCL           ;SE SDA ESTIVER ATIVO
    BSF     STATUS,C      ;SETA CARRY
    RLF     ADDR          ;ROTACIONA CARRY PARA REGISTRADOR ADDR
    DECFSZ  RCONT         ;REPERE POR 8 VEZES
    GOTO    R_LOOP

    RETURN

```

Figura 9: Subrotina do envio do Byte

```

ACK
BTFSC SCL ;ESPERA CLOCK ESTAR EM LOW
GOTO $-1
BANK1
BCF SDA_IO ;MUDA PORTA DO SDA PARA SAÍDA
BANK0
BCF SDA ;MANDA ACK, I.E SDA EM LOW

BTFSC SCL
GOTO $-1
BTFSS SCL ;ESPERA CLOCK IR PARA 1, I.E MASTER IRÁ LER ACK BIT
GOTO $-1

BTFSC SCL ;ESPERA CLOCK IR PARA ZERO
GOTO $-1

BCF SCL
BANK1
BCF SCL_IO ;MUDA PORTA DO SCL PARA SAÍDA
BSF SDA_IO ;LIBERA SDA
BANK0
BCF SCL ;JOGA CLOCK PARA LOW, MASTER ESPERAR

RETURN

;*****

```

Figura 10: Subrotina do Protocolo I²C

4 Conclusão

O projeto não foi concluído e não consegui finalizar com os testes. A execução da solução desse problema possibilitou avançar nos conhecimentos do PIC. Dentre eles, o protocolo I²C e o TIMER1, além de utilizar dos conhecimentos visto em sala de aula, como: conversor A/D e Interrupções. Ficou em aberto os testes, além disso a exibição de uma cada decimal.

5 Referências bibliográficas

- Datasheet *PIC12F629/675 Data Sheet 8-Pin FLASH-Based 8-Bit CMOS Microcontrollers*. Datasheet
- David José de Souza *Desbravando o PIC. Ampliado e Atualizado Para PIC 16f628a*. Livro