



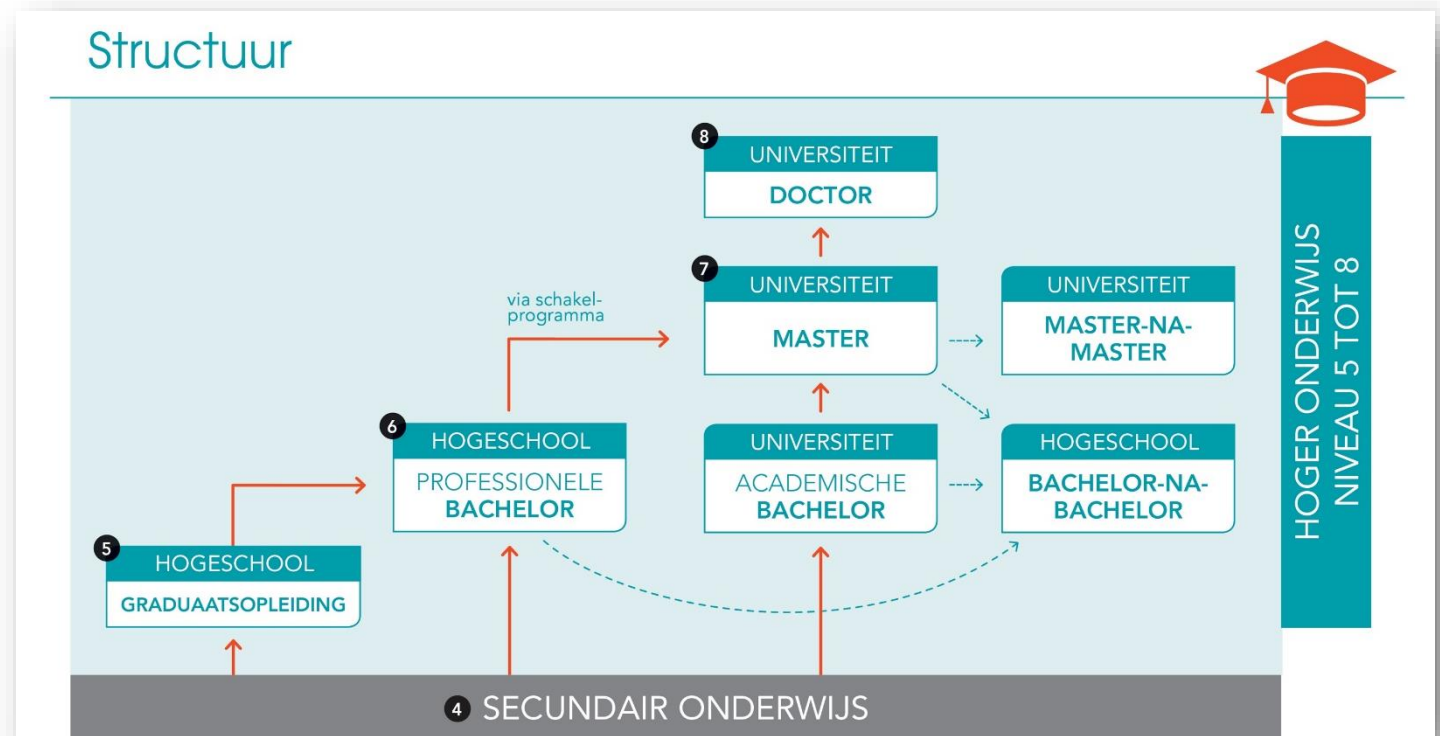
The Xplained board: Getting two-year program students on track in embedded systems

Maarten Van Lint

- Introduction
- Considerations target audience
- Choices in the past
- Xplained Mini

- Associate degree

- 2 year program (level 5)
- Organised by Universities Of Applied Science since 2019
- Previously by so called Adult Education



- Program
 - Main focus on
 - Embedded systems
 - Home Automation
 - Graduating students should have a good insight in microcontrollers
 - Only 2 years (from which 1 semester internship)!!!
- Audience
 - Students
 - With interest in IoT
 - Potentially without any prior knowledge of electronics(25% knows ohm's law)
 - No prior knowledge of C language

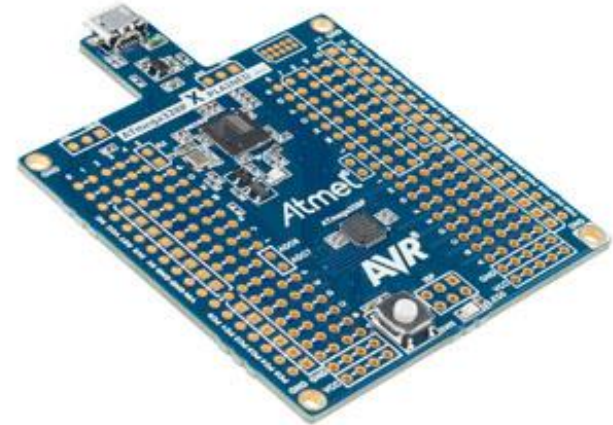
- Device to chose
 - The student should learn about the basic concepts of microcontrollers
 - Using the datasheet and other reference documents
 - Registers
 - Interrupts
 - Pointers
 - The microcontroller shouldn't have an overkill on peripherals and added hardware

- Device to chose
 - The student should learn about the basic concepts of microcontrollers
 - Using the datasheet and other reference documents
 - Registers
 - Interrupts
 - Pointers
 - The microcontroller shouldn't have an overkill on peripherals and added hardware

- MSP 430 (Texas instruments)
 - Simple development board from TI
 - Free development environment Code Composer from TI
 - Interesting option
 - Nowadays TI tends more to API's
 - » Insight in registers missing
 - » One still can write code without API's
- CC3200 (Texas Instruments)
 - Powerfull and interesting IoT device
 - Wifi on Chip
 - ARM cortex-M4
 - Mainly use of API's
 - » Insight in registers missing

- Xplained Mini board

- Atmega328PB microcontroller
- Free development environment
 - ~~Atmel Studio~~
 - Microchip Studio
- Footprint is Arduino compatible
- If required one wants, one could use the Arduino framework
- Very cheap



- Features Peripherals

- Well documented
 - Remark: In the examples Microchip uses a << command of 1 to manipulate a bit in a register (eg `1 << PINB5` which is 0010 0000)
- Debugging via a dedicated port
- Only basic peripherals are implemented
 - Digital IO
 - AD converter
 - USART
 - SPI
 - I²C
 - Timers & PWM

- Disadvantages of Arduino
 - Debugging via serial interface
 - Sketch hasn't much debugging options
 - Printf takes a huge amount of time
 - Use of API's
 - The Arduino framework adds a lot of unnecessary code

- **Until Academic year 2022-2023**
 - Semester 01 (1 course: C, use of datasheet, basics, IO, USART)
 - Very hard
 - Slow progress
 - Reason: students had to learn at the same time:
 - » analyse a problem and provide a logic to solve it
 - » C language
 - » Working with microcontrollers and registers
 - Semester 02 (1 course: AD, Interrupts, SPI, I²C, timers and PWM)
 - Mainly very good
 - Students use their board in projects and implement interrupt driver code

- **From Academic year 2022-2023**
 - Semester 01
 - 2 courses (parallel):
 - » C language
 - » Use of datasheet, basics, IO, USART
 - Semester 02
 - 1 Course: AD, Interrupts, SPI, I²C, timers and PWM)

