RD40 cards

Installation and user manual.

@ 1998 *ELPA* **sas**

Corso Kennedy 32/21 - 10098 Rivoli (TO) - ITALY
Tel. 011.9574088 – Fax: 011.9594900 – e-mail: elpa@elpa.it

1 - Configuration

The board you have is already configured to manage 2 or 3 encoders, working at 5V or 12V. The user cannot change this configuration.

The only thing the user can change is the I/O mapping address; this can be done by inserting/removing some jumpers from the socket.

The preset address is 210 hexadecimal, that normally isn't used from PC; if it's already used by other cards, or if you need to install more than one RD40 card in the same PC, you need to change the address of one or more cards to avoid address conflicts, that can lock the PC.

This address can be changed by inserting/removing some jumpers on the horizontal strip socket, on the left of the card. There are 8 jumpers, corresponding to 8 address lines; the leftmost is A9, the following is A8, up to the last one that is A2.

Bits A1 and A0 are always fixed at 0.

If you insert a jumper, you request that bit to be 1 to address the card.

The presetted configuration has jumpers A9 and A4 inserted, corresponding to address 210 (hexadecimal).

The jumpers must be inserted only vertically.

Example: if you need to setup the address 300 hexadecimal, you need to insert only the 2 leftmost jumpers, A9 and A8.

2 - Installation

RD40 board must be inserted in a PC compatibile computer, in an 8 bit ISA slot. It's also possible to use a 16 bit ISA slot or a VESA slot; in these cases, the card will be inserted only in the 8 bit ISA portion of the slot.

To insert the board, please read the computer installation manual, and remember that the board can be inserted or removed only with the power supply switched off.; failure to do so disables warranty.

Your card has 2 or 3 female DB9 connectors, that are functionally equal; each of them can be connected to one incremental encoder (or similar device).

Depending by the device you are using, you must wire:

- Unipolar 5V device ('A' type cards):
  - 1 - 0V (common)
  - 2 - +5V
  - 3 - Phase 'C'
  - 4 - Phase 'B'
  - 5 - Phase 'A'

- Bipolar 5V device ('A' type cards):
  - 1 - 0V (common)
  - 2 - +5V
  - 3 - Phase 'C' positive
  - 4 - Phase 'B' positive
  - 5 - Phase 'A' positive
  - 7 - Phase 'C' negative
  - 8 - Phase 'B' negative
  - 9 - Phase 'A' negative

- Unipolar 12V device ('B' type cards):
  - 1 - 0V (common)
  - 2 - +12V
  - 3 - Phase 'C' (connect to pin 2 if not present)
  - 4 - Phase 'B'
  - 5 - Phase 'A'

Not indicated pins must not be connected.

If you are using unipolar 5V encoders (with an 'A' card), you must remove from their sockets the resistor arrays R8, R9 and R10.
It's not good practice to connect powered devices, but it's possible to (dis)connect the encoders to the RD40 card when it's powered up.
Because there are ESD protection connected to the inputs, it's forbidden to drive the input signals when the computer is powered off.

3 - MS-DOS™ interface

To interface with MS-DOS, in the provided disk there are several directories, one for each supported compiler. In these directories there are also some static libraries, named "RD40LIB*.LIB", where the letter that substitutes the '*' sign is the leading letter of the memory model used to compile the library (for additional informations, please refer to the ".TXT" file in the same directory).

To use these libraries you need to include in the C source code the file "RD40.H" that contains the function's prototypes, and to add to your project the library needed.

In the disk there is also the program "RD40DTST.EXE". This program tests the RD40 card; it displays the counter values, and verifies the home search procedure. This procedure can be (de)activated for tha various axes by pressing the keys 1, 2 or 3.

To quit the program you have to press the X key.

The program searches for a RD40 card at the address 210 hexadecimal; if the board is mapped at a different address, you must run the program with the address parameter, in decimal format. Example: "RD40DTST 768" to test a card mapped at address 300 hexadecimal.

4 - Windows™ and Windows 95™ interface

To interface with Windows and Windows 95, you can use static or dynamic libraries.

If you use static libraries, please read the chapter "MS-DOS interface", because the operations requested are exactly the same as the previous case; otherwise, in the same directories there are also dynamic libraries, named "RD40DLL*.DLL", with their link modules named "RD40DLL*.LIB" and the include file "RD40.H". Also in this case, you must substitute the "*" sign with the leading letter of the memory model you wish to use.

To use dynamic library, you need to include in the C (or C++) source code the file RD40.H, and to add to your project the file RD40DLL*.LIB corresponding to the memory model used. Before running the compiled program, you have also to copy in the Windows directory (or in the same directory where there is the program) the file RD40DLL*.DLL.

Attention: not all libraries works on all operating systems. Please refer to the ".TXT" file in the directory.

In the disk there is also a program, named "RD40WTST.EXE", to test the card.

It must be started from Windows (or Windows 95), and it's similar to the MS-DOS version. Also the optional parameter is identical, and the default value is 210 hexadecimal. The only difference is that the user interface is the Windows' one.

5 - Library functions' description

All libraries contain the same functions:

- char _far rd40_config(char encoder,unsigned int address,char channel)

It configures an encoder input; this function must be called only at startup, once for every managed encoder. The first parameter is the encoder to setup (internal numeration, from 0 to 31); the second parameter is the card's address (please refer to the "Installation" chapter), the third one is the connector on the card (0=bottom, 1=center, 2=top).

If you specify an address in which there isn't a RD40 card or a not existing connector, the function returns RD40_NOT_FOUND.

- char _far rd40_getcnt(char encoder,long *count);

Reads into the variable "count" the counter value of the selected encoder; the "encoder" parameter follows the internal numbering. In the read value, only the 24 LSB are valid; the 8 MSB are only a sign extension.

- char _far rd40_ishomed(char encoder);

It returns RD40_HOMING if the selected encoder is searching the home position.

- char _far rd40_home(char encoder,char enable);

Enables or disables (depending by parameter "enable") the home search procedure of the selected encoder.

This procedure is automatically ended when the phase "C" signal becomes active.

These functions can return the following values:

RD40_OK if they complete correctly.

RD40_BAD_PARAMETER if one or more parameters are bad.

RD40_NOT_DEFINED if you are trying to use an encoder that wasn't configured with the function rd40_config.

6 - Internal registers

The card can also be used by reading/writing directly to its internal registers, mapped in the I/O space. It uses 4 consecutive addresses, starting from the one selected with the jumpers.

To read the 24 bit counter value, you need to access 3 8 bit register; the register mapping is the following:

Offset	Access	Register description
00	Write	Read configuration
00	Read	Counter bits 0..7 or identification code
01	Write	Reserved
01	Read	Counter bits 8..15
02	Write	Home search configuration
02	Read	Counter bits 16..23
03	Write	Reserved
03	Read	Home search status

Offset 00 (Write): Read configuration register

Bit	Description
0	0 = Nothing
	1 = Latches the first encoder position in an internal register
1	0 = Nothing
	1 = Latches the second encoder position in an internal register (if present)
2	0 = Nothing
	1 = Latches the third encoder position in an internal register (if present)
3-5	Reserved
6-7	00 Selects first encoder reading
	10 Selects second encoder reading (if present)
	01 Selects third encoder reading (if present)
	11 Selects identification register reading

8

Offset 00 (Read): Encoder latched counter value or identification register

Bit	Description
0-7	Encoder latched counter value bits 0..7
0-7	0x59 Card RD40-2
	0x5A Card RD40-3

Offset 01 (Write): Reserved

Offset 01 (Read): Encoder latched counter value

Bit	Description
0-7	Encoder latched counter value bits 8..15

Offset 02 (Write): Home axes configuration register

Bit	Description
0	0 = First encoder home search disabled
	1 = First encoder home search enabled
1	0 = Second encoder home search disabled
	1 = Second encoder home search enabled
2	0 = Third encoder home search disabled
	1 = Third encoder home search enabled
3-7	Reserved

Offset 02 (Read): Encoder latched counter value

Bit	Description
0-7	Encoder latched counter value bits 16..23

Offset 03 (Write): Reserved

Offset 03 (Read): Home axes status of the selected encoder

Bit	Description
0-6	Reserved
7	0 = Home search disabled

7 - Technical specifications

Maximum input frequency	2 MHz
Maximum count frequency	8 MHz
Latency time	250 ns
Counter lenght	24 bit
Load resistance (5V cards)	1.1 Kohm
Dynamic termination resistance	120 Ω
Load resistance (12V cards)	3.3 Kohm
Differential threshold (5V cards)	100 mV
Unipolar threshold level (5V cards)	1.66 V
Input low current (5V cards)	1.62 mA
Threshold level (12V cards)	6 V
5V Supply current[*] (RD40-3A card)	210 mA
12V Supply current[*]	0 mA
Maximum encoder supply current	750 mA
Operating temperature	from 0 to 70°C

---

[*] Doesn't include encoders' supply current

9 - Visual Designer™ interface

In the disk there is also the directory "VIS_DES", that contains several directories, one per every implemented language. To install the driver you have to copy from the right directory the file "RD40_VD.DLL" in the "\IO" subdirectory of the directory in which there is Visual Designer, and the file "RD40_VD.HLP" in the Visual Designer directory.

After doing this, from Diagram you can select a new type IO block, called "Encoder RD40". Further explanations about configuration, inputs, outputs and working are in the help file, that you can access in the classic Visual Designer™ style.