

MSX Article

MARMSX

Joynet Básica

Resumo

O objetivo deste artigo é demonstrar como é possível construir uma rede alternativa para MSX, através dos conectores de joystick do micro.

1- Introdução

A comunicação entre computadores envolve diversos benefícios. Com ela, é possível trocar arquivos entre máquinas, compartilhar programas, jogar uma partida de um jogo com outras pessoas etc.

O MSX possui alguns dispositivos de comunicação como a interface RS-232 e modernas placas de rede ethernet [1][2]. Entretanto, a rede *joynet* surgiu a partir de uma lista de discussão internacional de MSX [3][4].

A concepção da rede *joynet* seguiu o modelo criado pela Konami para o jogo F1 Sircuit 3D. Entretanto, não foi definido qualquer protocolo de comunicação ou trabalho para que pudesse servir de referência para a rede *joynet*.

A proposta deste artigo é mostrar a confecção de cabos *joynet* MSX-MSX e MSX-PC, previamente desenhados e como utilizar a porta do joystick para a comunicação entre MSXs.

2- O cabo da rede *joynet*

Foram criadas duas soluções para conexão via *joynet*: um cabo para ligar dois MSXs via porta de joystick e outro cabo para ligar um MSX ao PC via joystick e porta paralela.

A construção de um cabo *joynet* é extremamente simples e barata. O custo total dos componentes do cabo fica em torno de R\$ 10,00¹.

2.1 - O cabo MSX-MSX

O material necessário para se construir um cabo MSX-MSX [3] é o seguinte:

- 2 conectores DB-9 fêmeas.
- 2 caixas protetoras para os DB-9.
- 7 fios de telefone (diâmetro aproximado de 25 mm) com 1,5 m de comprimento.
- Ferro para soldar.

A numeração de 1 a 9 normalmente vem impressa no conector DB-9. A figura 1 apresenta essa numeração, que é vista pelo lado da solda.

1 Preço de 2003, época que o artigo foi escrito.

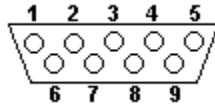


Figura 1. Conexão fêmea do conector DB-9 pelo lado da solda.

A ligação entre os pinos do primeiro conector DB-9 com o segundo conector, é feita de acordo com a tabela 1. Por exemplo, um fio deve ser soldado com uma ponta no pino 1 do conector 1 e com outra ponta no pino 6 do conector 2.

Conector 1	Conector 2
1	6
2	7
3	8
6	1
7	2
8	3
9	9

Tabela 1. Conexão MSX-MSX.

O mapa da fiação entre os conectores, relativo à tabela 1, é apresentado na figura 2.

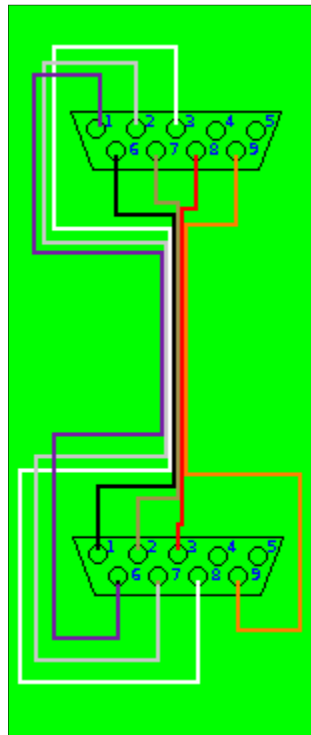


Figura 2. Conexão MSX-MSX.

2.2 - O cabo MSX-PC

O material necessário para se construir um cabo MSX-PC [3] é o seguinte:

- 1 conector DB-9 fêmea.
- 1 conector DB-25 macho.
- 1 caixa protetora para o DB-9 e outra para o DB-25.
- 7 fios de telefone (diâmetro aproximado de 25 mm) com 1,5 m de comprimento.
- Ferro para soldar.

O DB-25 é apresentado na figura 3. Assim como o DB-9, a imagem representa o lado da solda.

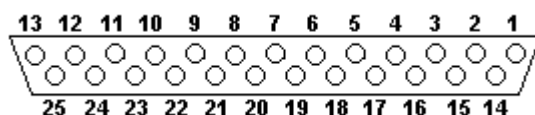


Figura 3. Conexão macho do conector DB-25 pelo lado da solda.

A ligação entre os pinos do conector do MSX com o conector do PC será de acordo com a tabela 2. O esquema de fios é apresentado na figura 4.

DB-9	DB-25
1	2
2	3
3	4
6	13
7	12
8	10
9	18 ~ 25

Tabela 2. Conexão MSX-PC.

O Pino 9 do DB-9 deverá ser ligado a um dos seguintes pinos do DB-25: 18, 19, 20, 21, 22, 23, 24 ou 25. Estes pinos correspondem ao terra do PC.

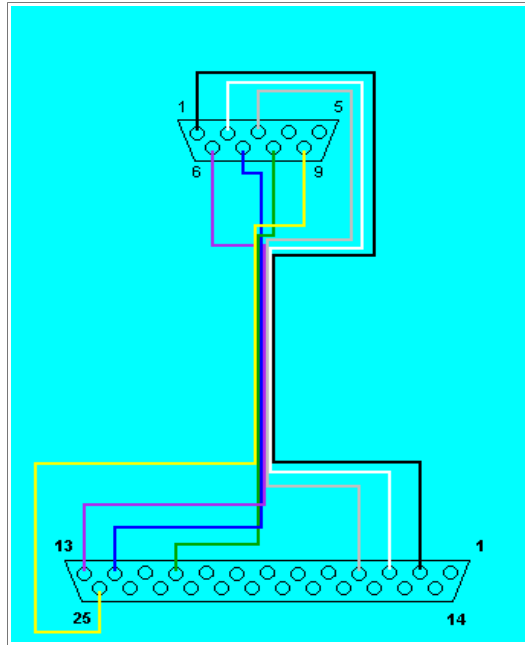


Figura 4. Conexão MSX-PC.

2.3 - A Porta de joystick do MSX

O MSX contém, normalmente, duas portas de joystick. As portas são compostas de conectores padrão DB-9 macho, onde a configuração de pinos vistos de frente (lado do encaixe) é a mesma da figura 1. Quando um joystick é conectado a esta porta, os pinos delas se comportam conforme a tabela 3.

Pino	Função	Sinal
1	Para cima	recebe
2	Para baixo	recebe
3	Esquerda	recebe
4	Direita	recebe
5	+ 5 Volts CC	-
6	Botão 1 de tiro	ambos
7	Botão 2 de tiro	ambos
8	Saída	envia
9	GND (terra)	-

Tabela 3. Funções do conector DB-9 do MSX.

O joystick é um circuito simples, que recebe 5 Volts pelo pino 5 e possui diversas chaves que habilitam/desabilitam o sinal para os pinos 1, 2, 3, 4, 6 e 7.

O circuito integrado responsável pelo controle das portas de joystick é o PSG AY-3-8910-A. A sigla PSG vem de *Programmable Sound Generator* ou gerador de sons

programável. Com isso, concluímos que o processador de sons do MSX é o responsável por controlar as portas de joystick.

O PSG tem 15 registradores e se comunica com o processador Z80-A através de portas. Somente 2 registradores são utilizados para este fim: 14 e 15. As portas de acesso ao PSG são as portas &HA0, &HA1 e &HA2. A porta &HA0 é a responsável por escolher o registro atual do PSG, onde dados serão escritos pela porta &HA1 e lidos pela porta &HA2.

Configuração do registrador 14 [5]:

Bit	7	6	5	4	3	2	1	0
Descrição	Entrada de Cassete	Modo de Teclado	Botão 2	Botão 1	Direita	Esquerda	Abaixo	Acima

Configuração do registrador 15 [5]:

Bit	7	6	5	4	3	2	1	0
Descrição	Kana LED	Seleção de Joystick	Pulso 2	Pulso 1	1	1	1	1

As regiões sombreadas indicam que estes bits não serão utilizados pela *joynet*.

2.4 – A descrição dos registradores 14 e 15 do PSG

O registrador 14 utiliza a porta A do PSG, que é a entrada de dados. Ele tem apenas 6 bits que interessam a porta de joystick, que são os bits de 0 a 5. Todos eles estão relacionados ao estado dos botões do joystick. O valor igual a 1 indica que o respectivo botão/direcional está solto, ou seja, não há corrente chegando ao PSG, enquanto que o valor 0 indica que o respectivo botão/direcional está sendo pressionado.

O registrador 15 utiliza a porta B do PSG, que é a porta de saída de dados. Ele tem 7 bits de interesse, que vai de 0 a 6. O bit 6 seleciona qual o joystick em uso, e o valor igual a 0 indica o joystick 1, enquanto que o valor igual a 1 indica o joystick 2. Os outros bits enviam sinais para os pinos 6, 7 e 8 da porta de joystick, como será descrito a seguir.

Os dois pulsos de bits (4 e 5) são utilizados para gerar uma emissão de pulso curto a qualquer paddle ligado a uma das entradas de joystick do MSX [5]. Eles serão utilizados para enviar dados ao conector ligado ao outro MSX. Então, o sinal é enviado através do pino 8, de forma que se o bit 4 estiver em 0, nada é enviado, e se estiver em 1, o pulso é enviado.

Os bits 0 e 1 do registrador 15 indicam qual pino emissor deverá emitir um sinal contínuo. O bit 0 corresponde ao pino 6 e o bit 1 ao pino 7. Ambos podem enviar dados ao mesmo tempo. A tabela 4 apresenta as diversas possibilidades de se um enviar sinal pela porta de joystick.

Registrador 15								Pino 6	Pino 7	Pino 8
7	6	5	4	3	2	1	0			
X	X	0	0	1	1	1	1	0	0	VCC
X	X	0	1	1	1	1	1	0	0	PULSO
X	X	0	0	1	1	1	0	VCC	0	VCC
X	X	0	1	1	1	1	0	VCC	0	PULSO
X	X	0	0	1	1	0	1	0	VCC	VCC
X	X	0	1	1	1	0	1	0	VCC	PULSO
X	X	0	0	1	1	0	0	VCC	VCC	VCC
X	X	0	1	1	1	0	0	VCC	VCC	PULSO

Tabela 4. Possíveis sinais emitidos a partir do joystick do MSX.

O valor XX001111 é devolvido ao registrador 15 após cada operação. Portanto, para se manter algum pino aceso (6 ou 7), é necessário criar um loop que altere constantemente o registrador 15 para o valor desejado, de acordo com a tabela 4.

2.5 - Manipulando dados nos Registradores do PSG

O primeiro passo para se manipular um registrador do PSG, seja na leitura ou na escrita, é indicá-lo à porta &HA0. Então, o seguinte código em linguagem de máquina deve ser feito:

```
LD A,&H0F      ; Para escolher o registrador 15
OUT (&HA0),A
```

Ou

```
LD A,&H0E      ; Para escolher o registrador 14.
OUT (&HA0),A
```

Sempre quando se for utilizar um registrador diferente do atual, os comandos acima deverão ser utilizados.

O passo seguinte é ler ou escrever em um registrador. Para escrever, será usada a porta &HA1 e para ler, a porta &HA2.

Quando o programador desejar alterar apenas determinados bits de um registrador, deverá tomar cuidado em preservar os valores dos outros bits. Nesse caso, deverá utilizar uma das operações lógicas E e OU, de acordo com o caso.

A operação E (AND) é utilizada para zerar determinados bits e preservar os demais. Já a operação OU (OR) é utilizada para setar determinados bits e preservar os demais. Em ambos os casos, uma máscara irá indicar quais bits serão modificados e quais bits serão preservados. Na máscara da operação E, o valor igual a 0 modifica, enquanto que o valor igual a 1 preserva. Já a máscara da operação OU, o valor igual a 1 modifica, enquanto que o valor igual a 0 preserva.

Para exemplificar, suponha que desejemos alterar somente o bit 3 do registrador 15 para o valor igual a 0. Primeiramente, deve-se ler o conteúdo desse registrador. Em seguida, aplicar a operação E (zerar) do valor com a máscara apropriada. Em seguida, enviar o novo valor para o registrador 15.

```
REG_15 = &B00001111
máscara = &B11110111
-----
REG_15 = &B00000111
```

E como fazer para setar o bit 3 do mesmo registrador?

```
REG_15 = &B00000011
máscara = &B00001000
-----
REG_15 = &B00001011
```

Estes cuidados são indispensáveis para a modificação dos registradores. Se quisermos alterar o valor de algum registrador, teremos que ler ele e usar a operação lógica adequada, de modo a preservar os demais bits.

O programa em Assembly a seguir é um exemplo mais completo, onde seleciona o joystick 1 e recebe o estado dos botões/cursos no registrador A.

```
LD A,&H0F
OUT (&HA0),A ; Escolhe o registrador 15
IN A,(&HA2) ; Lê o registrador 15, passando-o para A
AND &HBF ; Faz operação A AND x0xxxxxx
OUT (&HA1),A ; Escreve no registrador 15
; Aqui foi selecionado o joystick 1

LD A,&H0E
OUT (&HA0),A ; Escolhe o registrador 14
IN A,(&HA2) ; A contém os dados da porta 1 do joystick
```

2.6 - Testando se o cabo está ligado a outro MSX

O teste a seguir é um teste básico, porém necessário, para fazer verificações periódicas se o cabo está conectado a outro MSX. Dado que o pino 8 do emissor envia um sinal de corrente para o pino 3 do receptor, faz-se a verificação do estado do bit 2 do registrador 14 do lado do receptor. Para utilizar esse teste, o pino 8 não deve enviar pulsos.

O programa em Assembly que faz essa verificação é descrito a seguir. A resposta é escrita no endereço de memória do MSX igual a &HC110.

```
LD A,&H0E ; Seleciona registro 14
OUT (&HA0),A
LD A,1
LD (&HC110),A ; Coloca flag de ligado como default em &HC110
CHECK: IN A,(&HA2)
BIT 2,A ; Verifica sinal no bit 2 (esquerda)
JR Z,END ; Se ligado, termina
XOR A
LD (&HC110),A ; Senão, indica que está desligado
END: RET
```


3 - Créditos e Referências

Este artigo foi escrito por Marcelo Teixeira Silveira, engenheiro de sistemas e computação, formado pela UERJ.

Escrito em: Dezembro de 2003.
Revisado em Junho de 2017.
e-mail: flamar98@hotmail.com
homepage: <http://marmsx.msxall.com>

Referências:

- [1] – ObsoNET, Daniel Berdugo. Em <http://www.konamiman.com/msx/msx-e.html#obsonet>
- [2] – MSX Ethernet Cartridge, Tecnobytes. Em <http://www.tecnobytes.com.br/2015/09/msx-ethernet-cartridge-available.html>
- [3] – Werner Kai em <http://www.msxpro.com/mirror/werner/joynetpo.htm>
- [4] – International MSX Mailing List, desativada. Correspondente ativa: <http://www.msx.org>
- [5] – O Livro Vermelho do MSX, Avalon Software, editora McGraw Hill.