



Computer Hardware Engineering (IS1200)

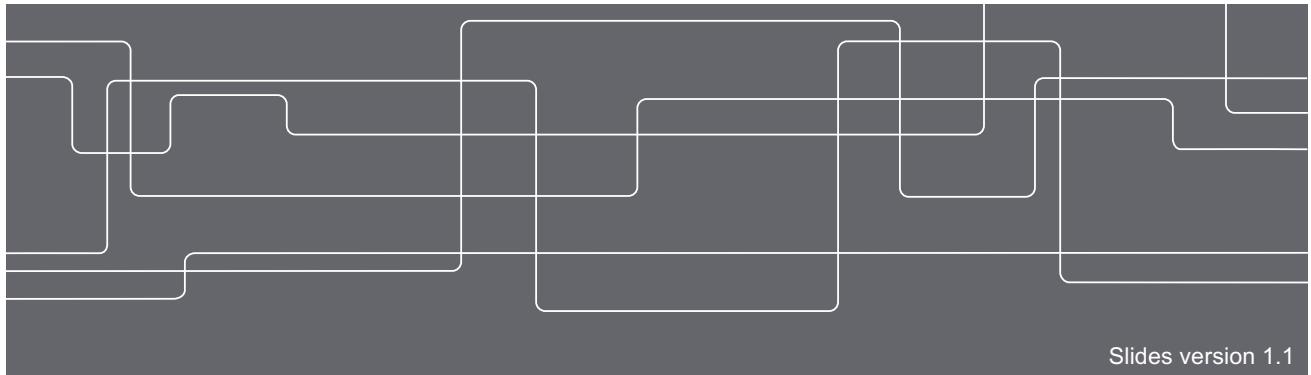
Computer Organization and Components (IS1500)

Spring 2019

Lecture 14: Course Summary and Project Awards

David Broman

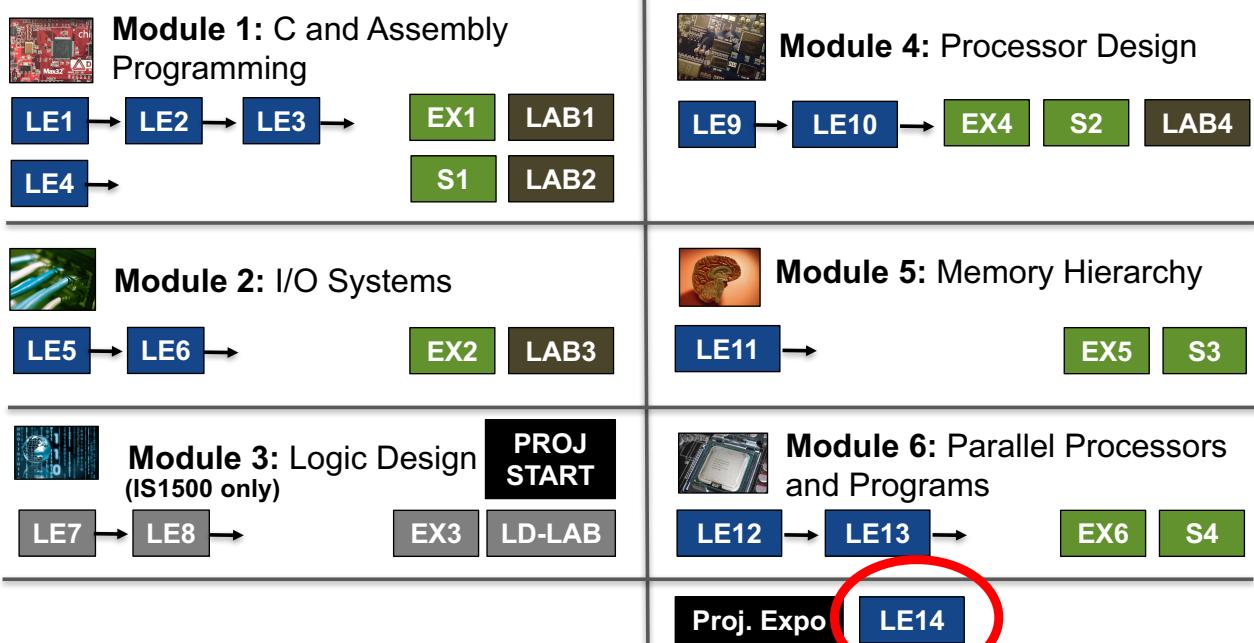
Associate Professor, KTH Royal Institute of Technology



2



Course Structure



David Broman
dbro@kth.se

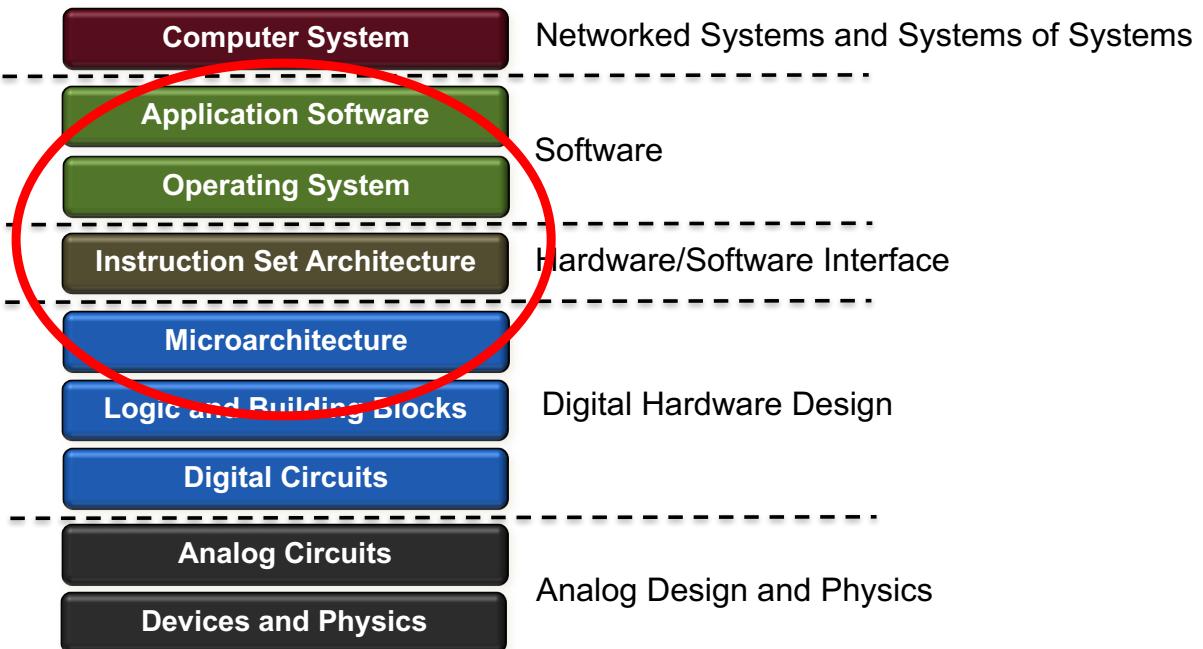
Part I
Exam Structure
and Grading

Part II
Study
Advice

Part III
Key Concepts and
Previous Exam Questions



Abstractions in Computer Systems



David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

Part II
Study
Advice

Part III
Key Concepts and
Previous Exam Questions



Agenda

Part I
Exam Structure and Grading

Part II
Study Advice

Part III
Key Concepts and Previous Exam Questions



Project and Awards!

David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

Part II
Study
Advice

Part III
Key Concepts and
Previous Exam Questions



Part I

Exam Structure and Grading



David Broman
dbro@kth.se



Part I
Exam Structure
and Grading

Part II
Study
Advice

Part III
Key Concepts and
Previous Exam Questions



Examined Course Parts



1.5hp ANN1. LA-LAB, Logic Design
Grades P and F
(IS1500 only, not IS1200)

4.5hp LAB1. Labs 1,2,3,4 and
Mini Project
Grades P and F

3hp TEN1. Written Exam (tenta)
Grades A, B, C, D, E, FX, F

David Broman
dbro@kth.se



Part I
Exam Structure
and Grading

Part II
Study
Advice

Part III
Key Concepts and
Previous Exam Questions



Written Exam

Written Exam (Tenta)

- March 15, 8-13, 2019, (5h)
- Retake exams June 2019, January 2020
- Allowed aids: One sheet of handwritten A4 paper (both sides) with notes.

The exam has two parts

Part I: Fundamentals

- Max 40 points.
- 8 points for each of the 5 modules.
- Short questions with short answers.

Part II: Advanced

- Max 50 points.
- Comprehensive questions.
Discuss, analyze, construct.

Grading of Exam

- To get a pass grade (A, B, C, D, or E), it is required to get at least 2 points on each module and in total 30 points on Part I (including bonus points).

Grading scale:

- A: 41-50 points on Part II
- B: 31-40 points on Part II
- C: 21-30 points on Part II
- D: 11-20 points on Part II
- E: 0-10 points on Part II
- FX: At least 30 points on Part I, and at most one module with less than 2 points.
- F: otherwise

To get A or B, you also need to have done an advanced project.

David Broman
dbro@kth.se

Part I

Exam Structure
and Grading

Part II

Study
Advice

Part III

Key Concepts and
Previous Exam Questions



Part II

Study Advice



David Broman
dbro@kth.se

Part I

Exam Structure
and Grading

Part II

Study
Advice

Part III

Key Concepts and
Previous Exam Questions



Some advices on how to study for the exam...



Lecture slides

Go through all slides very carefully. Make sure that you understand **every** key concept.



Read the books and check references

Read **selected parts** of the course book, according to the reading guidelines. Check both Patterson & Hennessy and Harris & Harris.



Exercises and old exams

Practice on the exercises and the last exams. Solve the exercises **yourself**. Look at the solution **after** you have made a solution.



A4 sheet (handwritten notes) and MIPS Sheet

Summarize. Make notes on formulas, structures, and key concepts. Make sure you understand every detail of the MIPS reference sheet.



Write questions and answers in Canvas

Ask questions and try to help your fellow students by making a Q&A in the discussion forum on Canvas!

David Broman
dbro@kth.se

Part I

Exam Structure
and Grading



Part II

Study
Advice

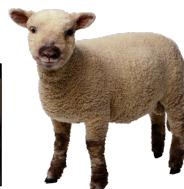
Part III

Key Concepts and
Previous Exam Questions



Part III

Key Concepts and Previous Exam Questions



David Broman
dbro@kth.se

Part I

Exam Structure
and Grading



Part III

Key Concepts and
Previous Exam Questions

Part II

Study
Advice

Module 1

C and Assembly Programming



Key Concepts

- Variables, if-statements, and expressions
- Loops (while and for)
- ISA
- Registers
- Instructions
- Translating from C to ASM and back
- Pointer and Arrays
- Two's complement and Sign extension
- Machine instruction encoding
- R, I, and J types
- Stack, Parameters and Arguments

David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

Part II
Study
Advice

 **Part III**
Key Concepts and
Previous Exam Questions

(a) **English:** What is the binary machine code representation of the MIPS instruction

1b \$s1, -67 (\$t2)

Answer as a 32-bit binary number. Note that a page with the structure of the encoding of MIPS is available at the end of the exam. (4 points)

David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

Part II
Study
Advice

 **Part III**
Key Concepts and
Previous Exam Questions



MIPS Reference Sheet

INSTSTRUCTION SET (SUBSET)

Name (format, op, funct)	Syntax	Operation
add (R,0,32)	add rd,rs,rt	reg(rd) := reg(rs) + reg(rt);
add immediate (I,8,na)	addi rt,rs,imm	reg(rt) := reg(rs) + signext(imm);
add immediate unsigned (I,9,na)	addiu rt,rs,imm	reg(rt) := reg(rs) + signext(imm);
add unsigned (R,0,33)	addu rd,rs,rt	reg(rd) := reg(rs) + reg(rt);
and (R,0,36)	and rd,rs,rt	reg(rd) := reg(rs) & reg(rt);
and immediate (I,12,na)	andi rt,rs,imm	reg(rt) := reg(rs) & zeroext(imm);
branch on equal (I,4,na)	beq rs,rt,label	if reg(rs) == reg(rt) then PC = BTA else NOP;
branch on not equal (I,5,na)	bne rs,rt,label	if reg(rs) != reg(rt) then PC = BTA else NOP;
jump and link register (R,0,9)	jalr rs	\$ra := PC + 4; PC := reg(rs);
jump register (R,0,8)	jr rs	PC := reg(rs);
jump (J,2,na)	j label	PC := JTA;
jump and link (J,3,na)	jal label	\$ra := PC + 4; PC := JTA;
load byte (I,32,na)	lb rt,imm(rs)	reg(rt) := signext(mem[reg(rs) + signext(immm)] _{7,0});
load byte unsigned (I,36,na)	lbu rt,imm(rs)	reg(rt) := zeroext(mem[reg(rs) + signext(immm)] _{7,0});
load upper immediate (I,14,na)	lui rt,imm	reg(rt) := concat(immm, 16 bits of 0);
load word (I,35,na)	lw rt,imm(rs)	reg(rt) := mem[reg(rs) + signext(immm)];
multiply, 32-bit result (R,28,2)	mul rd,rs,rt	reg(rd) := reg(rs) * reg(rt);
nor (R,0,39)	nor rd,rs,rt	reg(rd) := not(reg(rs) reg(rt));
or (R,0,37)	or rd,rs,rt	reg(rd) := reg(rs) reg(rt);
or immediate (I,13,na)	ori rt,rs,imm	reg(rt) := reg(rs) zeroext(immm);
set less than (R,0,42)	slt zd,rs,rt	reg(rd) := if reg(rs) < reg(rt) then 1 else 0;
set less than unsigned (R,0,43)	sltu rd,rs,rt	reg(rd) := if reg(rs) < reg(rt) then 1 else 0;
set less than immediate (I,10,na)	slti rt,rs,imm	reg(rt) := if reg(rs) < signext(immm) then 1 else 0;
set less than immediate	sltiu rt,rs,imm	reg(rt) := if reg(rs) < signext(immm) then 1 else 0;
unsigned (I,11,na)		
shift left logical (R,0,0)	sll rd,rt,shamt	reg(rd) := reg(rt) << shamt;
shift left logical variable (R,0,4)	sllv rd,rt,rs	reg(rd) := reg(rt) << reg(rs _{4,0});
shift right arithmetic (R,0,3)	sra rd,rt,shamt	reg(rd) := reg(rt) >> shamt;
shift right logical (R,0,2)	srl rd,rt,shamt	reg(rd) := reg(rt) >> shamt;
shift right logical variable (R,0,6)	srlv rd,rt,rs	reg(rd) := reg(rt) >> reg(rs _{4,0});
store byte (I,40,na)	sb rt,imm(rs)	mem[reg(rs) + signext(immm)] _{7,0} := reg(rt);
store word (I,42,na)	sw rt,imm(rs)	mem[reg(rs) + signext(immm)] _{15,0} := reg(rt);

REGISTERS

Name	Number	Description
\$zero	0	constant value 0
\$at	1	assembler temp
\$v0	2	function return
\$v1	3	function return
\$a0	4	argument
\$a1	5	argument
\$a2	6	argument
\$a3	7	argument
\$t0	8	temporary value
\$t1	9	temporary value
\$t2	10	temporary value
\$t3	11	temporary value
\$t4	12	temporary value
\$t5	13	temporary value
\$t6	14	temporary value
\$t7	15	temporary value
\$s0	16	saved temporary
\$s1	17	saved temporary
\$s2	18	saved temporary
\$s3	19	saved temporary
\$s4	20	saved temporary
\$s5	21	saved temporary
\$s6	22	saved temporary
\$s7	23	saved temporary
\$t8	24	temporary value
\$t9	25	temporary value
\$k0	26	reserved for OS
\$k1	27	reserved for OS
\$gp	28	global pointer
\$sp	29	stack pointer
\$fp	30	frame pointer
\$ra	31	return address

Part I

Exam Structure
and Grading

David Broman
dbro@kth.se

Part II

Study
Advice

Part III

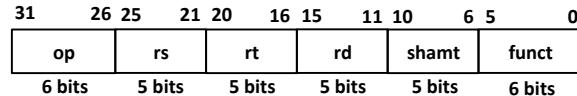
Key Concepts and
Previous Exam Questions

14

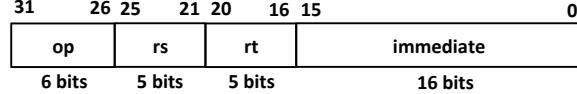


INSTRUCTION FORMAT

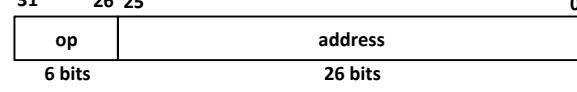
R-Type



I-Type



J-Type



David Broman
dbro@kth.se

Part I

Exam Structure
and Grading

Part II

Study
Advice

Part III

Key Concepts and
Previous Exam Questions



- Statement 1:

```
int x = 5;
int y = 6;
int *p = &x;
*p = x + y;
p = &y;
*p = x + y;
printf("%d,%d", x, y);
```

What is printed to standard output?

Answer: 11,17

David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

Part II
Study
Advice

 **Part III**
Key Concepts and
Previous Exam Questions



Module 2

I/O Systems



Key Concepts

- Memory Mapped I/O
- Volatile keyword in C
- GPIO
- Timers
- Exceptions and Interrupts
- Buses (synchronous vs. asynchronous)
- DMA

David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

Part II
Study
Advice

 **Part III**
Key Concepts and
Previous Exam Questions



- (a) **English:** Assume that an external interrupt occurs, a program A is preempted, and the program counter is changed so that an interrupt handling routine is executed. Explain shortly how it is possible to continue to execute program A correctly at the program point where it was interrupted. (4 points)

David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

Part II
Study
Advice

 **Part III**
Key Concepts and
Previous Exam Questions



- Statement 2:
English: A memory-mapped general-purpose I/O (GPIO) pin can be configured to be either an output or input port, by writing a configuration parameter to a specific memory address that is dedicated for configuring this GPIO pin.
- Statement 3:
English: Direct memory access (DMA) is a good alternative to instruction level parallelism (ILP) because it enables multiple instructions to be fetched by the datapath and executed in parallel since the code is read directly from memory.
- Statement 4:
English: Declaring a pointer `volatile` in C (as in the example below) means that the pointer itself is volatile and may change to point to a different address at any point in time.

```
volatile int* x = (volatile int*) 0xff88;
```

David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

Part II
Study
Advice

 **Part III**
Key Concepts and
Previous Exam Questions



Module 3

Logic Design

(only for IS1500)



Key Concepts

- Combinational Logic
- Sequential Logic
- Two's complement
- Boolean algebra
- Proof by Perfect Induction
- Truth Tables
- Multiplexers
- Decoders
- Carry Propagate Adder
- Latch vs. Flip-Flop
- Register and Register File

David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

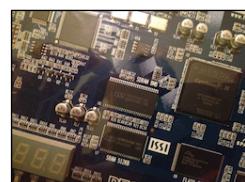
Part II
Study
Advice

 **Part III**
Key Concepts and
Previous Exam Questions



Module 4

Processor Design



Key Concepts

- ALU
- Datapath
- Control Unit
- Single cycle processor
(details of signals)
- Performance Analysis
- Critical Path
- CPI and IPC
- Pipelining (concept)
- Pipelined datapath
- Data Hazards
- Control Hazards
- Forwarding and Stalling

David Broman
dbro@kth.se

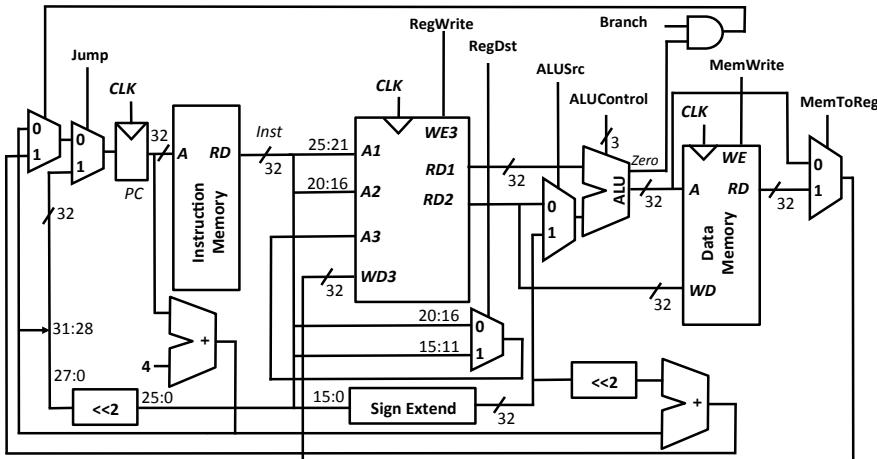
Part I
Exam Structure
and Grading

Part II
Study
Advice

 **Part III**
Key Concepts and
Previous Exam Questions



- (a) **English:** Consider the figure below that shows the datapath for a single-cycle MIPS processor. Assume that the current instruction that is executing is `slt $t0,$t1,$t3`. What are then the values of signals *MemToReg*, *ALUSrc*, *Branch*, and *A3*? (4 points)



David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

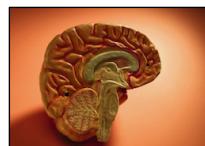
Part II
Study
Advice

Part III
Key Concepts and
Previous Exam Questions



Module 5

Memory Hierarchy



Key Concepts

- Temporal Locality
- Spatial Locality
- Miss and Hit Rate
- Capacity, Sets, Blocks, Associativity
- Direct Mapped Cache
- Instruction vs. Data Cache
- N-way Set Associative Cache
- Replacement Policy
- Write Policy
- Virtual Memory
- Page Table

David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

Part II
Study
Advice

Part III
Key Concepts and
Previous Exam Questions



- (a) **English:** Assume that we have a 2-way set associative cache for a processor that uses 32-bits for addressing. The cache has 1024 cache blocks in total and the block size is 16 bytes. How many bits are then the tag field, the set field (also called the index), and the byte offset field, and how many validity bits does the cache contain in total? (4 points)

Answer: tag field = 19 bits , set field = 9 bits, byte offset field = 4 bits

David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

Part II
Study
Advice

 **Part III**
Key Concepts and
Previous Exam Questions



Module 6

Parallel Processors and Programs



Key Concepts

- Moore's law
- Power Wall
- Parallelism vs. Concurrency
- Speedup
- Amdahl's law
- Data-level vs. task-level parallelism
- SISD, SIMD, MIMD
- Instruction-Level Parallelism (ILP)
- VLIW vs. Superscalar (basic idea)
- Hardware Multithreading
- Shared Memory Multiprocessor
- Cache Coherence and False Sharing
- Processors, Threads, and Multicore
- Semaphores
- Clusters and Message Passing

David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

Part II
Study
Advice

 **Part III**
Key Concepts and
Previous Exam Questions

Project and Awards!



David Broman
dbro@kth.se

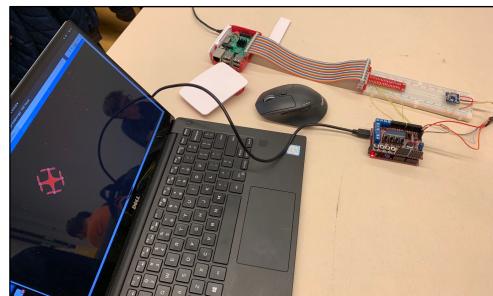
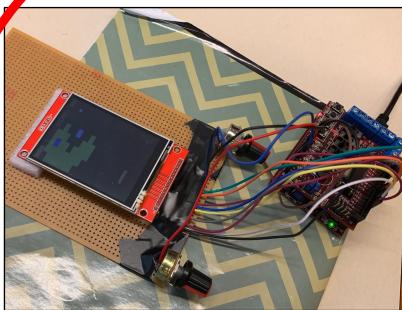
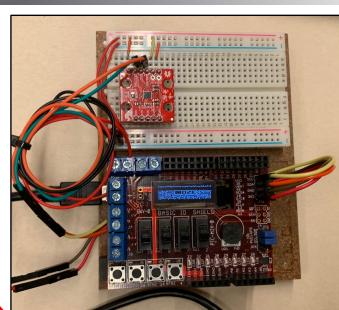
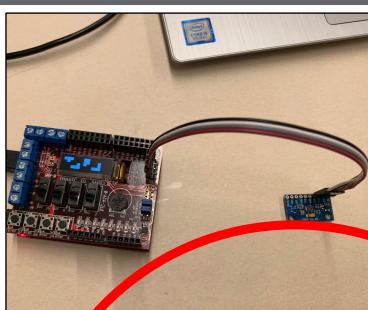
Part I
Exam Structure
and Grading

Part II
Study
Advice

 **Part III**
Key Concepts and
Previous Exam Questions



Category “Greatest Hardware Project Award”



David Broman
dbro@kth.se

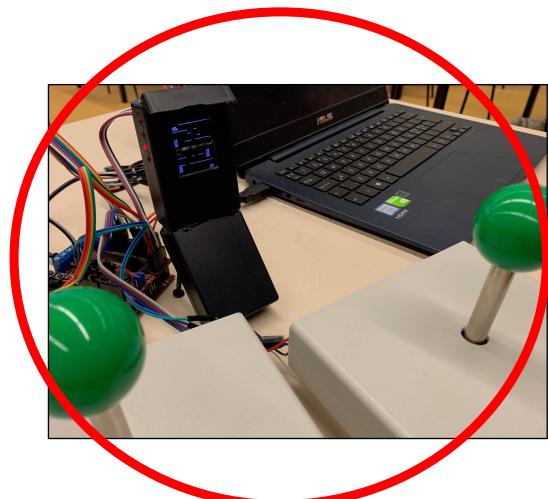
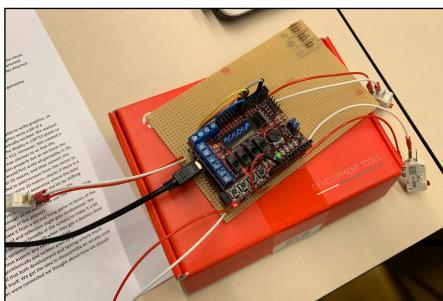
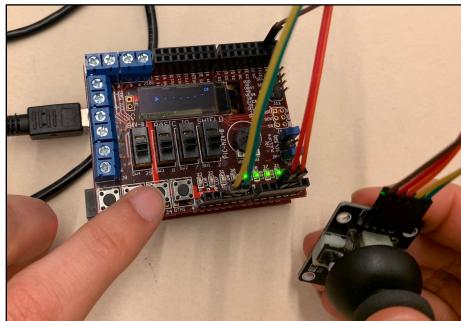
Part I
Exam Structure
and Grading

Part II
Study
Advice

 **Part III**
Key Concepts and
Previous Exam Questions



Category “Most Impressive Game Award”



David Broman
dbro@kth.se

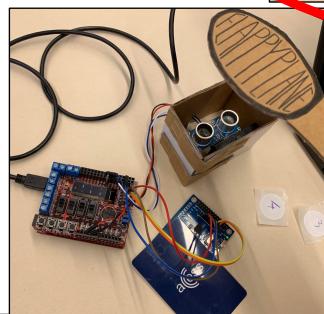
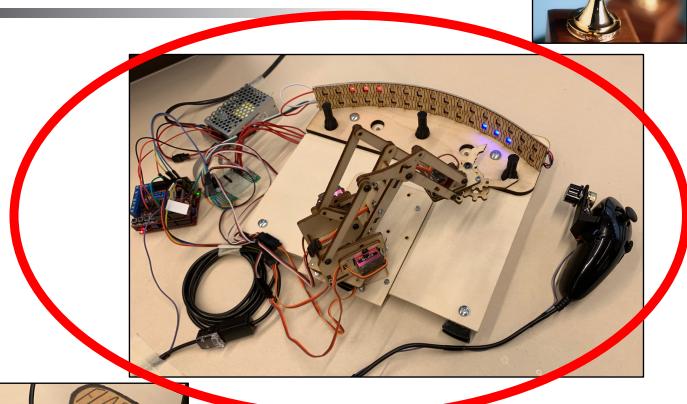
Part I
Exam Structure
and Grading

Part II
Study
Advice

Part III
Key Concepts and
Previous Exam Questions



Category “Coolest Project Award”



David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

Part II
Study
Advice

Part III
Key Concepts and
Previous Exam Questions



Summary

Some key take away points:

- **Computers** are fun!
- Please make sure to **study hard** for the exam.



Thank you and Good Luck!

David Broman
dbro@kth.se

Part I
Exam Structure
and Grading

Part II
Study
Advice

Part III
Key Concepts and
Previous Exam Questions