Programming Microcontroller IntroES

Dipl.- Ing. Falk Salewski

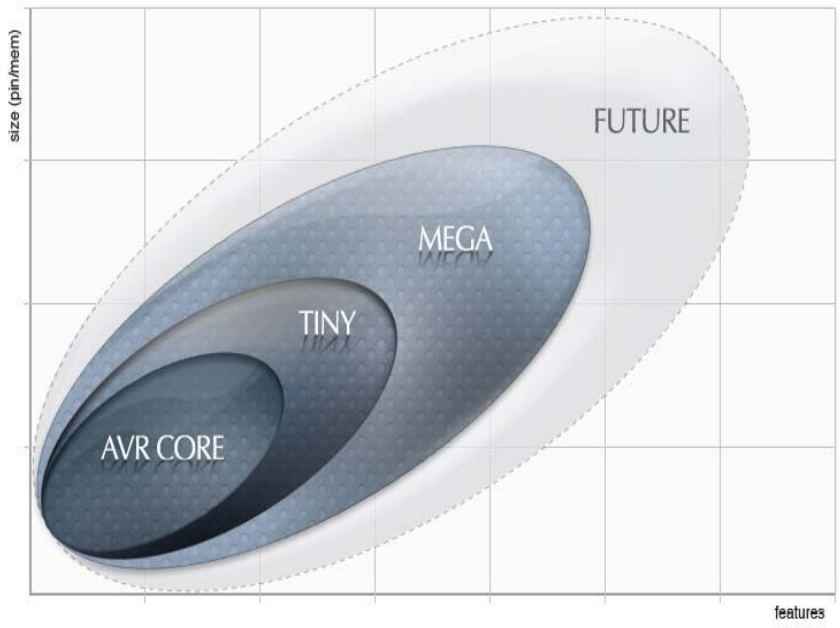
Lehrstuhl Informatik XI
RWTH Aachen
salewski@informatik.rwth-aachen.de

WS 05/06

Microcontroller basics

- Microcontroller = CPU + Memory + internal Peripherals

AVR Microcontroller Family



- Devices range from 1 to 256KB
- Pin count range from 8 to 100
- Full code compatibility
- Pin/feature compatible families
- One set of development tools
- In-System Programming
- In-System Debugging

More on http://www.atmel.com/products/avr/overview.asp

ATmega16

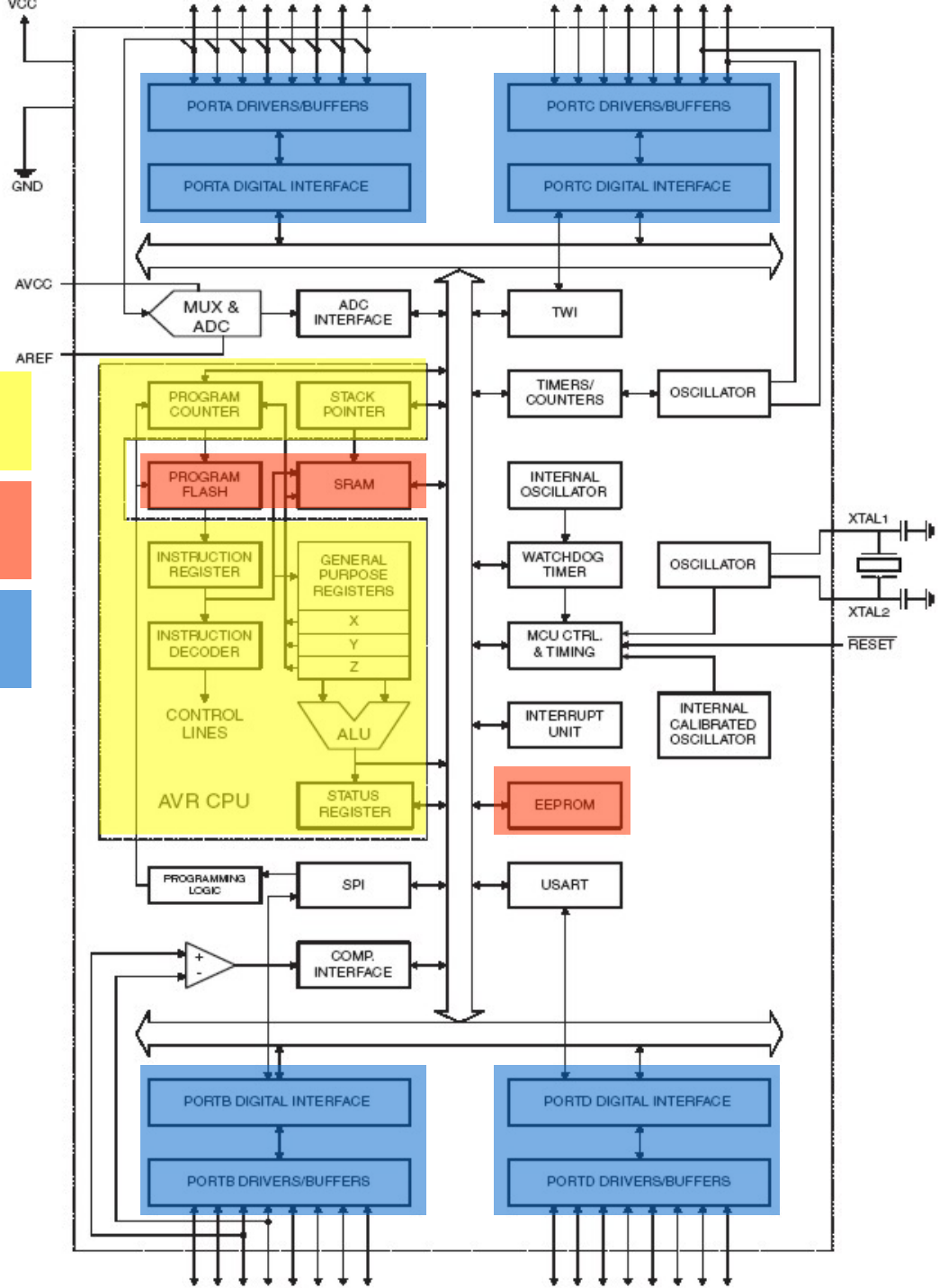
- We will use the ATMEL ATmega16
  - 8-bit Microcontroller with 16K Bytes In-System Programmable Flash
  - 512 Bytes EEPROM, 1K Byte Internal SRAM
  - Two 8-bit Timer/Counters, One 16-bit Timer/Counter
  - 8-channel, 10-bit ADC
  - Programmable Serial USART
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - JTAG Port for in-system programming and debugging
  - …
- More details: http://www.atmel.com/products/avr/

ATmega16

8bit CPU

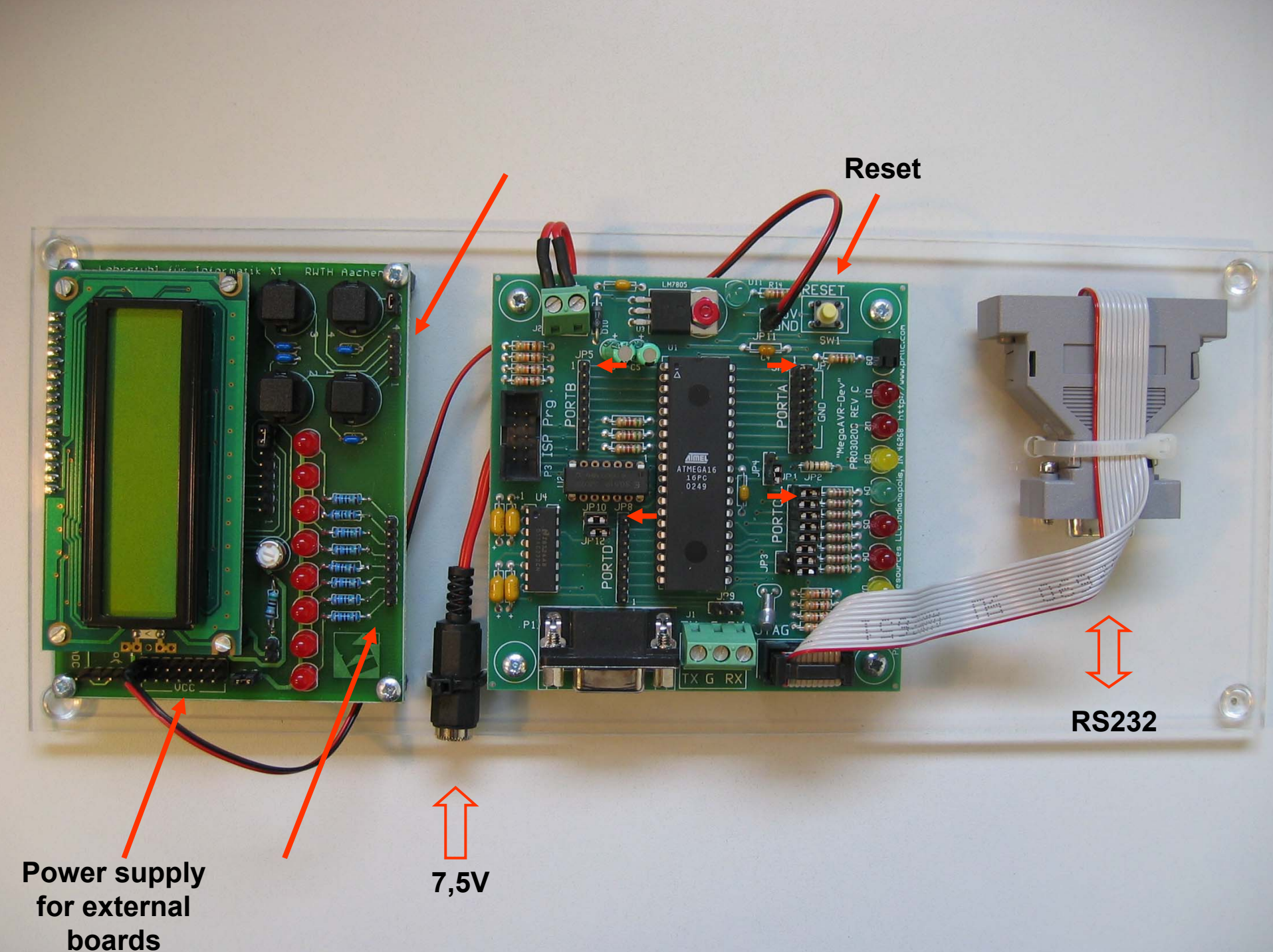
Memory

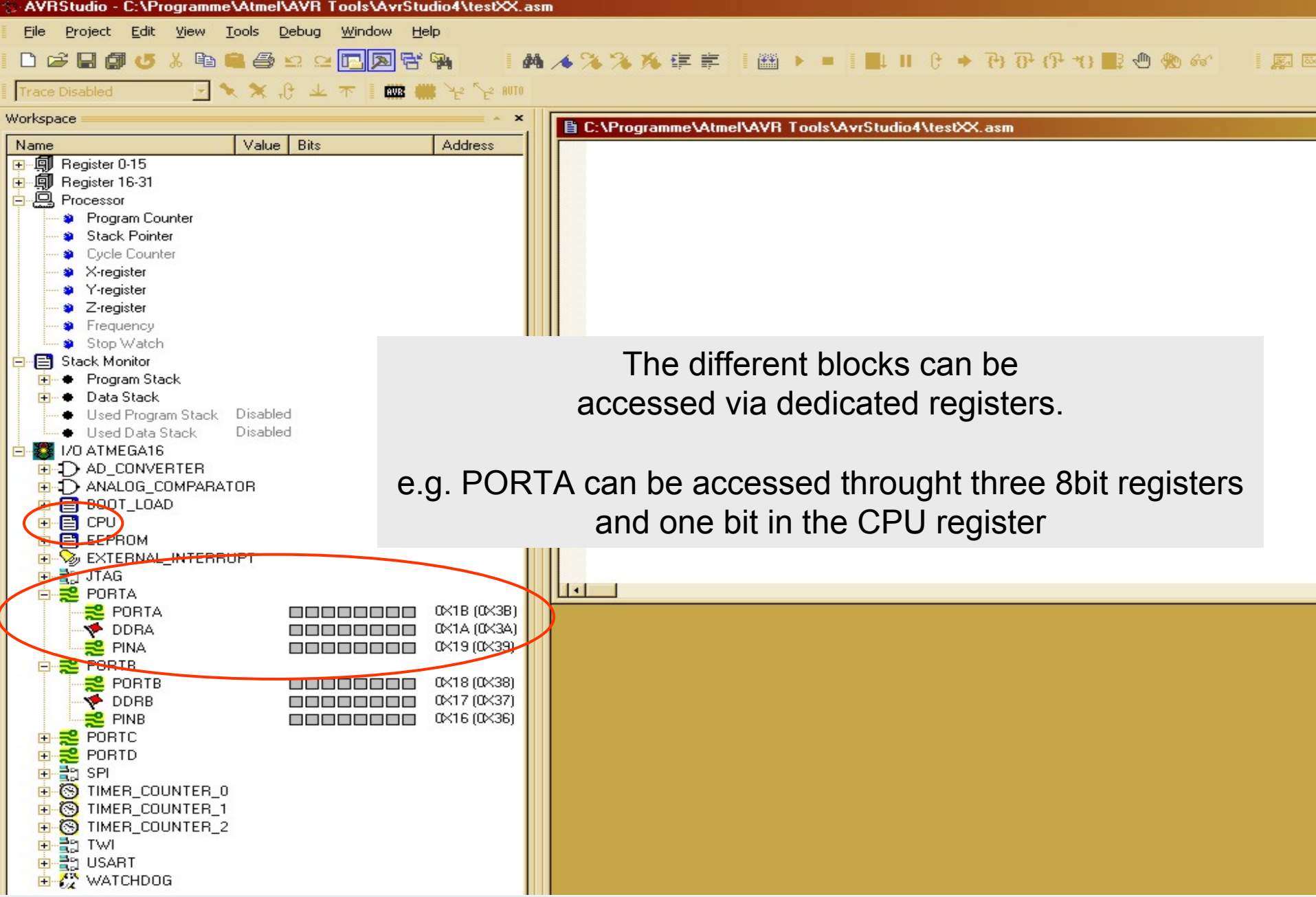
I/O Ports



RWTH

LEHRSTUHL INFORMATIK XI SOFTWARE FÜR EINGEBETTETE SYSTEME


Reset
RS232
7,5V
Power supply
for external
boards



The different blocks can be accessed via dedicated registers.

e.g. PORTA can be accessed throught three 8bit registers and one bit in the CPU register

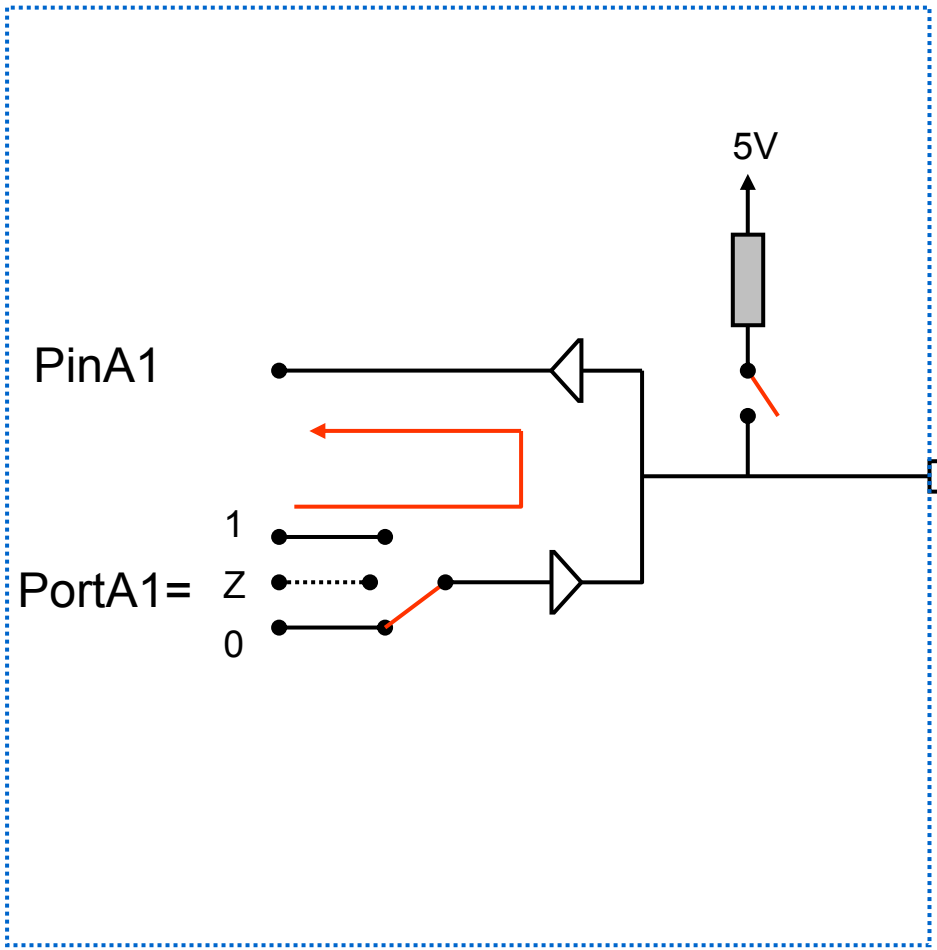
ATmega16 – I/O Ports

Port Pin Configurations PORTA:

DDRA	PORTA	PUD	I/O	Pull-up	Comment
0	0	X	Input	No	Tri-state
0	1	0	Input	Yes	
0	1	1	Input	No	Tri-state
1	0	X	Output	No	Output Low
1	1	X	Output	No	Output High

The port pin can always be read through the PINA Register bit.

Example: turn on an LED

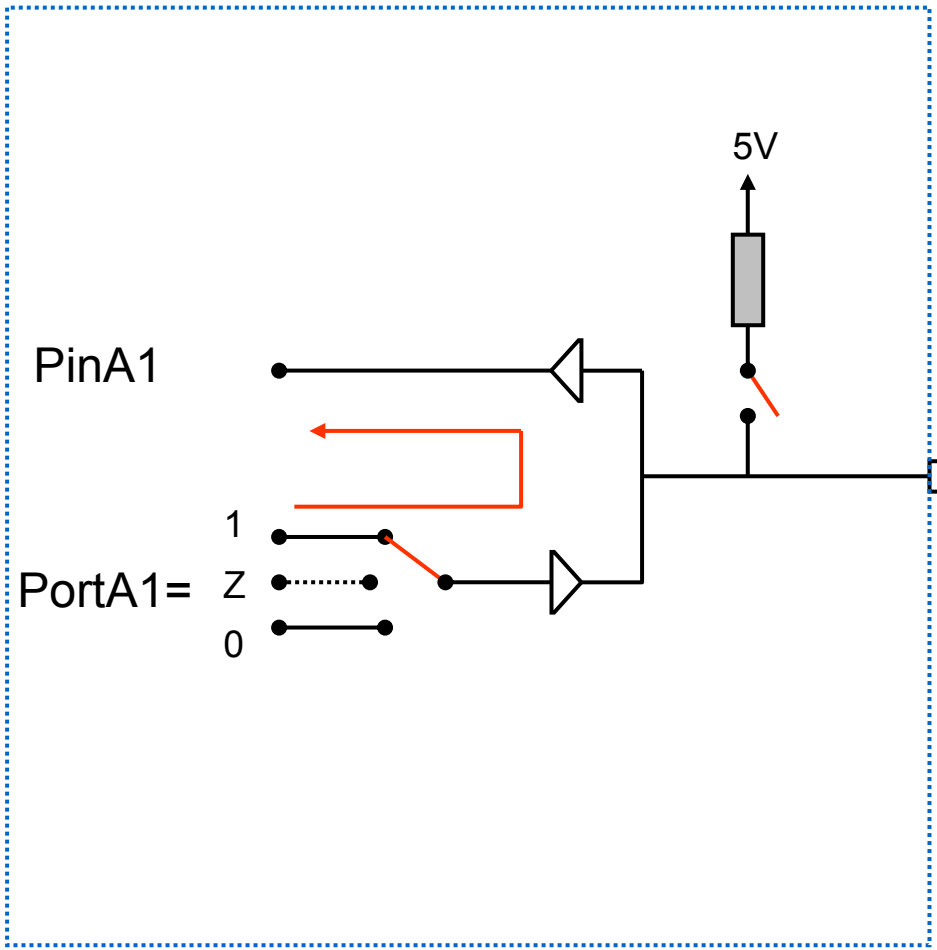


DDRA	PORTA	PUD	Signal on PINA1
0	0	X	S1 closed: 0 S1 open: U
0	1	0	S1 closed: 0 S1 open: 1
0	1	1	S1 closed: 0 S1 open: U
1	0	X	PortA1
1	1	X	PortA1

X: don‘t care
U: unknown

LED is on !

Example: turn the LED off

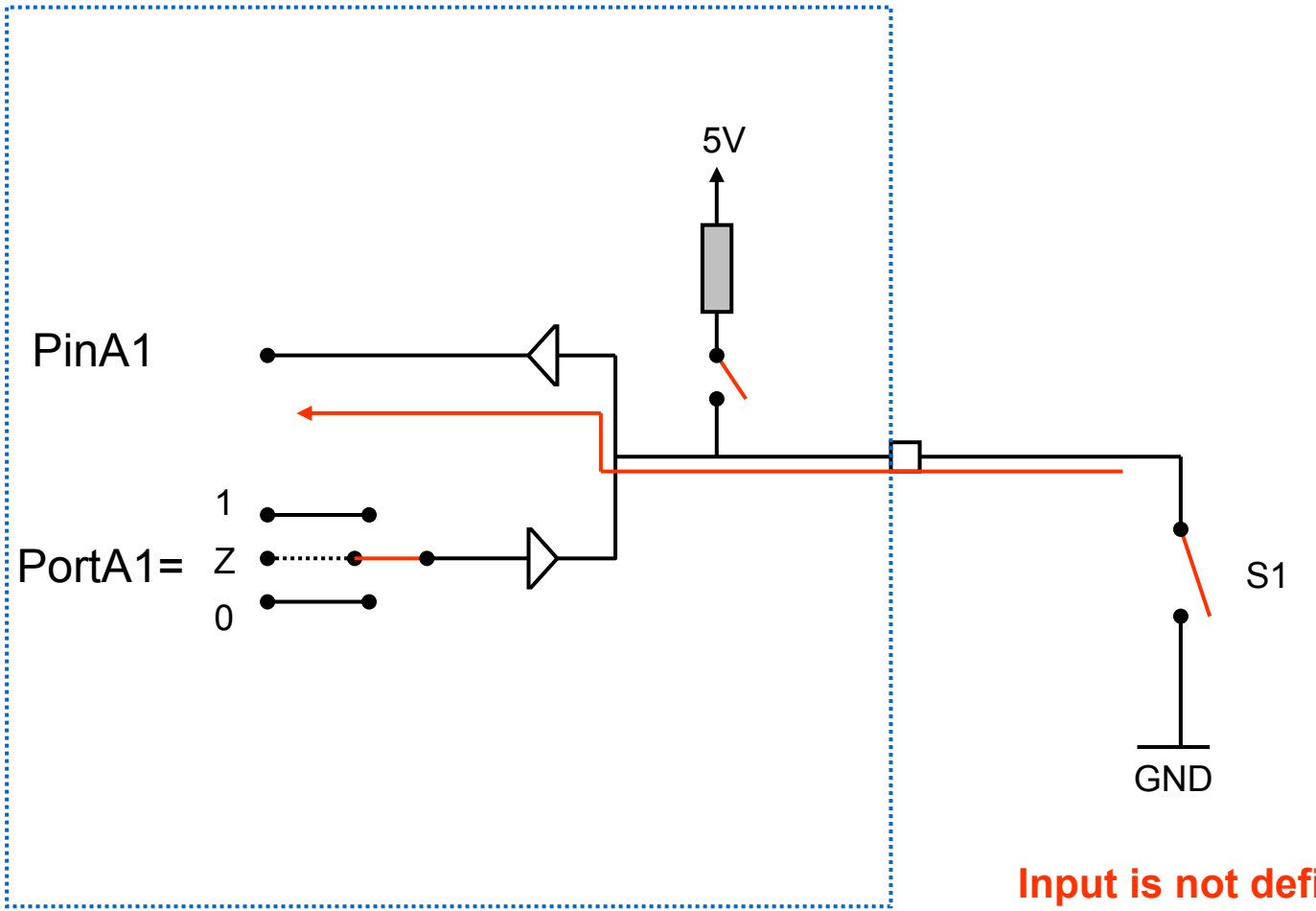


DDRA	PORTA	PUD	Signal on PINA1
0	0	X	S1 closed: 0 S1 open: U
0	1	0	S1 closed: 0 S1 open: 1
0	1	1	S1 closed: 0 S1 open: U
1	0	X	PortA1
1	1	X	PortA1

X: don‘t care
U: unknown

LED is off !

Example: Reading Input

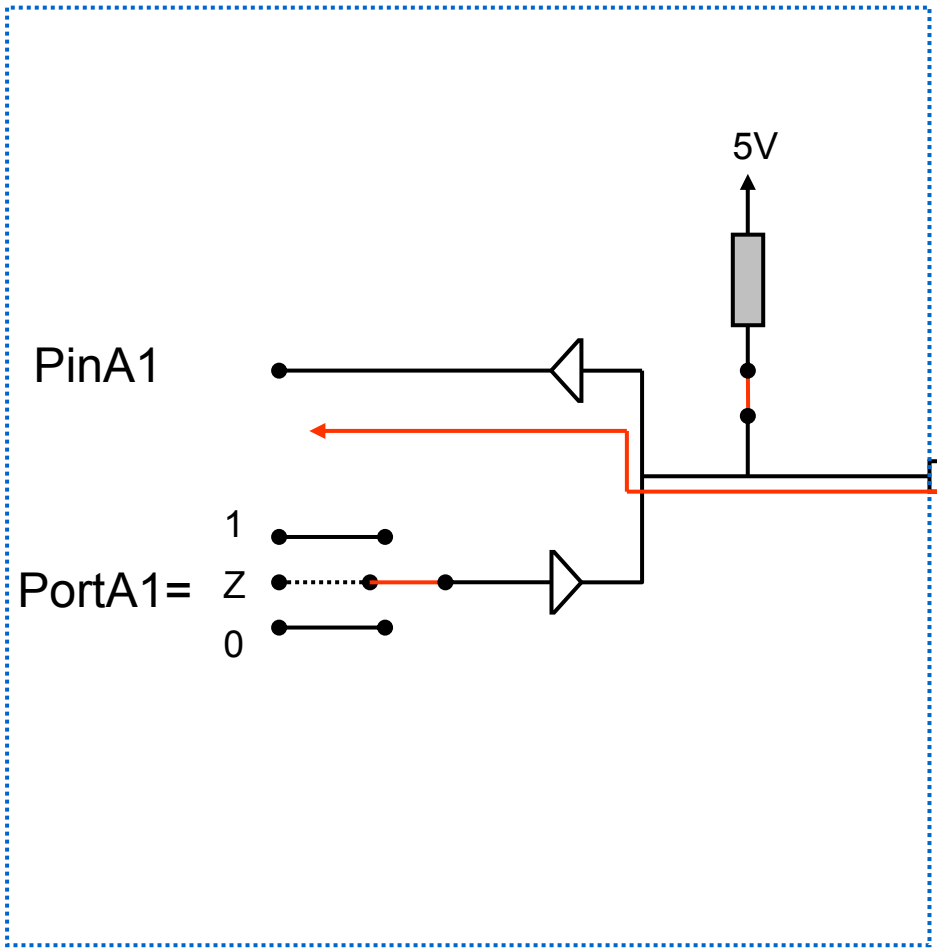


DDRA	PORTA	PUD	Signal on PINA1
0	0	X	S1 closed: 0 S1 open: U
0	1	0	S1 closed: 0 S1 open: 1
0	1	1	S1 closed: 0 S1 open: U
1	0	X	PortA1
1	1	X	PortA1

X: don‘t care
U: unknown

Input is not defined if S1 is open!

Example: Reading Input



DDRA	PORTA	PUD	Signal on PINA1
0	0	X	S1 closed: 0 S1 open: U
0	1	0	S1 closed: 0 S1 open: 1
0	1	1	S1 closed: 0 S1 open: U
1	0	X	PortA1
1	1	X	PortA1

X: don‘t care
U: unknown

Input is now also defined if S1 is open!

Register in C

	Lesen	Schreiben
Bitweise	`bit_is_set (<port>, <pin>);` `bit_is_clear (<port>, <pin>);`	`sbi (<register>, <bitnummer>);` `cbi (<register>, <bitnummer>);`
Rückgabewert	**bool** *(false=0, true=1)*	
Byteweise	`inp(<register>);`	`outp (<wert>, <register>);`
Rückgabewert	**unsigned char**	

Include the following header: *#include <avr/io.h>*

Example in C

```
//LED to 5V at PINA1;  SWITCH to GND at PINA0

#include <avr/io.h>

int main (void)
{
 outp(0xFE,DDRA);      //PortA: Pin0: Input,  Pin1..7: Output
 outp(0xFF,PORTA);     //PortA: Pin0: pull up,  Pin1..7: high = LED off
 while(1)
 {
      if(bit_is_set (PINA,0))            //check if PinA0 is high
      {
            cbi(PORTA,1);                //clear PinA1 = LED on
      }
      else
      {
            sbi(PORTA,1);                //set PinA1 = LED off
      }
 }
}
```

Design steps

Create C source code in Programmers Notepad

Create Makefile with Mfile and safe it to the folder of your source code

- „Main File Name...“ (Name of the main program without the extension(.c))
- „MCU Type” (ATmega16)
- “Debug format” (AVR-ext-coff)

in Programmers Notepad execute

Tools -> [WinAVR] Make All

Connect MCU board to PC

Start Atmel AVR Studio

Click "open" and choose the created .cof-File

Download should start automatically

Execution & Debugging

Use the following programs:

- Programmers Notepad
- Mfile
- Atmel AVR Studio

Useful Links

- http://www.mikrocontroller.net/wiki/AVR-GCC-Tutorial
- http://atmel.com/products/avr/
- Die Programmiersprache C. Ein Nachschlagewerk

  Regionales Rechenzentrum für Niedersachsen/Universität Hannover

  http://www.rz.rwth-aachen.de/computing/sw/rrzn/index.php
- C – Programmieren von Anfang an

  Helmut Erlenkötter, ISBN 3-499-60074-9
- Programming Embedded Systems in C and C++

  Michael Barr, ISBN 1-56592-354-5

➔use Windows calculater for 0x04 => 0b00000100 => 4