

# 8051

Prof. Evandro L. L. Rodrigues

## Técnicas de Interface

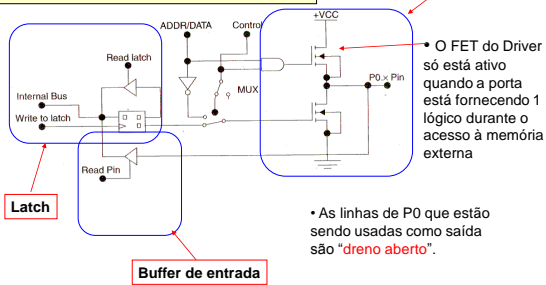
## Técnicas de Interface com Microcontroladores

(8051)

### Estrutura das Portas do 8051:

- Todas as quatro Portas (P0,P1,P2,P3) da família de Microcontroladores MCS-51 são bi-direcionais.
- Cada uma delas consiste de um **Latch**, um **Driver** de saída e um **Buffer** de entrada.
- Os Drivers de saída e os Buffers de entrada da Porta P3 são multifuncionais - podem também assumir funções alternativas.

### Arquitetura interna da Porta P0



Driver

• O FET do Driver só está ativo quando a porta está fornecendo 1 lógico durante o acesso à memória externa

• As linhas de P0 que estão sendo usadas como saída são "dreno aberto".

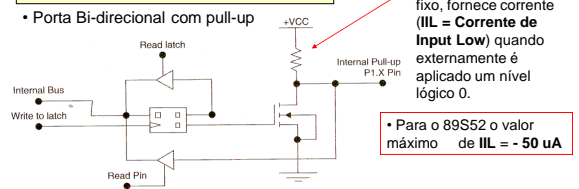
• Escrever nível lógico 1 em qualquer pino da P0 desativa os FETs de saída e assim o pino flutua.

Latch

Buffer de entrada

• Porta Bi-direcional verdadeira

### Arquitetura interna da Porta P1



• Porta Bi-direcional com pull-up

• O Resistor de Pull-up fixo, fornece corrente (IIL = Corrente de Input Low) quando externamente é aplicado um nível lógico 0.

• Para o 89S52 o valor máximo de IIL = - 50 uA

• Quando o Microcontrolador escreve nível lógico zero na saída, os Máximos Valores de IOL (Corrente de Output Low) para o 89S52 são:

Máxima IOL por pino: 10 mA

Máxima IOL para 8-bits: P0 = 26 mA

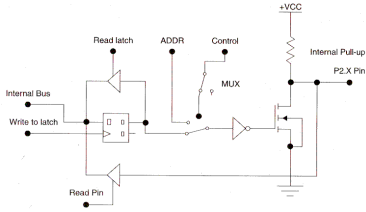
P1, P2 e P3: 15 mA

Máxima IOL - todos os pinos (P0, P1,P2,P3): 71 mA

• A máxima corrente de saída em nível alto (IOH = Corrente de Output High) para o 89S52 é = - 60 uA

### Arquitetura interna da Porta P2

- Porta Bi-direcional com pull-up



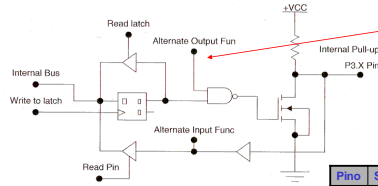
• Os valores de IOL, IOH, IIL e IIH são os mesmos da Porta P1

• Após um Reset, todas as saídas dos Latches vão para nível lógico 1, programando todas as Portas (P0, P1, P2, P3) como entradas.

• A Porta P2, como a P0, não pode ser usada como entrada/saída se estiver sendo usada como Barramento de Endereços.

### Arquitetura interna da Porta P3

- Porta Bi-direcional com pull-up



• Os valores de IOL, IOH, IIL e IIH são os mesmos da Porta P1

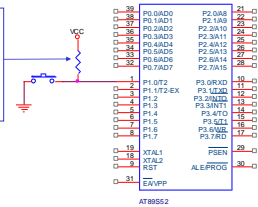
• Se o Latch da Porta P3 está em nível lógico 1, a saída é controlada pelo sinal de **Função de Saída Alternativa**

• Os Pinos da Porta P3 estão sempre disponíveis como Entrada das Funções Alternativas

Pino	Sentido	Função Alternativa
P3.0	Entrada	RxD (Entrada Serial)
P3.1	Saída	TxD (Saída Serial)
P3.2	Entrada	INT0 (Interrupção 0)
P3.3	Entrada	INT1 (Interrupção 1)
P3.4	Entrada	T0 (Contador 0)
P3.5	Entrada	T1 (Contador 1)
P3.6	Saída	WR (sinal de escrita)
P3.7	Saída	RD (Sinal de Escrita)

### Conectando Chaves mecânicas às entradas do Microcontrolador

É possível utilizar, mas não obrigatoriamente, um Resistor de **10K ohms** que consumirá uma corrente da fonte de aproximadamente 500 uA quando a chave for acionada.



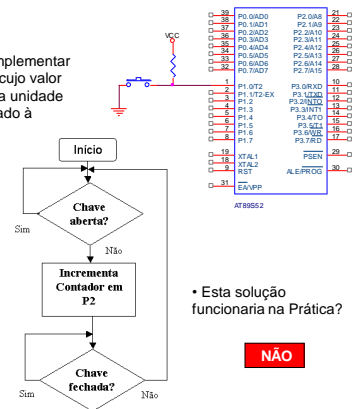
• É importante ressaltar que quando a chave for acionada, o resistor de Pull-up da porta fornece também no máximo IIL = 50 uA.

### Exemplo de Aplicação

• Considerando o esquema, implementar um contador binário de 8 Bits cujo valor deve ser incrementado de uma unidade a cada toque na chave e enviado à Porta P2

#### • Primeira Solução

• Testando-se a chave duas vezes verifica-se se o usuário já completou a operação de um toque, ou seja, um pulso completo.

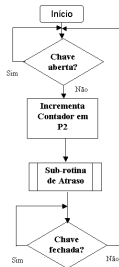
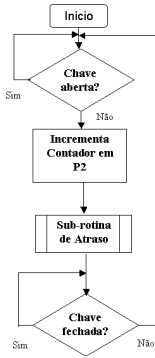
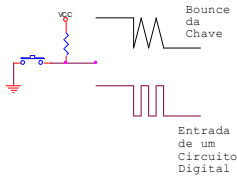


• Esta solução funcionaria na Prática?

**NÃO**

• As chaves mecânicas possuem inércia ao fechar que geram ruídos que poderão ser detectados pelo software ao se proceder a leitura da mesma.

• Este ruído, chamado de Bounce da chave deve ser eliminado, ou por hardware (capacitor, FF tipo D) ou por software inserindo um atraso antes da leitura da chave fechada.



```

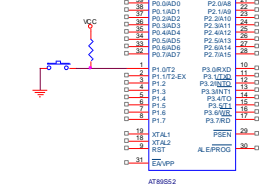
; Programa de debounce de chave mecânica
; Uma sub-rotina de atraso deve ser inserida para garantir o tempo de
; estabilização mecânica da chave. Para a maioria das chaves um atraso de
; 500 us é suficiente
;-----
ORG 0
CLR A           ; Inicializa o contador em zero
LOOP:          ; Testa se a chave está aberta
JB P1.0,$     ; Incrementa o contador
MOV P2,A     ; Mostra o valor do contador na Porta P2
LCALL ATRASO ; Realiza o debounce da chave
JNB P1.0,$   ; Verifica se a chave continua fechada
SJMP LOOP    ; Retorna para nova contagem de pulso
;-----

```

```

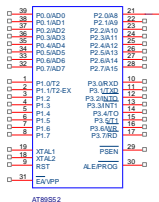
;-----
; Sub-rotina de atraso
; Valor do atraso para debounce da chave de
; aproximadamente 500 us
;-----
ATRASO:        MOV R0,#0FFH ; Valor do atraso para debounce da chave de
                DJNZ R0,$   ; aproximadamente 500 us
                RET
;-----

```



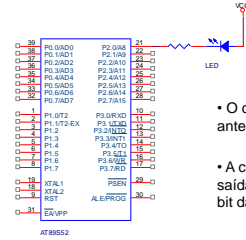
**Conectando Leds às saídas do Microcontrolador**

• Qual o valor do Resistor e da corrente necessária para acender o Led no esquema abaixo?



- O nível da porta para acionar o Led deve ser nível lógico 1, ou seja = 5v
- Um Led tem uma queda de tensão de 1,4 v
- Logo,  $V = RI + Vled \rightarrow R = (5 - 1,4) / I$

• E se alterarmos o esquema de conexão, qual o valor do Resistor e da corrente necessária para acender o Led ?

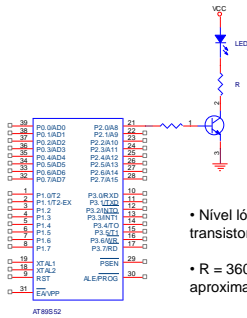


- O nível da porta para acionar o Led deve ser nível lógico 0, ou seja = 0v
- O cálculo do Resistor é o mesmo realizado anteriormente, ou seja,  $R = 360$  ohms
- A corrente a ser avaliada agora é a de nível de saída baixo, ou seja, IOL é no máximo 10 mA por bit da Porta P2.

- Como um Led necessita de aproximadamente 10 mA para uma boa luminosidade  $\rightarrow R = 3,6 / 10mA = 360$  ohms
- Como a corrente de saída em nível alto IOH é no máximo 60 uA, esta ligação é inviável

• O valor máximo de IOL para os 8 Bits da Porta P2 é 15 mA, ou seja, a solução é viável para um Bit mas não deve ser adotada para a porta toda, pois o nível de corrente ultrapassa o máximo recomendado.

### Bufferizando as saídas do Microcontrolador



• Utilizando-se um Buffer/driver com transistores, um para cada bit da porta, fornece-se a corrente necessária à operação do Led sem afetar os valores máximos permitidos para o Microcontrolador.

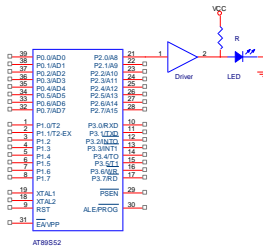
• Nível lógico 1 em P2.0 conduz o transistor acendendo o Led

• R = 360 Ohms fornece corrente de aproximadamente 10 mA ao Led.

• O resistor de base deve ser calculado conforme especificações do transistor com IOH máxima de 60 uA

### Bufferizando as saídas do Microcontrolador

• A solução mais indicada é utilizar um Circuito "Led Driver" que fornecerá a corrente necessária aos Leds em cada um dos Bits da Porta não permitindo ultrapassar os valores máximos recomendados.



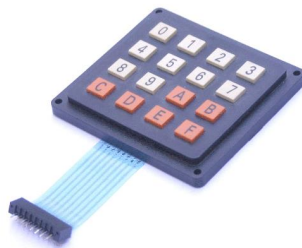
#### Alguns CI's de Led Driver

- 7407 (4 por chip)
- 74244 (8 por chip)
- ULN2803 (8 por chip)
- outros.....

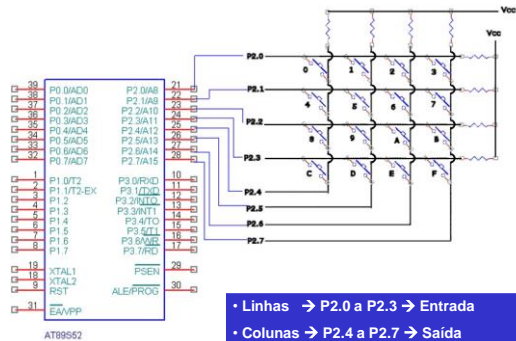
### Interface do Microcontrolador com Teclado Matricial

• Um Teclado Matricial de 4 linhas por 4 colunas permite a geração de 16 códigos independentes utilizando-se apenas 8 linhas de uma Porta do Microcontrolador.

• Dessa maneira é possível interfacear um Teclado Hexadecimal Matricial ao Microcontrolador para entrada de dados binários

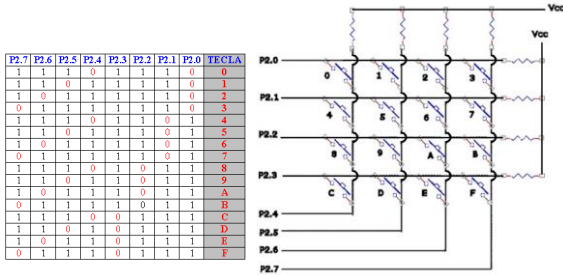


### Interface do Microcontrolador com Teclado Matricial



- Linhas → P2.0 a P2.3 → Entrada
- Colunas → P2.4 a P2.7 → Saida

Valores aplicados às colunas(P2.4 a P2.7) e lidos nas linhas (P2.0 a P2.3), se for acionada uma tecla presente na coluna que contém zero.



Caso nenhuma tecla seja acionada, o valor das linhas é = F

```

ORG 0
LOOP:
MOV P2,#0FFH ; Garante nível lógico 1 em todos os bits de P2
CLR P2.4 ; Zero a coluna das chaves 0,4,8,C
LOOP1:
JNB P2.0,ZERO ; Zero a linha das chaves 0,1,8,9
JNB P2.1,QUATRO ; Zero a linha das chaves 4,5,8,9
JNB P2.2,OITO ; Zero a linha das chaves 0,1,4,5,8,9
JNB P2.3,CCC ; Zero a linha das chaves 0,1,4,5,8,9
MOV A,#00H
CLR P2.5
LOOP2:
JNB P2.0,OH ; Zero a coluna das chaves 1,5,9,D
JNB P2.1,CINCO ; Zero a coluna das chaves 1,5,9,D
JNB P2.2,NOVE ; Zero a coluna das chaves 1,5,9,D
JNB P2.3,DOIS ; Zero a coluna das chaves 2,6,8,E
MOV A,#00H
CLR P2.6
LOOP3:
JNB P2.0,AAA ; Zero a coluna das chaves 2,6,8,E
JNB P2.1,SEIS ; Zero a coluna das chaves 2,6,8,E
JNB P2.2,DEZ ; Zero a coluna das chaves 2,6,8,E
JNB P2.3,TRÊS ; Zero a coluna das chaves 2,6,8,E
MOV A,#00H
CLR P2.7
LOOP4:
JNB P2.0,TRÊS ; Zero a coluna das chaves 3,7,9,F
JNB P2.1,SETE ; Zero a coluna das chaves 3,7,9,F
JNB P2.2,ONZE ; Zero a coluna das chaves 3,7,9,F
JNB P2.3,QUATRO ; Zero a coluna das chaves 3,7,9,F
MOV A,#00H
Sjmp LOOP
ZERO: MOV A,#00H
LOOP1: Sjmp LOOP1
OH: Sjmp LOOP2
QUATRO: MOV A,#02H
LOOP2: Sjmp LOOP2
OITO: MOV A,#04H
LOOP3: Sjmp LOOP3
CCC: MOV A,#06H
LOOP4: Sjmp LOOP4
NOVE: MOV A,#08H
LOOP1: Sjmp LOOP1
AAA: MOV A,#0AH
LOOP2: Sjmp LOOP2
SEIS: MOV A,#0CH
LOOP3: Sjmp LOOP3
OITO: MOV A,#0EH
LOOP4: Sjmp LOOP4
NOVE: MOV A,#0FH
LOOP1: Sjmp LOOP1
AAA: MOV A,#11H
LOOP2: Sjmp LOOP2
DEZ: MOV A,#13H
LOOP3: Sjmp LOOP3
TRÊS: MOV A,#15H
LOOP4: Sjmp LOOP4
QUATRO: MOV A,#17H
LOOP1: Sjmp LOOP1
NOVE: MOV A,#19H
LOOP2: Sjmp LOOP2
SEIS: MOV A,#1BH
LOOP3: Sjmp LOOP3
ONZE: MOV A,#1DH
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#1FH
LOOP1: Sjmp LOOP1
SETE: MOV A,#21H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#23H
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#25H
LOOP4: Sjmp LOOP4
DEZ: MOV A,#27H
LOOP1: Sjmp LOOP1
ONZE: MOV A,#29H
LOOP2: Sjmp LOOP2
TRÊS: MOV A,#2BH
LOOP3: Sjmp LOOP3
SETE: MOV A,#2DH
LOOP4: Sjmp LOOP4
ONZE: MOV A,#2FH
LOOP1: Sjmp LOOP1
TRÊS: MOV A,#31H
LOOP2: Sjmp LOOP2
SETE: MOV A,#33H
LOOP3: Sjmp LOOP3
ONZE: MOV A,#35H
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#37H
LOOP1: Sjmp LOOP1
SETE: MOV A,#39H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#3BH
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#3DH
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#3FH
LOOP1: Sjmp LOOP1
SETE: MOV A,#41H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#43H
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#45H
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#47H
LOOP1: Sjmp LOOP1
SETE: MOV A,#49H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#4BH
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#4DH
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#4FH
LOOP1: Sjmp LOOP1
SETE: MOV A,#51H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#53H
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#55H
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#57H
LOOP1: Sjmp LOOP1
SETE: MOV A,#59H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#5BH
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#5DH
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#5FH
LOOP1: Sjmp LOOP1
SETE: MOV A,#61H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#63H
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#65H
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#67H
LOOP1: Sjmp LOOP1
SETE: MOV A,#69H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#6BH
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#6DH
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#6FH
LOOP1: Sjmp LOOP1
SETE: MOV A,#71H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#73H
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#75H
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#77H
LOOP1: Sjmp LOOP1
SETE: MOV A,#79H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#7BH
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#7DH
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#7FH
LOOP1: Sjmp LOOP1
SETE: MOV A,#81H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#83H
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#85H
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#87H
LOOP1: Sjmp LOOP1
SETE: MOV A,#89H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#8BH
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#8DH
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#8FH
LOOP1: Sjmp LOOP1
SETE: MOV A,#91H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#93H
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#95H
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#97H
LOOP1: Sjmp LOOP1
SETE: MOV A,#99H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#9BH
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#9DH
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#9FH
LOOP1: Sjmp LOOP1
SETE: MOV A,#A1H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#A3H
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#A5H
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#A7H
LOOP1: Sjmp LOOP1
SETE: MOV A,#A9H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#ABH
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#ADH
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#AFH
LOOP1: Sjmp LOOP1
SETE: MOV A,#B1H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#B3H
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#B5H
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#B7H
LOOP1: Sjmp LOOP1
SETE: MOV A,#B9H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#BBH
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#BDH
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#BFH
LOOP1: Sjmp LOOP1
SETE: MOV A,#C1H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#C3H
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#C5H
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#C7H
LOOP1: Sjmp LOOP1
SETE: MOV A,#C9H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#CBH
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#CDH
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#CFH
LOOP1: Sjmp LOOP1
SETE: MOV A,#D1H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#D3H
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#D5H
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#D7H
LOOP1: Sjmp LOOP1
SETE: MOV A,#D9H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#DBH
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#DDH
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#DFH
LOOP1: Sjmp LOOP1
SETE: MOV A,#E1H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#E3H
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#E5H
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#E7H
LOOP1: Sjmp LOOP1
SETE: MOV A,#E9H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#EBH
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#EDH
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#EFH
LOOP1: Sjmp LOOP1
SETE: MOV A,#F1H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#F3H
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#F5H
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#F7H
LOOP1: Sjmp LOOP1
SETE: MOV A,#F9H
LOOP2: Sjmp LOOP2
ONZE: MOV A,#FBH
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#FDH
LOOP4: Sjmp LOOP4
TRÊS: MOV A,#FFH
LOOP1: Sjmp LOOP1
SETE: MOV A,#FFH
LOOP2: Sjmp LOOP2
ONZE: MOV A,#FFH
LOOP3: Sjmp LOOP3
QUATRO: MOV A,#FFH
LOOP4: Sjmp LOOP4

```

### Exercício 5

•Entrega pelo Site do Curso até o dia 05/06/2017

Fazer um programa em linguagem C para o 8051 que interfaceado a um Teclado Matricial, conforme esquema mostrado nessa aula, comande um Motor de Passo conectado à Porta P1 (P1.0 = Clock, P1.1 = Direção).

- O Número de passos (de 01 a 99) deve entrar pelo Teclado Matricial seguido da direção ( A = Anti-horária e B = Horária). Após os dados de número de passos e direção terem sido inseridos o programa aguarda uma tecla de início (usar a letra F).
- Quando o Motor executar os passos solicitados no item 1 deve parar e aguardar novos comandos de Número de Passos e Direção.
- Simular no MCU8051 colocando um contador em paralelo com o clock do motor para verificar a contagem dos pulsos.
- Letra C – corrige o campo atual
- Letra D – repete sequência pré-gravada
- Letra E – Limpa memória