

Comunicando com um teclado PS2

Vitor Amadeu Souza – vitor@cerne-tec.com.br

Introdução

Os teclados são corriqueiramente utilizados nos dias de hoje no mundo dos PCs. Neste artigo, irei apresentar ao leitor como é feita a comunicação entre o teclado e um PC e apresentar uma aplicação onde o teclado poderá controlar o estado de um relé, comunicando desta forma com um microcontrolador PIC programado através do mikroBASIC.

A comunicação com um PC

Existem dois conectores de teclado, sendo um chamado de DIN e outro de mini-DIN. O primeiro pode ser visto na figura 1 e o segundo na figura 2.



Figura 1 – Conector DIN



Figura 2 – Conector mini-DIN

O conector DIN é caracterizado por conter 5 pinos e ser maior em relação ao mini-DIN que possui 6 pinos. Neste artigo, irei explorar o teclado que utilizado o conector mini-DIN, por ser o mais usual nos dias de hoje. Este tipo de conector também é chamado de PS2. A configuração dos pinos deste conector pode ser visualizada na figura 3.

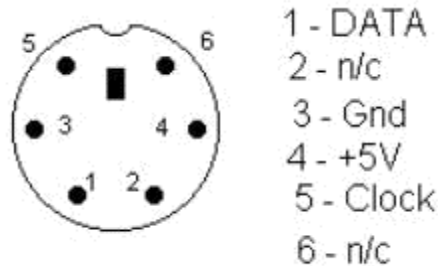


Figura 3 – Pinos do conector PS2

Observe que dos 6 pinos deste conector, somente 4 são utilizados sendo 2 para alimentação e 2 para comunicação. A comunicação utilizada é síncrona e funciona de 10 kHz a 16kHz no nível TTL com 1 bit de start, 8 de dados, 1 de paridade e 1 de stop. O elemento que gera o sincronismo na rede é sempre o teclado. Verifique na figura 4 um gráfico apresentando a comunicação do dado 10011100. Veja que a soma de todos os bits é par e como a paridade utilizada é ímpar, este bit no protocolo de comunicação fica em nível alto para que toda a soma agora resulte em um número ímpar. Note também que a comunicação sempre começa pelo bit LSB do dado a ser transmitido.

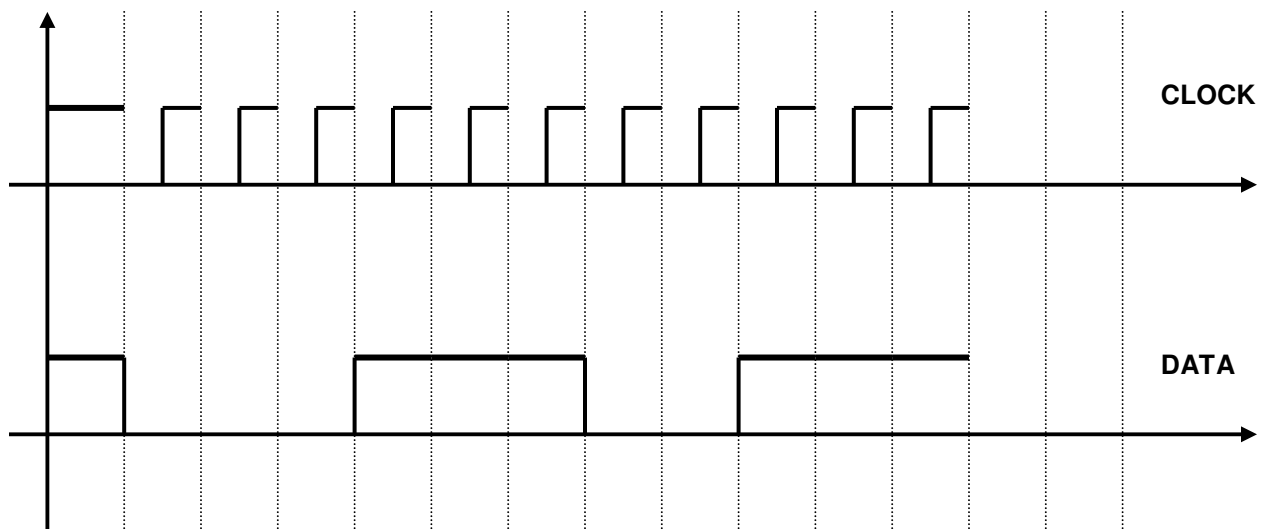


Figura 4 – Envio de Dados

Observe que a linha de dados somente muda de estado enquanto a linha de clock permanece em nível alto. Isto ocorre porque a transferência de dados é feita quando a linha de clock está em nível baixo.

Recursos de Hardware

O esquema elétrico para execução deste artigo está apresentado na figura 1. Note que apesar de poucos pinos utilizados do microcontrolador para este exemplo, o PIC18F452 foi utilizado pois podemos expandir mais este sistema, colocando por exemplo display gráfico, LCD ou comunicação serial. A lista de material pode ser observada na tabela 1.

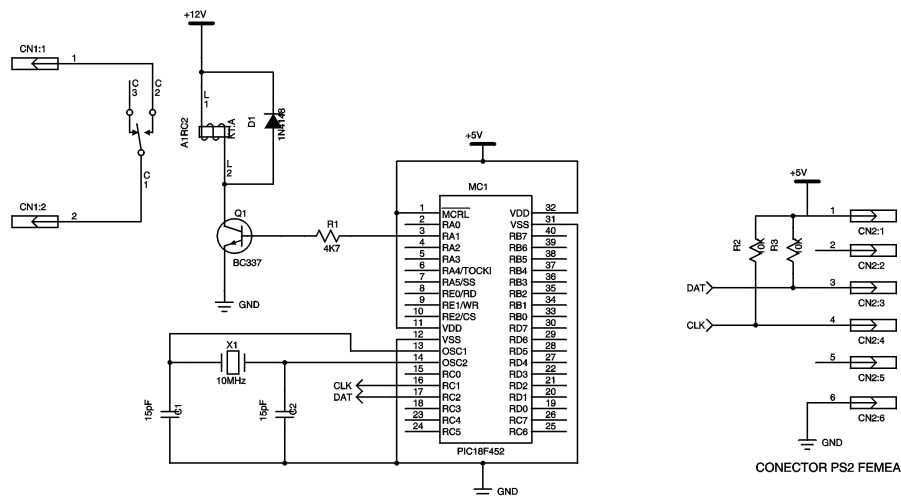


Figura 1 – Esquema Elétrico

Quantidade	Componente	Referência
2	Capacitor Cerâmico	C1, C2
1	Conector KRE02	CN1
1	Conector de teclado PS2 fêmea	CN2
1	Cristal de 10 MHz	X1
1	Transistor NPN BC337	Q1
1	Microcontrolador PIC18F452	MC1
1	Relé 12 V A1RC2	K1
1	Resistor 4K7 ¼ W	R1
1	Resistor 10K ¼ W	R2, R3

Tabela 1 – Lista de Material

Recursos de Software

O mikroBASIC disponibiliza duas funções que permitem a comunicação com este teclado. Observe a tabela 2.

```
ps2_config (PORTA, CLOCK, DATA)
```

```
ps2_key_read (VALOR, ESPECIAL, PRESSIONADO)
```

Tabela 2 – Funções de comunicação com o teclado

A primeira função é utilizada para configurar a ligação dos pinos de dados e de clock do teclado ao microcontrolador. Através de PORTA devemos informar em que porta do microcontrolador está ligado os pinos de DATA e CLOCK do teclado e através de CLOCK e DATA informar em que pino do PORT os mesmos encontram-se conectados. De acordo com o esquema elétrico apresentado na figura 1, podemos observar que o PORT onde está ligado o teclado é o PORTC e o pino de clock é o RC1 e o de dados o RC2. Desta forma, este comando ficaria como o apresentado no box 1.

```
ps2_config (PORTC, 1, 2) 'Configura os pinos de comunicação com o teclado
```

Box 1 – Configuração do teclado

Já o comando ps2_key_read (VALOR, ESPECIAL, PRESSIONADO) tem a função de verificar se alguma tecla foi pressionada e caso esteja, retornar o valor da tecla pressionada. Em valor, é retornado o caracter pressionado de acordo com a tabela ASCII caso seja pressionado algum caracter, número, pontuação e espaço. A tabela ASCII pode ser observada na tabela 3.

0	.	32	[space]	64	@	96	`
1	.	33	!	65	A	97	a
2	.	34	"	66	B	98	b
3	.	35	#	67	C	99	c
4	.	36	\$	68	D	100	d
5	.	37	%	69	E	101	e
6	.	38	&	70	F	102	f
7	.	39	'	71	G	103	G
8	**	40	(72	H	104	H
9	**	41)	73	I	105	I
10	**	42	*	74	J	106	j
11	.	43	+	75	K	107	kl
12	.	44	,	76	L	108	l
13	**	45	-	77	M	109	m
14	.	46	.	78	N	110	n
15	.	47	/	79	O	111	o
16	.	48	0	80	P	112	p
17	.	49	1	81	Q	113	q
18	.	50	2	82	R	114	r
19	.	51	3	83	S	115	s
20	.	52	4	84	T	116	t

21	.	53	5	85	U	117	u
22	.	54	6	86	V	118	v
23	.	55	7	87	W	119	w
24	.	56	8	88	X	120	x
25	.	57	9	89	Y	121	y
26	.	58	:	90	Z	122	z
27	.	59	;	91	[123	{
28	.	60	<	92	\	124	
29	.	61	=	93]	125	}
30	.	62	>	94	^	126	~
31	.	63	?	95	_	127	.

Tabela 3 – Tabela ASCII

Caso algum caracter diferente desses seja pressionado, como por exemplo o F1, o valor retornado será de acordo com o apresentado na tabela 4.

Tecla	Valor retornado
F1	1
F2	2
F3	3
F4	4
F5	5
F6	6
F7	7
F8	8
F9	9
F10	10
F11	11
F12	12
Enter	13
Page Up	14
Page Down	15
Backspace	16
Insert	17
Delete	18
Windows	19
Ctrl	20
Shift	21
Alt	22
Print Screen	23
Pause	24
Caps Lock	25
End	26

Home	27
Scroll Lock	28
Num Lock	29
Left Arrow	30
Right Arrow	31
Up Arrow	32
Down Arrow	33
Escape	34
Tab	35

Tabela 4 – Código retornado para caracteres especiais

Já o parâmetro ESPECIAL informa se algum caracter especial foi pressionado. Caso ESPECIAL esteja em 1, significa que a tabela 3 deve ser seguida e caso contrário, a tabela ASCII convencional. Através de PRESSIONADO, poderemos saber se a tecla ainda está pressionada ou a mesma foi pressionada novamente (atualizada). Caso este parâmetro retorne 1, significa que continua pressionada e caso retorne 0, a mesma foi atualizada.

Fluxograma

O fluxograma que irá reger o funcionamento deste exemplo está apresentado na figura 2. Note logo destarte os pinos de I/O do PORTA são configurados para funcionar em modo digital e o teclado é inicializado. Após esta tarefa, o fluxo entra em um laço em que constantemente é verificado se alguma tecla foi pressionada. Caso favorável, é checado se o mesmo é o caracter “a” e caso verdadeiro, o relé conectado no pino RA1 é ligado e caso qualquer caracter diferente deste seja recebido, o relé é desligado.

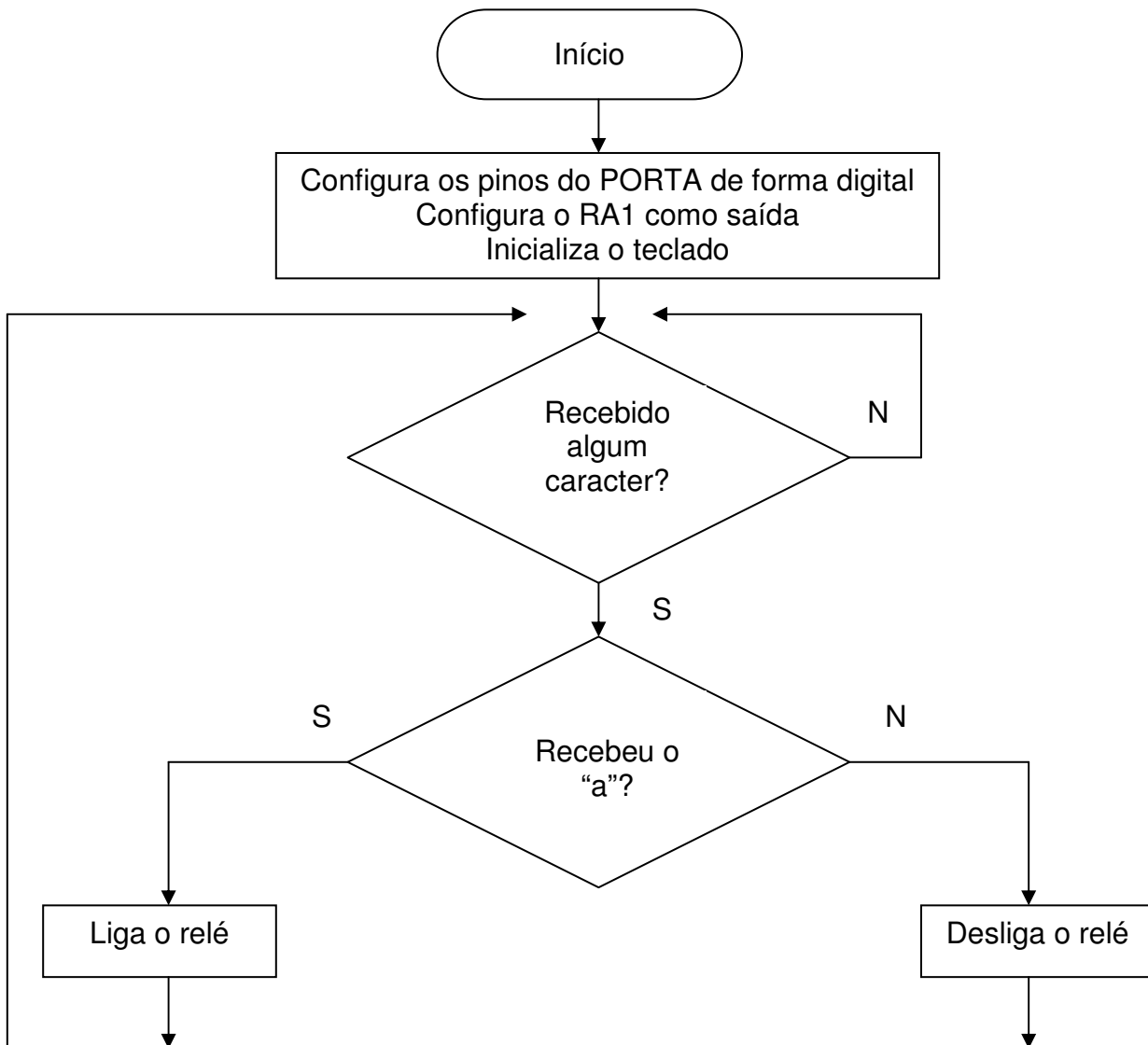


Figura 2 – Fluxograma do exemplo

Programa

Primeiramente você deve criar um projeto no mikroBASIC para execução deste exemplo. Os passos são os mesmos do apresentado no artigo que relata a utilização de um display gráfico, mudando apenas o microcontrolador utilizado, que neste caso é o PIC18F452 e o configuration bit, que deve seguir a forma que está apresentada na tabela 5. Note que o restante que não é mencionado é irrelevante para este projeto.

Configuration Bit	Descrição
HSPLL_OSC_1H	Seleciona a fonte de clock para o tipo cristal com PLL ativa, ou seja, ele irá multiplicar a frequência de entrada por 4 e processar internamente na mesma frequência do cristal que neste caso é de 10 MHz.
WDT_OFF_2H	Desliga o Watchdog
LVP_OFF_4L	Desliga a gravação em baixa tensão
DEBUG_OFF_4L	Desliga o modo de debugação

Tabela 5 – Ajuste dos Configurations bits

Como a PLL deste exemplo ficará ativa, internamente o microcontrolador irá funcionar a 40 MHz e processar 10 MIPS. Para finalizar, informe no parâmetro clock o valor de 40000000 (40 MHz). Feito isto, pressione *ok*.

O código completo para execução deste exemplo está apresentado no box 2.

```

program TecladoPS2

Main:
  dim keydata, special, down as byte
                                'Define as variáveis auxiliares da função PS2_Key_Read
  ADCON1=7                       'Desliga os ADs do pino do PORTA
  trisa.1=0                       'Configura o RA1 como saída
  Ps2_config(PORTC,1,2)          'Configura os pinos de CLOCK e DATA do teclado

  while true                       'Loop While

    if Ps2_Key_Read(keydata, special, down)=1 then
      if keydata="a" then          'Tem algum dado no buffer?
        porta.1=1                 'Sim! É o "a"?
                                  'Sim! Então liga o relé
      else
        porta.1=0                 'Não... Então desliga o relé
      end if
    end if

  wend

end.

```


Box 2 – Código completo do exercício

Vamos entender melhor o funcionamento deste código. Primeiramente, o código inicia com *program TecladoPS2* que foi o nome que eu dei na definição do projeto. Fique à vontade em escolher o nome que você achar melhor. Logo em seguida, vemos a declaração de um label chamado *main*. Note que todos os labels no mikroBASIC são terminados com : (dois pontos). Em seguida, são declaradas três variáveis do tipo byte, chamadas *keydata*, *special* e *down*. Estas variáveis serão utilizadas pelo comando de leitura do teclado *ps2_key_read*. Nos pinos RA0, RA1, RA2, RA3, RA5, RE0, RE1 e RE2 são os pinos de AD do microcontrolador. Este microcontrolador “acorda” deixando estes pinos configurados para a função analógica porém como no nosso exemplo vamos controlar um relé que é uma informação digital (ligado e desligado) precisamos desligar a função analógica e configurar como digital. Isto é feito atribuindo ao registrador ADCON1 o valor 7. O pino em que está conectado o relé também necessita ser configurado como saída, já que seu estado inicial é como entrada. Através de *trisa.1=0* configuramos este pino como saída. Finalmente o teclado é inicializado, passando como parâmetro neste caso o PORT onde o mesmo está conectado e os pinos do PORT utilizados como CLOCK e DATA. Agora o programa entra no loop *while* que faz o código escrito entre *while* e *wend* ser repetido indefinidamente, já que a condição passada para o *while* (*enquanto*) é verdadeira. A função *ps2_key_read* é chamada e caso a mesma retorne 1, indica que existe um caracter no buffer. Dentro do *if* que testa a existência do caracter, é verificado se o mesmo é o “a” e caso seja, o relé é acionado e caso qualquer caracter diferente do “a” seja recebido, o relé será desligado.

Compile este projeto e transfira o arquivo hex para o microcontrolador. Este exemplo pode ser montado em uma placa padrão, proto-board ou na placa didática PIC MASTER desenvolvida pela Cerne Tecnologia (www.cerne-tec.com.br).

Conclusão

Uma aplicação a mais onde pode ser utilizado um microcontrolador foi apresentado neste artigo. Desta forma, pudemos observar que o universo de utilização dos teclados não está restrita ao mundo dos PCs, podendo também ser utilizado em aplicações em que seja necessário algum ajuste de parâmetro, como por exemplo um taxímetro, onde que seja necessário alterar o valor da bandeirada ou fração esta interface pode ser feita diretamente com um teclado do tipo PS2. A grande vantagem da utilização dos teclados é o baixo custo no qual encontramos

estes hoje no mercado, tornado o seu uso bastante favorável em projetos que necessitem de algum ajuste de parâmetro.