fast real-time processing, measuring and controlling for windows

ADwin-LD

Hardware Manual

Register Assignment

Sample Programs

Version 1.2

June 98

Programming the *ADwin-LD* Board

The ***ADwin-LD*** board has 44 digital I/O ports, which can be accessed by three 16-bit words.

The digital inputs can be read out by using the addresses **04h** (DIO0 - DIO15), **14h** (DIO16 - DIO31) and **24h** (DIO32 - DIO44).

When power is switched on, all 44 I/O ports are configured as inputs. By using the **IO_SET** register at address **0B4h** the ports can be configured as outputs arranged in four blocks. If the **IO_SET** register is set to 1, ports 0 to 3 are configured as outputs. If the **IO_SET** register is set to 4 (= 0100B in binary code) the ports 8 to 11 are configured as outputs.

The digital outputs can be set by using the addresses **74h** (DIO0 - DIO15), **84h** (DIO16 - DIO31) and **94h** (DIO32-DIO44).

The board has 4 16-bit counters, which are controlled by the **CO_SET** register at address **0A4h.** If the bits 0,2,4 or 6 of the **CO_SET** register are set to "1", then the corresponding counters 1, 2, 3 or 4 will be cleared. If these bits are set to "0" the counter shows no reaction. Setting the bits 1,3,5 or 7 of the **CO_SET** register to "1" starts the counters which count all the rising edges at the corresponding counter input. When setting these bits to "0" the counters stop.

In order to read out the counter values you have to latch the counters by setting the **CO_LATCH** register (address **0C4h**) to 1 first. The counter values can then be read out by using the addresses **34h** (counter1), **44h** (counter2), **54h** (counter3) and **64h** (counter4).

Each of the counters 1,2 and 3,4 can be cascaded to a 32-bit counter. It is necessary to set the bit 13 (for counters 1,2) or bit 15 (for counters 3,4 respectively) in the **IO_SET** register.

Notes for the *ADwin-2LD* boards :

The ***ADwin-2LD*** board addresses are arranged differently:

```
DIN1  = 7F02h
DIN2  = 7F0Ah
DIN3  = 7F12h
DOUT1 = 7F3Ah
DOUT2 = 7F42h
DOUT3 = 7F4Ah
CO1 = 7F1Ah
CO2 = 7F22h
CO3 = 7F2Ah
CO4 = 7F32h
CO_SET = 7F52h
IO_SET = 7F5Ah
CO_LATCH = 7F62h
```

Notes for the *ADwin-9LD* boards :

The ***ADwin-9LD*** board addresses are arranged differently:

```
DIN1   = 20400004h
DIN2   = 20400014h
DIN3   = 20400024h
DOUT1  = 20400074h
DOUT2  = 20400084h
DOUT3  = 20400094h
CO1    = 20400034h
CO2    = 20400044h
CO3    = 20400054h
CO4    = 20400064h
CO_SET = 204000A4h
IO_SET = 204000B4h
CO_LATCH = 204000C4h
```

ADbasic sample programs

1. Reading digital inputs

```
dim DIN1, DIN2, DIN3, DOUT1, DOUT2, DOUT3 as integer
dim CO1, CO2, CO3, CO4, CO_SET, CO_LATCH, IO_SET as integer

INIT:
DIN1  = 04h
DIN2  = 14h
DIN3  = 24h
DOUT1 = 74h
DOUT2 = 84h
DOUT3 = 94h
CO1 = 34h
CO2 = 44h
CO3 = 54h
CO4 = 64h
CO_SET = 0A4h
IO_SET = 0B4h
CO_LATCH = 0C4h
POKE IO_SET,0        ' configure all ports as inputs

EVENT:
PAR_1 = PEEK(DIN1)
PAR_2 = PEEK(DIN2)
PAR_3 = PEEK(DIN3)
```

2. Setting digital outputs

```
DIM DIN1, DIN2, DIN3, DOUT1, DOUT2, DOUT3 AS INTEGER
DIM CO1, CO2, CO3, CO4, CO_SET, CO_LATCH, IO_SET AS INTEGER
DIM i1 AS INTEGER

INIT:
DIN1  = 04h
DIN2  = 14h
DIN3  = 24h
DOUT1 = 74h
DOUT2 = 84h
DOUT3 = 94h
CO1 = 34h
CO2 = 44h
CO3 = 54h
CO4 = 64h
CO_SET = 0A4h
IO_SET = 0B4h
CO_LATCH = 0C4h
POKE IO_SET,01111B  ' configure ports 0 to 15 as outputs
i1 = 0

EVENT:
POKE DOUT1, i1
i1 = i1 +1
```

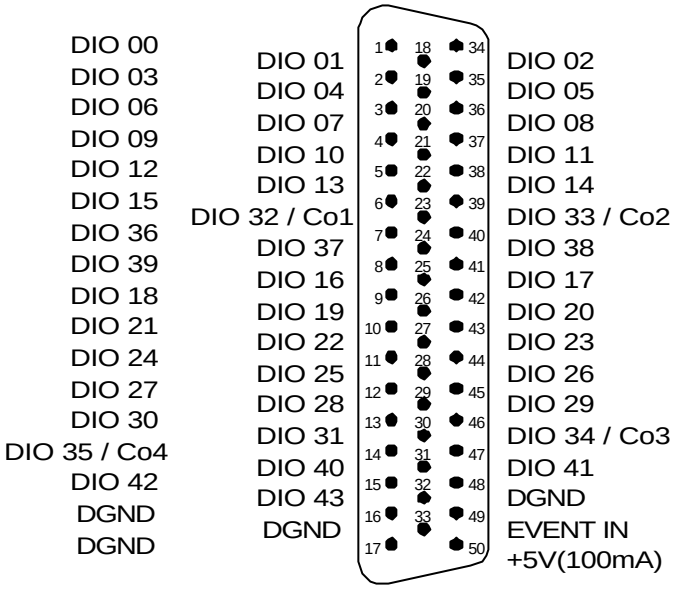
3. Reading out counters

```
DIM DIN1, DIN2, DIN3, DOUT1, DOUT2, DOUT3 AS INTEGER
DIM CO1, CO2, CO3, CO4, CO_SET, CO_LATCH, IO_SET AS INTEGER
DIM co1a AS INTEGER

INIT:
DIN1  = 04h
DIN2  = 14h
DIN3  = 24h
DOUT1 = 74h
DOUT2 = 84h
DOUT3 = 94h
CO1 = 34h
CO2 = 44h
CO3 = 54h
CO4 = 64h
CO_SET = 0A4h
IO_SET = 0B4h
CO_LATCH = 0C4h           ' clear all counters
POKE CO_SET, 010101010B   ' start all counters

EVENT:
POKE CO_LATCH, 1          ' latch the counters
PAR_1 = PEEK(CO1)         ' read counter1
PAR_2 = PEEK(CO2)
PAR_3 = PEEK(CO3)
PAR_4 = PEEK(CO4)
PAR_11 = PAR_1 - co1a     ' change of counter1
co1a = PAR_1
```

Pin assignment of the *ADwin-LD* board



ADwin-LD

The I/O socket at the rear of the board

Setting the Link I/O Address

The setting of the I/O address is made by using the DIP-switch on the board.
The DIP-switch sets the base address for the linkadapter on the ***ADwin*** board.

The base address is set in binary code, beginning with the third bit. For instance, the base address **150h** used by the ISERVER results in the binary code **1 0101 0000**. Omit the two least significant bits and set the remaining "1-figures" in reverse order to "ON", that is:

address bit:	2	3	4	5	6	7	8	9
switch no.:	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
base address:								
150h	**OFF**	**OFF**	**ON**	**OFF**	**ON**	**OFF**	**ON**	**OFF**
190h	**OFF**	**OFF**	**ON**	**OFF**	**OFF**	**ON**	**ON**	**OFF**
200h	**OFF**	**OFF**	**OFF**	**OFF**	**OFF**	**OFF**	**OFF**	**ON**
300h	**OFF**	**OFF**	**OFF**	**OFF**	**OFF**	**OFF**	**ON**	**ON**

Important:

In case of using a different base address than 150h the address has to be declared when booting the transputer with the “iserver“ -program.

If the digital I/O module is used together with an analog module the link address has to be set to a value which is not used (i.e. 0, that means, all switches are set to the position "OFF").
The same link address which is set on the active link adapter of the analog module **must not** be used.
Seen from the component side, the digital module must always be connected to the link adapter behind the analog module.