

182.694 Microcontroller VU

FAKULTÄT FÜR **INFORMATIK**

Martin Perner
SS 2017

Featuring Today:
Recitation for second Exam
Introduction Application 2

Weekly Training Objective

- Already done
 - 3.5.2 Noise
 - 3.5.3 Prescaler and accuracy *
 - 3.7.5 Dynamic memory analysis
- Until Exam
 - 3.4.1 Input capture
 - 3.6.3 SPI *
 - 3.7.2 Watchdog *
- Next week
 - 4.2.1 UART to GLCD †
 - 4.2.2 Keypad
 - 4.2.3 Debounced Buttons *

ÖH Wahl 2017

Go Vote!

The Hochschülerschaft election starts tomorrow and it would be great if you take the time to vote.

You can get further information, e.g., at <https://fsinf.at/wahl-2017>.

Important

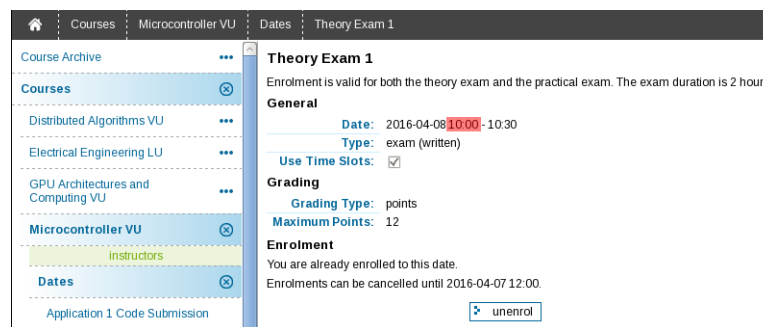
Registration for the Exam

- Do not forget to register for the exam, until Thursday 12:00, in myTI!
- You only need to register for the theory exam, the registration for the practical exam will be done automatically by us.
- The whole exam will last 2 hours.

There are enough slots for everyone, . . .

. . . but myTI will keep the number of slots as small as possible:
A new slot will only open when the previous one is full.

Important



The screenshot shows a web interface for a course named 'Microcontroller VU'. The left sidebar lists various course sections, with 'Microcontroller VU' selected. The main content area displays details for 'Theory Exam 1'. The 'Date' is 2016-04-08 10:00 - 10:30, and the 'Type' is 'exam (written)'. The 'Use Time Slots' checkbox is checked. The 'Grading' section shows 'Grading Type: points' and 'Maximum Points: 12'. The 'Enrolment' section states 'You are already enrolled to this date.' and 'Enrolments can be cancelled until 2016-04-07 12:00.' There is an 'unenrol' button at the bottom right.

Figure: Your Exam start-time. Please be sure to check to current exam, and not the previous ones!

Important

Toilets/WC

While there is a toilet in the lab, we highly recommend to use the public toilets before the exam.

It is your working time.

Exam Flow

- 1 Be at the glass door in front of the lab **before** your timeslot starts!
- 2 Keep your student-ID ready. We will check it multiple times!
- 3 A supervisor will fetch you there and bring you to the theory exam.
- 4 You will have approx. 25 minutes for the theory exam.
- 5 After that, a supervisor will bring you to the seminar room. There you will receive the exam paper with the task descriptions and have approx. 25 minutes of preparation time for the practical exam.
Please wait close to the building door, the group before you has to use the stairs!
- 6 Then a supervisor will bring you back to the lab for the practical exam (approx. 50 minutes working time, there will be a clock on the big screens in the lab).
The supervisor will assign you to a PC!
- 7 After you finished the practical exam you are only allowed to leave the room iff the supervisor **allows it!**

Attention

Cheating will not be tolerated!

- Usage of mobile phones/smart watches is forbidden until you leave the lab, i.e., throughout the entire exam!
- Talking to your colleague is **not allowed**.
- You are **not allowed** to use your own notes, paper, calculator, ... during the exam.
 - Everything you need is provided by us.
 - You can, of course, use your own pen and bring something to drink.
 - If you need a dictionary, you can use one.

Available Material

We will provide you with the following material during the ...

- theory exam and the preparation time: everybody will have a printed version of
 - BigAVR6 Schematic,
 - AVR Instruction Set Manual,
 - Atmega1280 Manual,
 - avr-libc Reference Manual, and
 - a simple calculator.
- practical exam: you have to share the printed version of the above manuals with a colleague, but the digital versions are available to you.
- There will be **no** oscilloscope available during the test! But you can still use LEDs to debug your program.

Theory Exam

- The questions are in English only!
- 8 questions/statements with 3 answers/statements each.
- Every answer can be checked either true or false.
 - If you checked correctly, you get 0.5 point.
 - If you checked incorrectly, you get -0.5 point.
 - You may also not check true or false (provide no answer), then you get 0 points.
- The grading is done cumulative, i.e., you can get between -1.5 and 1.5 points for a questions.
- Positive and negative points will be considered for the total points. But we will cap a negative total to 0 points.

Practical Exam

- The task descriptions are in English only!
- The environment used for the exam is the default/plain setting of the lab; without any customizations you did, e.g., editor configuration!
- Obviously, there is no internet access.
- If you have a problem or something is unclear, ask a supervisor. They will try to give you *hints*, if they have the time for it.
- This time, there will be one programming task and one debugging task.

Practical Exam – Programming Task

Organisation

- When you are done with the task: knock
A supervisor will come to check if your solution is correct. If it is, the supervisor will mark it as solved on the exam paper.
- You must deliver the task **before** the time is over!
⇒ do not wait for the last minutes to deliver your task!

Provided Code

- You will be provided with a program skeleton. The same that you have on the task description.
- You will also have access to a binary version of a solution.

Practical Exam – Debugging Task

- This task will be graded offline, i.e., after the exam, and there are partial points to gain!
- The last version in the working directory will be graded.
- The supervisors will **not** check your code during the exam.

Provided Code

- You will be provided with a program. The same that you have on the task description.
- You will also have access to a binary version of a solution.

Details

- The program has 5 bugs/errors in 3 categories/difficulties ($2 \times 0.5\text{pts}$ / $2 \times 1\text{pt}$ / $1 \times 2.5\text{pts}$).
- The errors/bugs have an effect on the behavior of the program.
- The error behavior/effect of the hardest bug will be hinted in the task description.
- The fixes are designed to be done in at most 3 lines (typically one line).

Practical Exam

Remember – Programming Task

The grading for the programming task is binary:
Do not spend all of your time trying to get one task running

Remember – Debugging Task

Points will only be rewarded for minimal fixes!
We will not check earlier version in a SCM!
Rewriting the program to satisfy the specification is not an accepted solution!

Viewing (Einsichtnahme)

- Viewing for the theory exam is by appointment (email).
- Viewing for the practical exam is by email only:
 - Within a week after the exam write us an email in which of your task you are interested in.
 - We will evaluate them and send you an explanation of your mistakes per email.
 - Depending on our workload our response time may vary between a day and 1-2 weeks.
 - The solution for the Debug Task will be online.

What will be asked in the Theory Exam?

We will ask you questions about the microcontroller, C example code (bit set/reset stuff), about our lab environment (avr-libc, ...), the lecture script and the lecture slides.

What will not be asked

Details about ASM code.

Theory Exam

Word of Warning

Usually, this theory exam is extremely underestimated!

Point average in this years first theory exam was ≈ 8.05 .

Last years first theory exam had an average of ≈ 7.81 whereas the second only had ≈ 6.25 !

What will be asked in the Practical Exam?

- Callback (function pointers!)
- timer/adc/...
- There are Makefiles this time

Attention – Checker

Checker

There will be a tool linked in your application which checks if you have only touched registers that need to be touched AND that you have only modified bits that need to be modified!

Attention – Checker

There will be one of the following outputs on the LCD

- Registers OK
- too many 1/0
- too many 1
- too many 0

The “too many” mean that, in all monitored registers, some bits are set incorrectly, and the majority of the wrong bit have the mentioned value. There is no information which register or bits are incorrect!

Attention – Checker

Warning

If your solution does not get an OK from that tool, your solution is considered incorrect ⇒ no points!

Default values

You **cannot** assume that any bit is in its default value!
You **have** to (re)set every bit you need on a certain value.

Guide

We provide a guide on the homepage which lists a few registers/control-bits which will be monitored. This guide is located in the Misc subdirectory.

Debug Task

The register checker is not used in the debug task!

Pitfalls and Hints

Calculations

Use defines if you need to calculate some values.

Add parentheses, errors due to operator precedence are annoying and hard to find!

Timer values

We will check that you have the correct timer values (checker). Check that you understand how the values are calculated!

The values need to be exact!

Pitfalls and Hints

avr-libc

The avr-libc is comprehensive. There are many helper functions defined (e.g., baudrate), and attributes available, which can help you implement the solution.

Check the hardware

This time you may need to set up some wiring on your own!

The supervisors may not have the time to check if every board is set to the default settings!

Check the jumpers and switches!

Knowledge of Components

There are other components/modes of the microcontroller which have not been part of Application 1, but were used in the weekly training objectives and thus can be part of the exam!

Application 1

- There were 40 submissions.
- We will need a few weeks for the corrections!
- For a viewing, just drop us an e-mail after the results are published. Please, include some dates which work for you!

Defect Hardware in the Lab?

We can only fix broken stuff we know about, thus:

- send an E-Mail to mc-leitung@tilab.tuwien.ac.at.
- add a handwritten note to the affected piece, for your colleagues.
- If you are unsure, there are enough demo tools (and other boards) to verify if a certain hardware is broken.

Questions?

Application 2

- Annoying Lunar Lander
 - Specification will online before the exam
 - Additional hardware will be in the Lab after the exam.
- Specification
 - given module interfaces/modules
- Log time (for protocol)

Hints

- Read the TinyOS Manual!
- Try to get a feeling for TinyOS before you start to program the Application!
- The Tutors are still there for your questions!
- In next week will be the second TinyOS lecture.
If there is a topic you want to have repeated/covered, just write an e-mail.

Questions?