Exercise Sheet 1- Embedded Systems and Microcontrollers

Exercise 1: Questions

a) What is an embedded system?

- A computer system (CPU, memory, software, bus, peripherals)
- Which is integrated into another technical system (embedding system)
- And influences the embedding system such that it behaves in a desired way.

b) Source of requirements for an embedded system?

- Requirements are derived from the requirements of the embedding system.

!
exam

c) Embedded systems can be categorized into two groups.

(1) What are these categories?
(2) Characteristics?
(3) Which kind of hardware?
(4) Programming languages?
(5) Examples?

1)	**Product automation**	**Production automation**
2)	-Many identical units -Cost per unit is critical -Customers aren't experts -Hardware determines platform	-Often just one identical unit -Cost is less critical -Customers are close to being experts -Programming system is more important than the platform
(3)	Microcontrollers, FPGAs	Programmable logic control (PLC), industrial PCs, distributed control systems (DCS)
(4)	C/C++, assembler, VHDL, MATLAB/Simulink	Instruction list (IL), sequential function chart (SFC), structured text (ST), function block diagram (FBD)
(5)	-Car engine controller -Washing machine controller -Weather station	-Chemical plant controller -Assembly line controller

!
exam

d) What is a microcontroller?

- Microprocessor
- RAM
- Permanent memory

mandatory

- Digital I/O
- Analog I/O
- Other peripherals

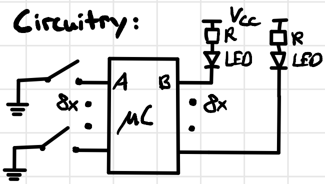
optional, but common

!
exam

Exercise 2: Digital I/O (ATmega16 microcontroller)

Description:

- 8 buttons connected to Port A and GND.
- 8 LEDs connected to Port B and VCC.



a) What are the registers that control these ports?

- Data direction: DDRA, DDRB
- Outputs / pull-up resistors: PORTA, PORTB
- Inputs: PINA, PINB

b) How should these registers be initialized?

- DDRA = 0b00000000 = 0x00
- DDRB = 0b11111111 = 0xFF
- PORTA = 0b11111111 = 0xFF
- PORTB = 0b11111111 = 0xFF
- PINA, PINB are read-only
- Remember: 0 - input, 1 - output
- Here: A 1 on PORTB will set off the respective LED by applying VCC on both sides. PORTA has to be 1 to have a defined voltage level of 0 on PINA when it is connected to GND by pressing the button.

c) Write a loop that allows control over the LEDs via the buttons:

(1) On a **1-to-1 basis** (pushing button 4 causes LED 4 to be lit).

```
while (1) {
    PORTB = PINA;
}
```

PINA has a value of 0 at the respective positions where the buttons are pushed. By copying this value to PORTB, the corresponding LEDs will be lit as the connection to VCC is now turned off inside the microcontroller.

(2) **Priority encoder:** Show the binary coding for the number of the highest button being pushed.

```
while (1) {
    int i;
    for (i=7; i>=0; i—) {
        if(~PINA & (1<<i)) {
            PORTB=~i;
            break;
        }
    }
}
```

!

Writing C code will not be part of the exam.

Starting from the highest i improves performance by not entering the if statement. The if statement checks whether the button at the ith position is pressed by shifting a to the ith position and comparing it to the inverted value in PINA with performing a bitwise AND operation. When the highest button being pushed is found, use a break statement to leave the for loop.

In the end, the highest pressed button is represented by the binary number shown by the LEDs that are lit.

d) What is bouncing? Implement a debouncing method.

- Bouncing is a short signal fluctuation before a signal change.

```
uint8_t first, fail;
uint8_t count = 3;
do {
    first = 1 & (PINA >> BTN_PIN);
    fail = 0;
    for(int i = 0; i < count; i++) {
        if(first != (1 & (PINA >> BTN_PIN))) {
            fail = 1;
            break;
        }
    }
} while (fail)
```

BTN_PIN is the index of PINA that we want to debounce by only accepting a change in the signal value that occurs for 3 consecutive times (count).

!

Writing C code will not be part of the exam.

Exercise 3: Interrupts and Polling

a) Choose interrupts or polling for the following scenarios.

(1) “Change input” button on a monitor - **Interrupts** as the signal is rare so polling would waste lots of CPU resources.

(2) Wireless Receiver of a garage opener - Scarcity of the signal makes **interrupts** attractive, but lots of noise could be a good reason to use **polling** instead to prevent erroneous openings for higher security (**undecidable**).

(3) Keyboard on a standard desktop - **Polling** as this is a frequent event and we don’t want the program to be interrupted at undesired points in time. Also, we want to be able to measure for how long a key was pressed.

(4) Temperature sensor of weather station - **Polling** as this allows for better planning of the overall resource usage. Interrupts might miss signal changes when they occur in a very high frequency.

b) When is an ISR called and how is it done?
Conditions: Global interrupt enable bit (I-bit), specific interrupt enable bit and specific interrupt flag need to be set (=1).

1) Hardware stores program counter (PC) on stack
2) Set global interrupt enable bit (I-bit) to 0
3) PC is set to the look-up table / vector table
4) PC jumps to specific interrupt service routine (ISR)
5) Context is stored by pushing it on the stack
6) ISR code is executed
7) Context is restored by popping it from the stack
8) Perform RETI (return from interrupt) instruction to restore the PC by popping it from the stack and setting the I-bit back to 1.

exam

Exercise 4: Timers and Counters

a) What is a counter? What is a timer?

- **Counter**: Hardware unit that counts external events (rising edges, falling edges, arbitrary edges)
- **Timer**: Special counter that only counts clock cycles (rising clock edges)

b) What components does timer 1 of ATmega16 have?

- Counter register TCNT1 (TCNT1H, TCNT1L)
- Compare register OCR1 (OCR1H, OCR1L)
- Input capture register ICR1 (ICR1H, ICR1L)
- Control register TCCR1 (TCCR1A, TCCR1B)

(Each sub register has 8 bits, so 16 bits in total)

exam

c) How is reading and writing of a 16-bit value made atomic?
Parallel reading and writing is used by reading the low byte first and automatically reading the high byte thereafter and writing to the high byte first and automatically writing to the low byte thereafter. TCNT1H serves as a buffer. However, ISR has to be disabled as there are always two operations involved.

d) What is a watchdog?
A timer that counts from an initial value to zero. When zero is reached, the microcontroller will be reset to resolve possible problems. The watch dog therefore has to be set back to its initial value repeatedly within the main loop of the program.

exam

e) Why might it be necessary to temporarily disable interrupts when reading 16-bit values?
Reading 16-bit values requires two cycles on an 8-bit platform. An interrupt occurring between those two cycles could alter the value inside the high byte register so that a wrong value is read in the end.

Exercise 5: Analog Devices

a) Analog devices on a ATmega16

- 4 PWM channels
- 8-channel-10-bit ADC (successive approximation converter - SAC)
- Analog comparator

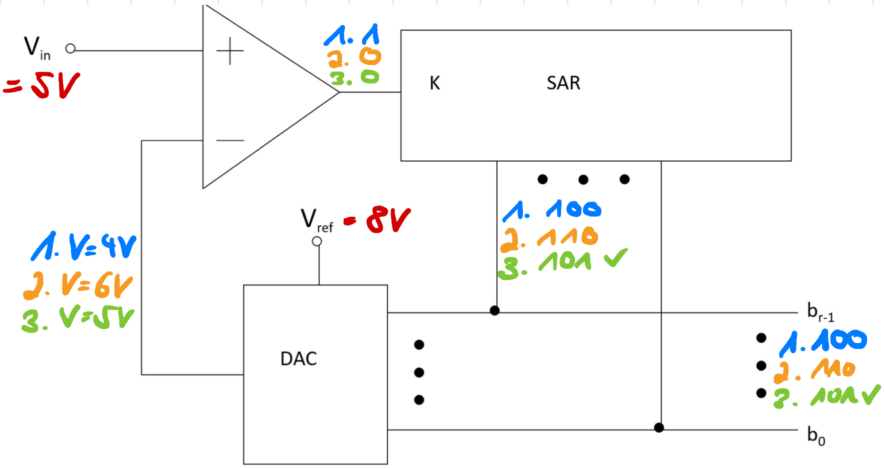
! Not part of the exam

b) What is PWM and how does it work?

Pulse width modulation produces a rectangular signal with a configurable frequency and high time. Higher frequencies can be achieved with it than by a signal generated by software. In total, PWM can be used to reduce the average voltage delivered on a PIN.

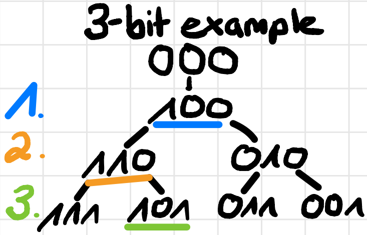
$$\text{average voltage} = \frac{\text{high time}}{\text{period}} \cdot V_{cc}$$

c) Sketch a successive approximation converter and explain how it works.



! exam

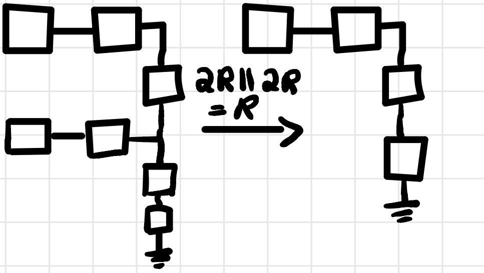
A successive approximation converter (SAC) performs a binary search through all possible signal levels. In each step, the value of one bit of the digital signal level is determined. All bits are initialized with 0. During the first step, the most significant bit of the output signal is set to 1, which is converted into half the reference voltage by the DAC. If the comparator produces a 1, the MSB stays at 1. Otherwise it would be set to 0. Now, the second most significant bit will be set to 1 and so on until the least significant bit is reached and the applied analog voltage is converted into a digital signal.



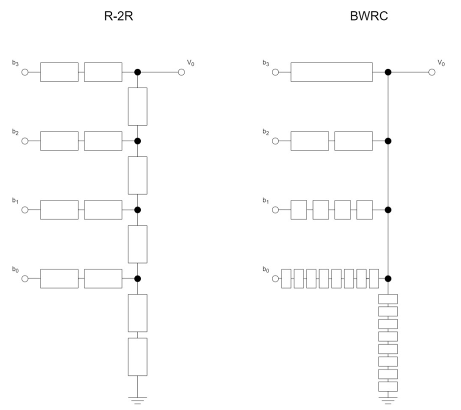
Problem: Oscillation possible if V_{in} = 6V e.g.

d) Imagine you only have 1-Ohm resistors available, which cost 10 cents each. Determine the minimum cost for a 4-bit R-2R and binary-weighted resistor circuit respectively with serial resistors.

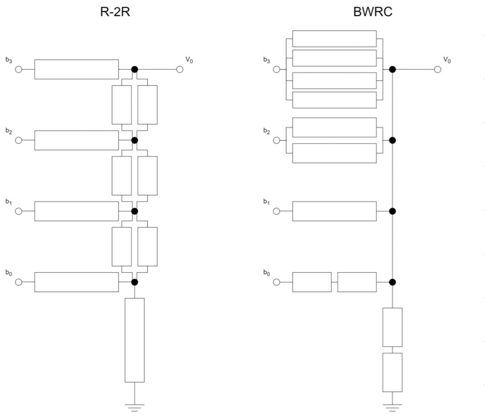
R-2R: 1.3 Euro (13 resistors)
BWRC: 2.3 Euro (23 resistors)



Ladder can be folded and unfolded



For parallel resistors, both result in the same amount.



e) What are the disadvantages of the binary-weighted resistor circuit?

- Many different types of resistors or just many resistors are needed for BWRC - Either bad quality of the produced voltage level due to imprecise resistors / high production costs if they need to be precise or high production cost due to higher amount of resistor needed in general for high bit resolutions.
- Usage of BWRC depends on the scale of the signal - Only applicable for very small bit widths (2 bits or less).

! exam