Port Definition

You are required to add memory-mapped IOs to your processor. You access the appropriate Register by issuing a STORE instruction to special addresses. You then have to implement the logic which intercepts that STORE to the special address and enables a WE on the Register. Proceed similarly with the load.

Ports

The ports can be used to read the 5 Pushbuttons by reading (LOAD) the register PIND (in ATMEGA8 at address 0x30) and to set the status of the 8 leftmost LEDs by writing (STORE) to the register PORTC (in ATMEGA8 at address 0x35).

PIND – Port IN D Register
Address: 0x30

-	-	-	RIGHT	UP	DOWN	LEFT	ENTER
			R	R	R	R	R

PINC – Port IN C Register
Address: 0x33

SW15	SW14	SW13	SW12	SW11	SW10	SW9	SW8
R	R	R	R	R	R	R	R

PINB – Port IN B Register
Address: 0x36

SW7	SW6	SW5	SW4	SW3	SW2	SW1	SW0
R	R	R	R	R	R	R	R

PORTC – Port C output Register
Address: 0x35

LED15	LED14	LED13	LED12	LED11	LED10	LED9	LED8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

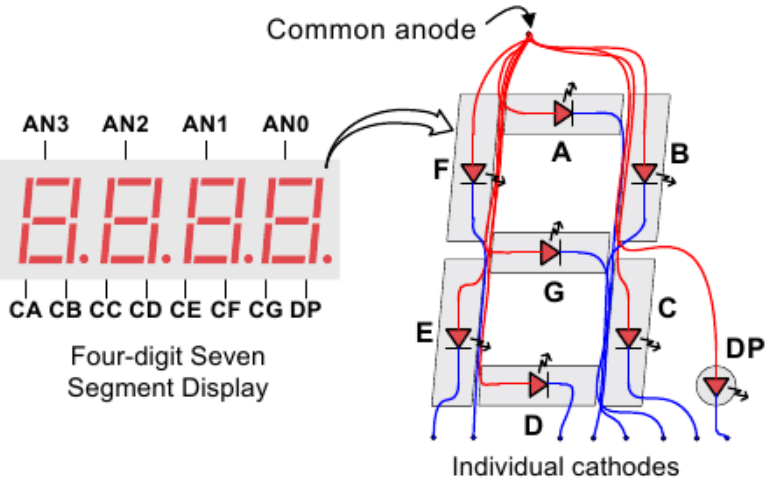
PORTB – Port B output Register
Address: 0x38

LED7	LED6	LED5	LED4	LED3	LED2	LED1	LED0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

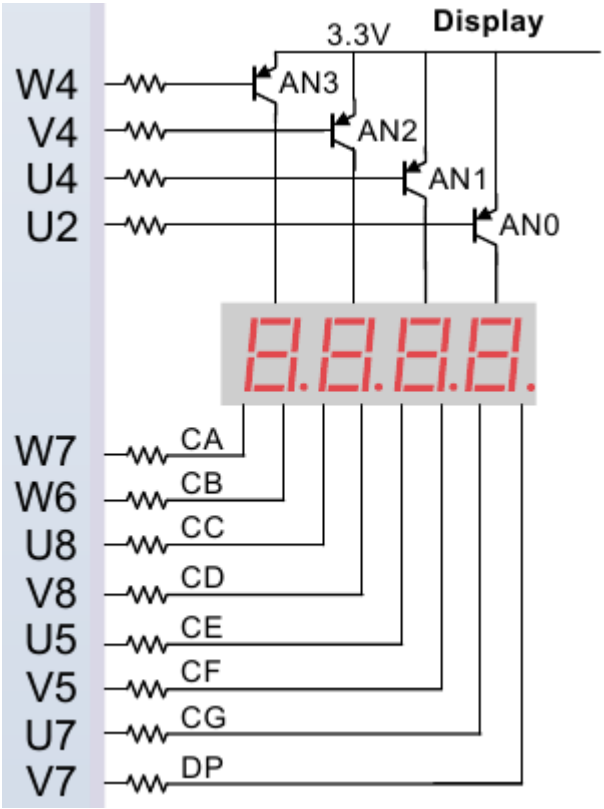
Four-digit Seven Segment IO

The following explanation including the figures is taken from the „Basys3TM FPGA Board Reference Manual“ [1]:

The anodes of the seven LEDs forming each digit are tied together into one “common anode” circuit node, but the LED cathodes remain separate, as shown below.



The common anode signals are available as four “digit enable” input signals to the 4-digit display.



1 See https://www.digilentinc.com/Data/Products/BASYS3/Basys3_rm.pdf

The peripheral module to drive the four 7 Segment diodes needs the following 5 registers:

SER – Segment Enable Register
Address: 0x40

-	-	-	-	AN3	AN2	AN1	AN0
				R/W	R/W	R/W	R/W

SEG0_N – Segment 0 contents
Address: 0x41

S0DP	S0G	S0F	S0E	S0D	S0C	S0B	S0A
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SEG1_N – Segment 1 contents
Address: 0x42

S1DP	S1G	S1F	S1E	S1D	S1C	S1B	S1A
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SEG2_N – Segment 2 contents
Address: 0x43

S2DP	S2G	S2F	S2E	S2D	S2C	S2B	S2A
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SEG3_N – Segment 3 contents
Address: 0x44

S3DP	S3G	S3F	S3E	S3D	S3C	S3B	S3A
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

I²C Interface

The I²C Interface consists of 3 Registers:

I2C_SCR – I²C Status and Control Register
Address: 0x50

-	ack	start	stop	start_clk	DaTR_empty	RXC	read/write
	R	R/W	R/W	R/W	R	R/W	R/W
	‘0’	‘0’	‘0’	‘0’	‘1’	‘0’	‘0’

I2C_SCR(6) - ack: the acknowlegde bit of last transmission - is written by hardware into this bit.

I2C_SCR(5) - start: writing '1' to this bit triggers the start sequence. Bit is cleared by hardware once the start sequence is finished. Writing '0' to this bit has no effect.

I2C_SCR(4) - stop: writing '1' to this bit triggers the stop sequence. Bit is cleared by hardware once the stop sequence is finished. Writing '0' to this bit has no effect.

I2C_SCR(3) - start_clk: writing '1' to this bit triggers the generation of 9 SCC pulses. If SCR(0) is '1' the data of DaTR are shifted out. In any case data at SDA are shifted into the DaRR. If SCL is not already low because of the start sequence, the behaviour is undefined. The Bit is cleared by hardware once the 9 pulses were generated. Writing '0' to this bit has no effect.

I2C_SCR(2) - DaTR_empty: This bit is set by hardware if there are no data to be transmitted in the DaTR. This bit is cleared by hardware if the I2C is busy transmitting data. DaTR should only be written if DaTR_empty is set.

I2C_SCR(1) - RXC: This bit is set by hardware if there are new data in DaRR. The bit has to be cleared via software by writing a '1' to its bit location.

I2C_SCR(0) - read/write: writing a '1' to this bit will cause to shift out the data in the DaTR in the next transmission. Writing a '0' to this bit will just shift the data from the SDA into the DaRR with the next transmission. This bit should only get altered when the start_clk bit is '0'.

I2C_DaTR – I²C Data Transmit Register
Address: 0x51

DaTR7	DaTR6	DaTR5	DaTR4	DaTR3	DaTR2	DaTR1	DaTR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

I2C_DaRR – I²C Data Receive Register
Address: 0x52

DaRR7	DaRR6	DaRR5	DaRR4	DaRR3	DaRR2	DaRR1	DaRR0
R	R	R	R	R	R	R	R