

Itautec

RPLY

CODIGO DO DOCUMENTO

08599-01-006

REVISÃO

01

Documentação de Produto

DATA

20/03/84

FL

01/21

OBJETO

PCI MONT. RACK-INTERFACE SERIAL 8251X8

PROJETO

DATA

APROVAÇÃO

DATA

OBSERVAÇÕES

15008.01. X
nova placa

REPRODUÇÃO PROIBIDA

			/	/
			/	/
			/	/
01	PRIMEIRA VERSÃO		/	/
NR	REVISÃO	APROVAÇÃO		

DESCRIÇÃO FUNCIONALI - Introdução

A placa "Rack Interface Serial 51x8" é feita baseada no Bus Padrão Itautec (antigo Bus Terminal Caixa) e no padrão de dimensões e disposições de conectores do Rack-Padrão Itautec.

Conta com até 8 chips de interface serial síncrona ou assíncrona 8251A, um eficiente esquema de entrada/saída baseado em interrupção e coloca os sinais Tx C, Rx C, CTS, RTS, DSR, Tx D e Rx D da 8251A no conector para as placas de Drivers. Essas características fazem com que a placa fique bastante versátil, em condições de se obter um bom "casamento" com vários tipos de drivers e várias aplicações.

Esta descrição funcional será dividida nas seguintes partes:

- Esquema de atendimento de interrupção.
- Interface 8251A / Placa(s) de drivers.
- Jumper de configuração das placas.

Obs: Ao leitor não familiarizado com o chip 8251 é interessante a leitura das especificações do mesmo antes de prosseguir na leitura desta descrição funcional.

Itautec

Documentação de Produto

OBJETO

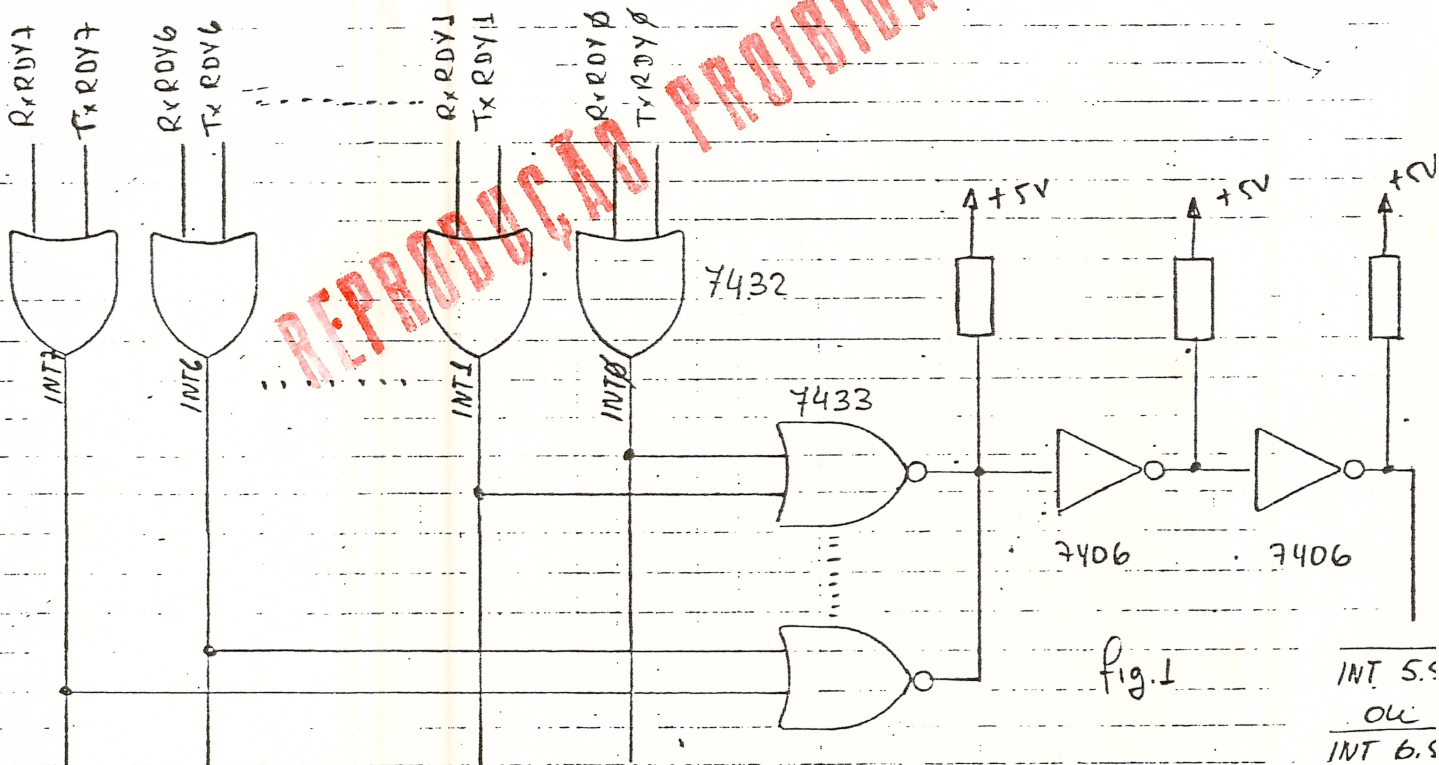
PCI MONT RACK-INTERFACE SERIAL 8251X8

CÓDIGO DO DOCUMENTO	REVISÃO
08599-01-006	01
DATA	FL
20/03/84	03/21

II- Esquema de atendimento de interrupções.

A interface serial 8251A oferece a facilidade de sinalizar ao processador quando este pode ler ou escrever um caractere (byte) na mesma (através dos pinos RxRDY e TxRDY).

A lógica abaixo gera o pedido de interrupção ao processador, através da sinalização das 8251A.



Qualquer um dos TxRDY e/ou RxRDY ativados (nível lógico "1") ativa o pedido de interrupção (nível lógico "0"). Esse pedido pode chegar à CPU através do $\overline{INT5.5}$ ou $\overline{INT6.5}$.

Uma vez interrompido o processamento, o processador precisa identificar quem gerou o pedido de interrupção e o

Para identificar quem gerou o pedido de interrupção, o processo é o seguinte:

- o nível de interrupção ($\overline{INT6.5}$ ou $\overline{INT5.5}$) deve indicar que uma placa I/O serial está requisitando serviço do processador.

A partir daí, o circuito abaixo procederá à identificação do chip mais prioritário que necessita serviço, mediante a execução de duas instruções pelo processador:

IN NUMR51
CMA

onde NUMR51 EQU 11111011B ou 01111011B
seja utilizado A10 ou A15, como vemos adiante (JP41) conforme

Obs: É executado um CMA porque o hardware fornece o n.º do chip complementado.

O circuito da figura 2 utiliza as linhas (senais) $\overline{INT0}$ a $\overline{INT7}$, geradas pelo circuito da figura 1. (Obs: ao leitor não familiarizado com o chip 74LS348, recomenda-se a leitura do "data sheet" do mesmo, que se encontra nos manuais TTL de vários fabricantes; antes de prosseguir na leitura desta descrição funcional).

Os sinais $\overline{INT0}$ a $\overline{INT7}$ significam pedido de serviço de um dos 8 chips 8251A da placa.

Na execução de "IN NUMR51", os sinais $\overline{IOR}$ (que também podem ser $\overline{IO/M}$, conforme vemos adiante - JP40) e

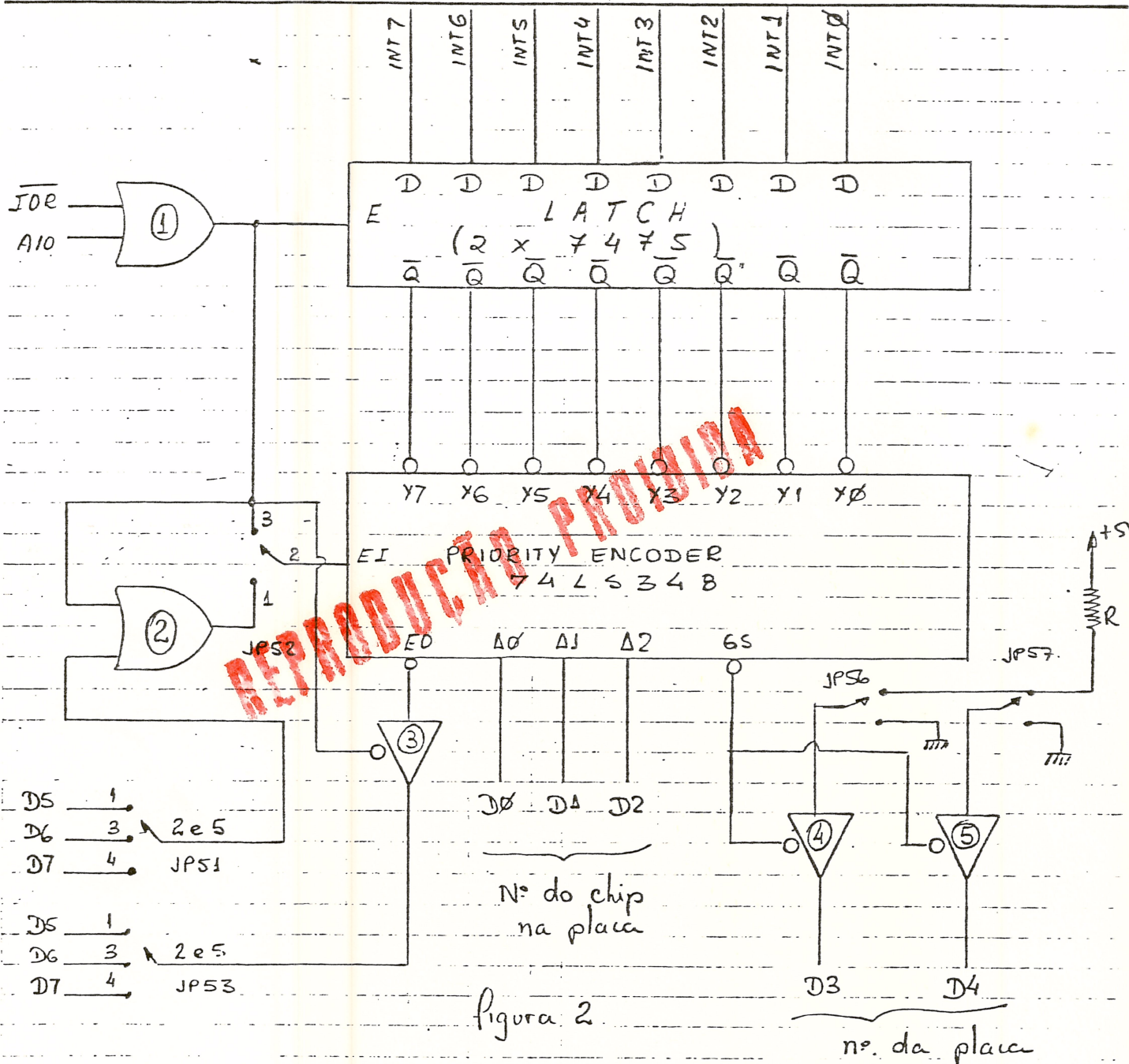


Figura 2

nº. da placa

A10 (que também poderia ser A15, conforme vemos adiante - JP41) terá nível lógico "0", fazendo com que o sinal E tenha o latch (Se houver mais de uma placa no sistema, os latches são ativados em todas as placas ao mesmo tempo, nesse instante. Isso garante que o nº do chip a ser codificado o seja a partir de um estado estável.) Este sinal também ativa, em todas as placas, a not. correspondente à nota (3) da fig 2.

Na primeira placa (placa mais prioritária), JP52 tem a ligação 3-2. Neste caso, o sinal que trava o latch também ativa o priority encoder desta placa (neste instante, só o priority encoder da primeira placa está ativo).

Se um dos sinais INT0 a INT7 dessa placa estiver ativo (nível lógico "1"), uma das entradas do priority encoder (Y7 a Y0) também estará (nível lógico "0" - observe que é utilizada a saída $\bar{Q}$ do flip-flop).

Neste caso, será codificado o nº da entrada ativa mais prioritária, nº este complementado e colocado nas saídas A0 a A2 que o processador lê através de D0 a D2.

Nesse priority encoder a saída E0 será ativada (nível lógico "0"), ativando as portas correspondentes às portas ④ e ⑤ da figura 2. Essas portas colocadas nos pinos D3 e D4, respectivamente, o nº da placa programado nos jumpers JP56 e JP57. É importante observar que o nº da placa a ser programado em JP56 e JP57 deve ser complementado, para que fique coerente com o nº do chip enviado pelo 74LS348. (P. ex: p/a placa nº 0, JP56 $\rightarrow$ Vcc e JP57 $\rightarrow$ Vcc).

Nesse priority encoder, a saída EO estará inativa, com nível lógico "1".

Neste ponto, após o término do ciclo de leitura de I/O e após execução da instrução EMA, o processador terá no acumulador:

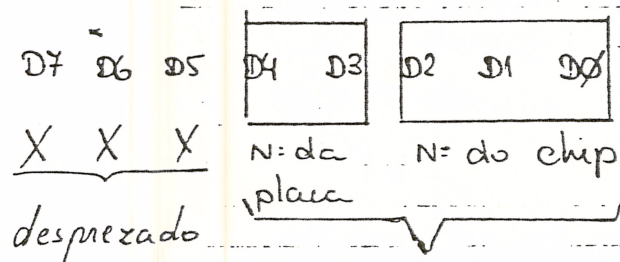


Fig. 3

N: da interface serial mais prioritária que requer serviço.

Foi então identificada a interface serial mais prioritária que requer serviço do processador. Esse é um processo relativamente rápido que requer a execução de duas instruções somente.

A partir deste ponto, é necessário saber se a interrupção foi gerada por um processo de Recepção ou de Transmissão. Isso é feito a partir da leitura da palavra de status da 8251A identificada. Uma observação importante: o TxRDY da palavra de status tem o significado de TxEMPTY. Portanto, o software precisa ter um controle adicional para saber quando o TxRDY ativo na palavra de status realmente significa interrupção por processo de transmissão. A implementação desse controle dependerá do software do sistema onde a placa estiver sendo utilizada.

Após o cabo deste procedimento, o chip que gerou o pedido de interrupção e o motivo da mesma (Tx ou Rx), já estão identificados.

Examinaremos agora o esquema de endereçamento das placas.

Para cascatear até 2 placas:

NUMR51 EQU 11111011B e
 JP40 2 → 3 ($\overline{I/O}$)
 JP41 2 → 3 (A10)

Para cascatear até 4 placas:

NUMR51 EQU 01111011B
 JP40 2 → 1 ($\overline{I/O}/M$)
 JP41 2 → 1 (A15)

O n.º da placa deve ser programado:

- em JP56 e JP57 complementado
- em JP54 e JP55 não complementado

Obs: mais detalhes sobre os jumpers na parte IV.

O esquema de cascateamento se baseia nas funções EI, EO e GS do priority encoder e nos jumpers JP52, JP51, JP53 (vide fig 2 e lou esquema elétrico da placa), além dos acima citados.

O pino EI tem a função de ativar a codificação (ativo baixo).

O pino EO estará inativo se EI assim o estiver. Se EI ativo, EO será ativado se nenhuma das entradas Y7 a Y0 estiverem ativas (neste caso GS estará inativo).

O pino GS estará inativo se EI assim o estiver. Se EI ativo, GS será ativado se ao menos uma das entradas Y7 a Y0 estiver ativa (neste caso EO estará inativo).

Isto posto, vejamos o esquema de cascataamento propriamente dito: (vide fig 2)

Placa mais prioritária → JP52 2-3 → a execução do ciclo de leitura em I/O trava os latches em todas as placas, ativa as portas correspondentes a porta ③ na fig 2 em todas as placas.

Nesse ponto o 1º priority encoder foi ativado. Se nenhuma das entradas do mesmo estiver ativa:

- GS do 1º priority encoder inativo.
- EO do 1º priority encoder ativo.

Como a porta ③ (em todas as placas) está ativa, o sinal EO, através do JP53 vai para uma das linhas do bus (entre D5, D6 ou D7). Vamos supor que em JP53 1-2; daí EO vai p/D5. Este sinal será usado pela placa seguinte (a 2ª placa em prioridade) para gerar o EI para o seu priority encoder. Vejamos como:

O sinal EO da 1ª placa está presente em D5.

Na 2ª placa, JP51 é configurado para 1-2, passando então o sinal EO p/ a porta ② (vide fig 2). Na 2ª placa JP52 é configurado para 1-2. Como se pode ver, o sinal EO gerado pelo priority encoder da 1ª placa gerou o sinal EI do priority encoder da 2ª placa. E assim por diante, utilizando também as linhas D6 e D7, pode-se cascatear até 4 placas. O primeiro dos 4 priority encoders que

tiver ao menos uma entrada (10 a 17) ativa, não gerar EO inativo p/ fonte, e, conseqüentemente, manter os priority encoders menos prioritários inativos. Da mesma forma, o primeiro priority encoder que tem ao menos uma entrada ativa, não ativa seu pino Gs, ativando no bus, através das portas correspondentes as portas (4) e (5) (fig 2) de sua placa, o nº dessa placa nos pinos D3 e D4.

Existe também na placa um circuito composto de um latch (CI19 - 74LS174), um comparador (CI20 - 74LS85) e um decodificador (CI21 - 74LS138), que tem por função a geração do $\overline{CS}$ para as 8251A. Além disso, auxilia a implementação das rotinas de tratamento das 8251A.

O processador insere no latch o nº do chip que vai tratar. O comparador analisa os bits correspondentes ao nº da placa para geração do sinal "placa" (pino 6 do 7485). Esse sinal e A11 ativam o decoder (74LS138)

NUMW51 EQU 11111011B; end latch

ADD51 EQU 11110110B; end de dados da 8251A tratada

ADC51 EQU 11110111B; end de comando da 8251A tratada

PCI MONT RACK-INTERFACE SERIAL 8251x8

Exemplos de utilização desse sistema:

- inicializando uma 8251A;

```

MVI A, 01 ; inicializando chip n:1
OUT NUMWS1
MVI A, MODO
OUT ADCS1
MVI A, COMANDO
OUT ADCS1

```

- lendo status de uma 8251A numa rotina de interrupção

```
IN NUMRS1
```

```
CMA
```

```
ANI 1FH
```

```
OUT NUMWS1
```

```
MVI A, COMANDO
```

```
OUT ADCS1.
```

; identificar o chip.

; salvar em mem o n: da 8251A.

; n: da 8251A está no acumulador

Num sistema onde a rotina de interrupção atua sobre o registrador NUMWS1, duas condições devem ser satisfeitas:

- durante o tratamento da interrupção, nenhum nível de interrupção que atue sobre NUMWS1 pode estar permitido (em geral basta que o próprio nível fique desabilitado).

- a nível de programa principal, quando se efetuar operações sobre o registrador NUMWS1 e sobre as 8251A (escrever comandos ou dados, ler status ou dados) deve-se fazê-lo com o nível de interrupção que trata as 8251A desabilitado (isto se a rotina de interrupção atuar sobre NUMWS1).