Department of Engineering Cybernetics

Examination paper for TTK4155 Industrial and Embedded Computer Systems Design

Academic contact during examination: Jo Arve Alfredsen

Phone: 90945805

Examination date: Thursday 2017-11-30

Examination time (from-to): 09:00 - 13:00

Permitted examination support material: D

Standard pocket calculator permitted.

Printed and handwritten material not permitted.

Other information:

Answers may be given in English or Norwegian

Read the text carefully. Each question may have several parts.

Answers should be concise.

Exam counts 50% of final grade.

Language: English

Number of pages (front page excluded): 6

Number of pages enclosed: 0

Informasjon om trykking av eksamensoppgave

Originalen er:

1-sidig □ **2-sidig** □

sort/hvit □ **farger** □

skal ha flervalgskjema □

Checked by:

Date Signature

Problem 1. (30 %)

a. Figure 1 shows the diagram of the 8 bit Timer/Counter 2 module of the Atmel AVR ATmega162 microcontroller. This timer module may be clocked by an external 32.768 kHz watch crystal to make it operate as a low-power real-time clock (RTC). Determine the value of the prescaler clock select bits CS22:20 that will give a cyclic timer 2 overflow interrupt (TOV2) at exactly 1 Hz.

b. The microcontroller in a) should be interfaced to a CAN controller featuring SPI and an interrupt line, constituting an embedded computer intended for a critical control task in a CAN network denoted as CAN node A. See figure 5 for the pinout of ATmega162 and show how you would connect the CAN controller to it.

c. Write a program in C/pseudo-code for node A using the TOV2 interrupt that:

    1. Keeps track of the number of days, hours, minutes and the total number of seconds the microcontroller has been running since last reset

    2. Transmits a CAN message every second containing the total number of seconds since last reset, serving as the "heart beat" signal of node A (assume you have a *CAN_Transmit(CAN_msg_t)* function available)

d. A different CAN node B with the same hardware as node A is set to monitor the running state of node A. Node B should trigger an alarm within two seconds if node A stops for some reason. Write a program in C/pseudo-code that implements this mechanism in node B.

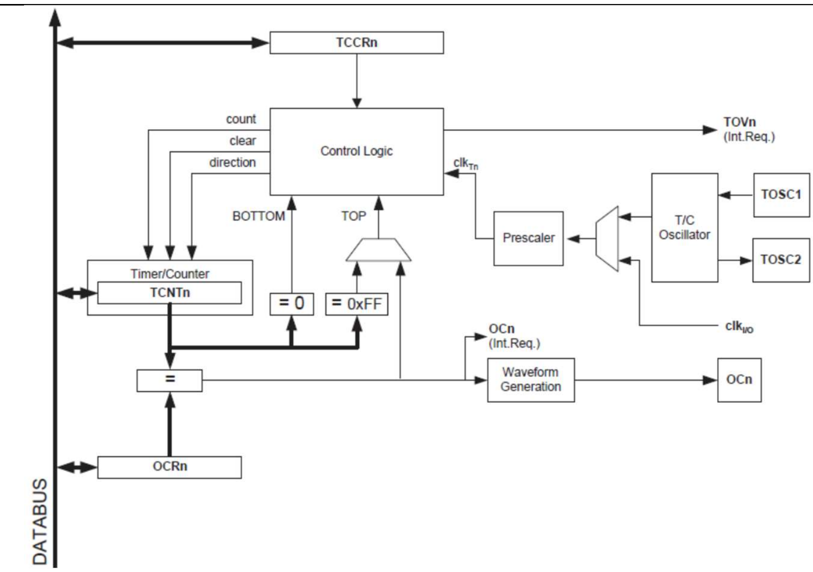
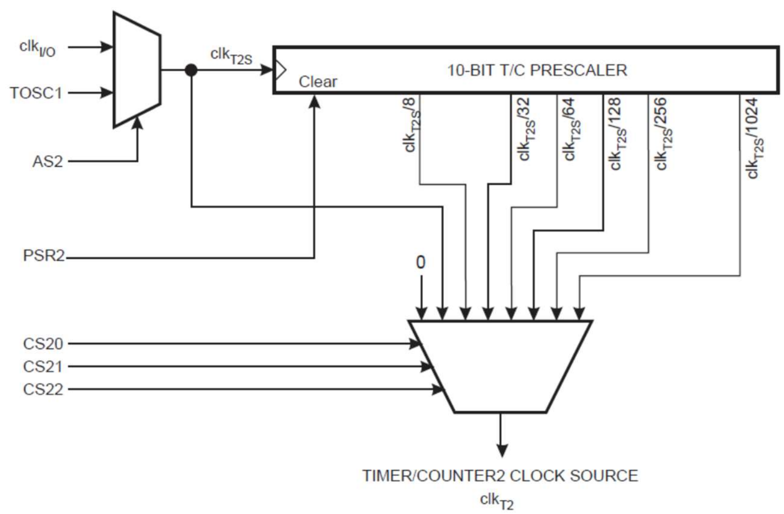


Figure 1. Diagram showing Timer/Counter 2 (n = 2) in the AVR ATmega162.

- Registers TCNT2, TCCR2 and OCR2 are all 8 bits wide.
- In normal mode, Timer/Counter 2 increments TCNT2 by one for each pulse of the clock signal clk_{T2} (0x00→0x01→…→0xFF→0x00→…). The interrupt signal TOV2 (timer overflow) is generated every time TCNT2 overflows from 0xFF to 0x00.
- The signal clk_{T2} is driven by a prescaled version of either the internal clock source $clk_{I/O}$, or an external clock crystal connected between pins TOSC1 and TOSC2.
- The clock signal clk_{T2} is selected by the control bits CS22:20 according to the following diagram:



CS22	CS21	CS20	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	clk_T2_S/(No prescaling)
0	1	0	clk_T2_S/8 (From prescaler)
0	1	1	clk_T2_S/32 (From prescaler)
1	0	0	clk_T2_S/64 (From prescaler)
1	0	1	clk_T2_S/128 (From prescaler)
1	1	0	clk_T2_S/256 (From prescaler)
1	1	1	clk_T2_S/1024 (From prescaler)

Problem 2. (30 %)

a. Assume that node A introduced in Problem 1 has an unregulated 6-12 V voltage supply available through an extra wire pair in the CAN cable. To function properly, node A requires both a stable 5 V supply for the microcontroller as well as a 15 V supply for powering an on board ultrasonic transducer.
Explain how you would design a proper power supply for node A and specify the component types you would use.

b. Node A employs the ultrasonic transducer described in Figure 2 to monitor the fluid level of a tank. Assume the ATmega162's system clock is driven by an external 4 MHz crystal oscillator and use the output compare match unit B of Timer/Counter 1 module described in Figure 3 to time the 10 ms measurement interval of the transducer. The OC1B pin can be set and cleared on match to generate the digital trigger signal.
What value should be written to the OCR1B register?

c. The ATmega162's Analog Comparator module described in Figure 4 can be used to trigger an input capture event in Timer/Counter 1. Explain how you can exploit this mechanism together with a few external resistors to make level measurements with the ultrasonic transducer.

d. Write a program in C/pseudo-code for node A that implements the above mechanisms to make a level measurement and transmit it as a CAN message every second.

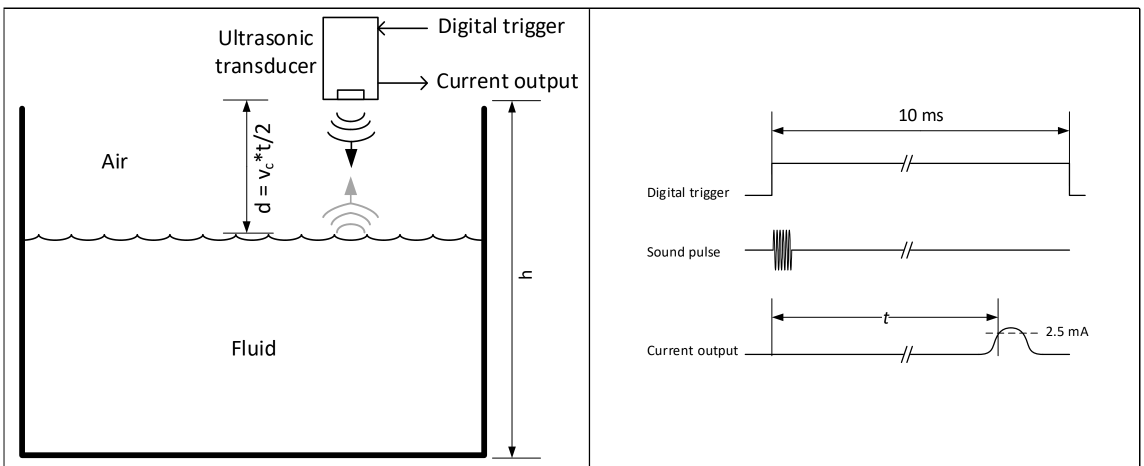


Figure 2. Ultrasonic transducer measuring the fluid level of a tank.

When triggered by a digital signal, the transducer emits a short pulse of sound and then immediately starts to sense sound energy reflected back from objects along the path of the sound wave. The time it takes for the sound to travel from the transducer to the object and back, represents the distance to the object (time-of-flight measurement). The level of the tank can then be calculated as $l = h - d = h - v_c*t/2$, where t is the time-of-flight of the ultrasonic signal and $v_c = 340$ m/s is the speed of sound.

The transducer outputs a current signal proportional to the rms-amplitude (root-mean-square) of the reflected signal during the 10 millisecond time interval immediately after emission of the sound pulse. Calibration tests have shown that a current output signal greater than 2.5 mA indicates reflection off the fluid surface at all relevant distances.

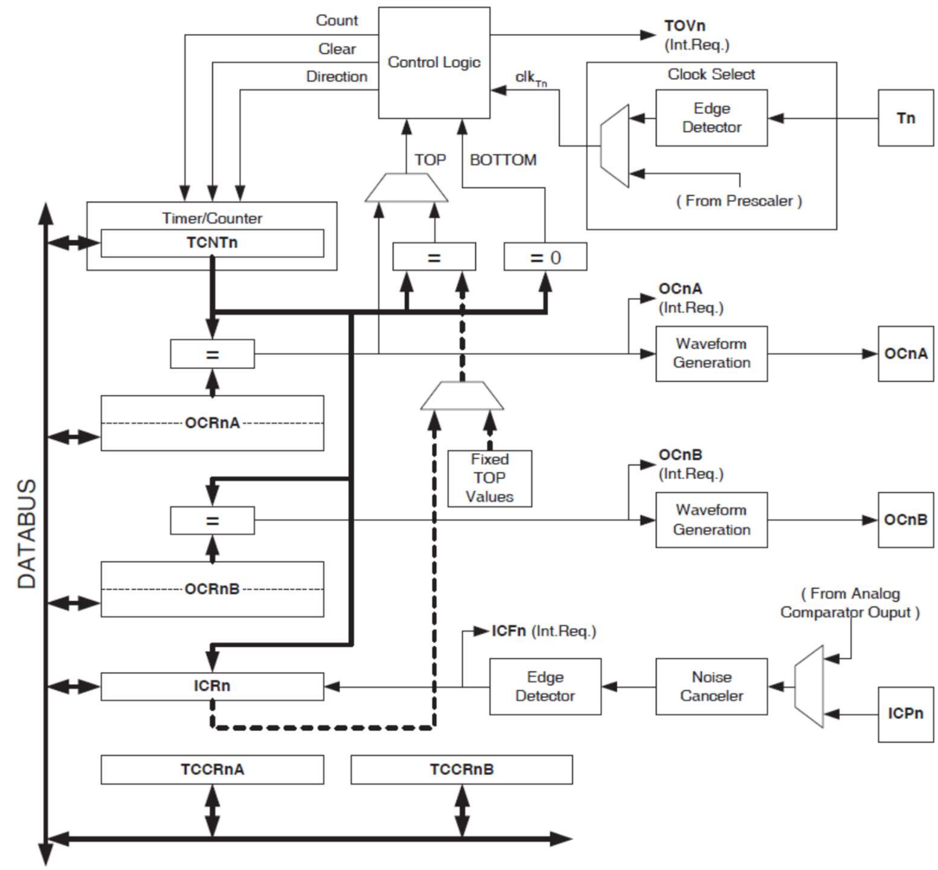


Figure 3. Diagram showing Timer/Counter 1 and 3 in the AVR ATmega162.

- In Timer/Counter n (n = 1 or 3), the TCNTn, OCRnA, OCRnB and ICRn registers are all 16 bit.
- The Timer/Counter can be driven by internal or external clock sources, either directly from the system clock, via a prescaled (divided) version of the system clock, or via a clock signal on pin Tn. The selectable prescaler values are 8, 64, 256 or 1024.
- In *Normal* mode, TCNTn increments by one for each clock pulse (0x0000→0x001→…→0xFFFF→0x0000→…). The interrupt signal TOVn (timer overflow) is generated every time the TCNTn overruns from 0xFFFF to 0x0000.
- The ICRn (input capture register) is used in normal mode to capture/timestamp external events in terms of rising and falling edges on the ICPn pin or a trigger signal from the microcontroller's internal Analog Comparator unit. When a rising or falling edge is detected on the ICPn pin or comparator output, the current value of the timer register TCNTn gets loaded immediately into ICRn and the input capture interrupt signal ICFn is generated. The ICESn bit of the timer control register TCCRnB selects which edge of the ICPn pin signal triggers the capture (0 = falling edge, 1 = rising edge).
- The OCRnA and OCRnB are two output compare registers that are used to generate the interrupt signals OCnA and OCnB as well as setting or resetting the OCnA and OCnB pins when the timer TCNTn matches their content.
- When Timer n is set to *Clear Timer on Compare (CTC) mode*, TCNTn will be cleared to zero when a match with the content of OCRnA is detected and the interrupt signal OCnA will be generated and the OCnA pin will be toggled.
- When Timer n is set to *Fast PWM mode*, TCNTn counts from BOTTOM (0x0000) to TOP and then restarts from BOTTOM again. TOP can be set to fixed values 0x00FF, 0x01FF, 0x03FF or the content of OCRnA or ICRn. The OCnx pin is set on a compare match between TCNTn and OCRnx, and cleared at TOP. A TOVn interrupt is generated when TCNTn rolls over from TOP to BOTTOM.

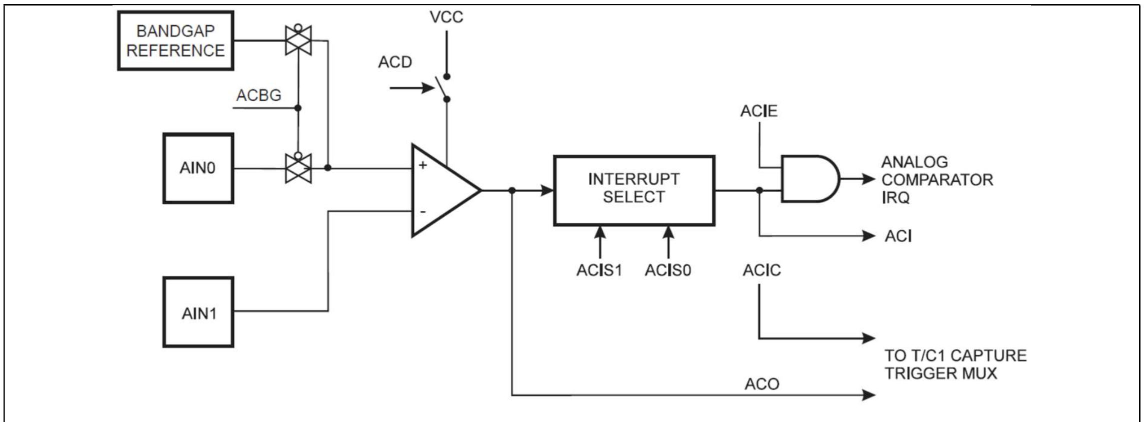


Figure 4. Diagram of the internal Analog Comparator module of the AVR ATmega162.

The Analog Comparator compares the input voltages on the positive pin AIN0 and negative pin AIN1. When the voltage on the positive pin AIN0 is higher than the voltage on the negative pin AIN1, the Analog Comparator Output, ACO, is set. In addition, the comparator can trigger a separate interrupt (ANA_COMP), exclusive to the Analog Comparator. The user can select interrupt triggering on comparator output rise, fall or toggle by setting the control bits ACIS1 and ACIS0 to 11, 10 or 00 respectively. The comparator's output can also be set to trigger the Timer/Counter1 Input Capture function by setting the ACIC control bit. When the ACBG control bit is set, a fixed bandgap reference voltage of 1.10V replaces the positive input AIN0 to the Analog Comparator.

Problem 3. (40 %)

Node A from Problem 1 and 2 should be developed further into a fully functional industrial sensor device featuring more memory and a display. Sensors for temperature and humidity should also be included as sound velocity depends on these parameters. The device's detailed additional specification is given as follows:

- The system should be built up around a microcontroller of type AVR ATmega162 as shown in Figure 5.
- ATmega162 features a relatively small amount of internal SRAM and requires inclusion of an extra external 32 kByte SRAM.
- The temperature and humidity sensors measure temperature in the range -15 – 50 °C and relative humidity in the range 0 – 100 %, respectively, and outputs analog signals in the range 0 – 5 V.
- To read sensor signals you have a 2-channel AD converter with 16-bit resolution and a parallel bus interface available. The AD converter accepts voltage signals in the range 0 – 5 V. The AD converter has an internal address space of 6 bytes. The AD conversion is started when an 8-bit control word is written to the control register at address 0 in the AD converter. Address 1 is reserved for status information (status register). When the conversion has finished, the result (16-bit) will be stored at the addresses 2n+2 and 2n+3, where $n \in [0, 1]$ is the channel number. The interrupt pin of the AD converter is then pulled low. Note: the AD converter features a parallel bus interface. In other words, it must be interfaced as a pure memory mapped I/O unit using the external address and data bus of the microcontroller.
- The display has 2 lines x 20 characters and features a parallel bus interface. The display operates as a pure memory mapped I/O and can be accessed by writing/reading to/from a linear 40 byte memory inside the display where the addresses corresponds to the character

positions on the display (first character on the first line has address 0, first character on the second line has address 20). Moreover, the display features ten 8-bit status- and control registers, making the display require a total of 50 bytes of the address space.

a. Describe the system in terms of a *high-level* block diagram (a detailed circuit schematic not requested here). Read the specification above in detail, focus on identifying and drawing the individual function blocks/modules that together make up the system, and then indicate the interface between them using simple arrows and labels (no detailed bus/signal connections requested here). If deemed necessary, make your own reasonable assumptions that will make the system function properly.

b. Assume that the base address of the display should put at 0x4000. Explain how you would organize the address space of the computer in the simplest possible way, and derive the associated decoding logic (remember that the 1280 lowest addresses (0x0000 - 0x04FF) of the ATmega162 are reserved).

c. Draw and explain a circuit schematic that shows how the components of the system are connected together (the details can be limited to central signal lines, but the width of all buses and which ports/bit signals are connected to must be shown). Specify the circuits and signals you find necessary to add.

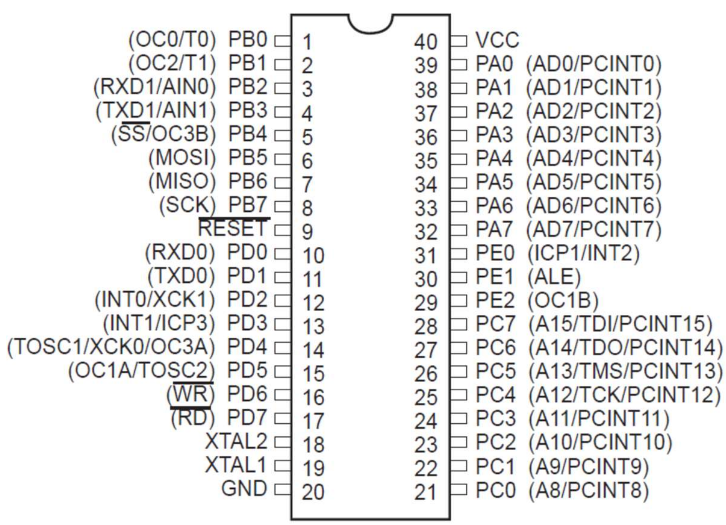


Figure 5. Atmel AVR ATmega162.