WORKSHOP

PHYSICAL COMPUTING with Raspberry Pi PICO

date: 14/10 hrs: 9:30 - 17:00
location: Seminarylokaal 5
The Raspberry Pi PICO is an electronics platform based on easy-to-use hardware and software. It's intended for anyone making interactive projects. We look at exisiting projects and we start programming the Pico itself. Hello world, led blink!
---
https://www.raspberrypi.org/products/raspberry-pi-pico/
With Wendy Van Wynsberghe & Hendrik Leper

SCHEDULE:
INTRO 1hour

- Intro Open hardware
- Microcontroller & washing machine exercise
- Raspberry Pi
    - PICO (vs Raspberry PI vs Arduino)
    - python vs micropython vs C++
- Physical Computing projects

GETTING READY 1-2hours
Dirty the hands 3-5hours

1. Intro Open Hardware

(olimex, arduino, makeymakey, theremin, ...)
computers: Olimex versus raspberry Pi
https://www.olimex.com/Products/OLinuXino/open-source-hardware
https://freedomdefined.org/OSHW
Really Open hardware:
https://github.com/OLIMEX/OLINUXINO/tree/master/HARDWARE/A20-OLinuXino-MICRO/2.%20Older%20hardware%20revisions/Hardware%20revision%20G1
WVW: show Gerber files Olimex

WVW: So we know what computers are - and about Open Hardware -->

2. What is a microcontroller?

A **single-board microcontroller** is a microcontroller (chip) built onto a single printed circuit board. This board provides all of the circuitry necessary for a useful

control task: a microprocessor, I/O circuits, a clock generator, RAM, stored program memory and any necessary support ICs. The intention is that the board is immediately useful to an application developer, without requiring them to spend time and effort to develop controller hardware.

What is a microcontroller > **the washing machine paradigm** (exercise)
What are the steps that the washing machines' microcontroller takes when washing your dirty laundry.
what are the sensors and what are the actuators that the microcontroller needs to feel and control the washing procedure?

(sensor on the doorlid, temperature sensor, on and off, measure temperature of water, type of cyle -- motors to turn, heat up water, pump wauter, empty water, spin drum)

3. RASPBERRY PI

WVW: Mini history
- first launched in 2012 - developed in the UK by the Raspberry Pi Foundation.
- originally leaned towards the promotion **of teaching basic computer science in schools and in developing countries**
- widely used in many areas, such as for weather monitoring, robotics, because of its low cost, modularity, and open design.
https://en.wikipedia.org/wiki/Raspberry_Pi
Walk door website https://www.raspberrypi.org/

3.1 RBP PICO vs the other RBP family (as ZERO, 1,2,3,4 model A & B, 400)
a **microcontroller** vs a **microcomputer**
Pico is a great device for dedicating to a very specific task (it runs only one program at a time)
whereas **the other Raspberry Pi**'s are a multipurpose devices. A *regular* Raspberry Pi is a Linux computer that boots up into a full operating system, it has HDMI out, a camera interface, etc.

3.2 RBP PICO vs Arduino
https://all3dp.com/2/raspberry-pi-pico-vs-arduino/
- Processors

- 1 core, 8bit, 16MHz vs
- 2 core, 32bit 133MHZ

- input / output

- 5V, 6 analog input pins, 14 GPIO of which 6 provide PWM output vs
- 3.3V, with 26 GPIO pins, 3 of which can be used as analog inputs and 16 as PWM outputs

- ADC higher resolution (10-bit vs. 12-bit) and a faster sampling rate (125 kHz vs 500 kHz)

- Programming

- C++ sketches, which are processed, compiled, and loaded onto the board as machine code with IDE
- MicroPython = a version of the popular Python language for microcontrollers. easier & faster

- Price

- 23€
- 4.5€

4. Physical Computing projects

Two Arduino garden projects - very practical
WVW: https://create.arduino.cc/projecthub/londonium2021/automatic-indoor-vegetable-garden-9ea453?ref=tag&ref_id=plants&offset=6
WVW: https://www.researchgate.net/publication/330212779_An_Automated_Irrigation_System_Using_Arduino_Microcontroller
WVW: https://www.kobakant.at/KOBA/trombone-breathing/

HL
eigen werk voor studenten & gasten KASK
-

- - Leon de bruijne http://leondebruijne.nl/remora/
- - Lana Schneider http://lanaschneider.be/2012.html

5. Getting ready

duration 1h synchronous (all together)

- 1. Introduction
- 2. the Raspberry Pi Pico Board
- 3. the MicroPython Firmware
- 4. the Software
  - Bring Thonny in
  - a Walk through the Thonny UI
  - Linking Thonny to Pico

- 5. What the Shell! Conversing with (Micro)Python
- 6. Over to Script Mode
    - Switch that LED ON & OFF
    - ON/OFF in Loop
- 7. Let's Get Physical
    - Your Pico’s Pins
    - Common Components
    - Reading Resistor Colour Codes
- 8. Wiring Diagrams & Schematics

6. Dirty the hands

duration 3 to 5h asynchronous

- 9. Next Level LED Blinking.
- 10. a Pushbutton Digital Inputs
    - a Pushbutton
    - One Circuit Multiple Behaviours
    - a Pushbutton with Interrupt (optional)
    - Other On/Off Sensors
- 11. Sensors Analog Inputs
    - Let's Read the Value of a Potentiometer
    - Controlling the Speed of our Blinking LED with a Potentiometer.
    - Other analog sensors
- 12. PWM Analog Outputs
    - Fading an LED with the Potentiometer & PWM
    - Fading an LED IN & OUT with PWM
    - Other PWM-controlled Actuators
- 13. Data logger
    - File storage
    - Running without a Host Computer

https://projects.raspberrypi.org/en/projects/getting-started-with-the-pico/
https://github.com/theBlackBoxSociety/CodeCrashCourses/blob/master/RaspberryPiPicoTutorial.md

Friday
>> experiment with sensors & actuators etc.
>> see also https://itp.nyu.edu/physcomp/