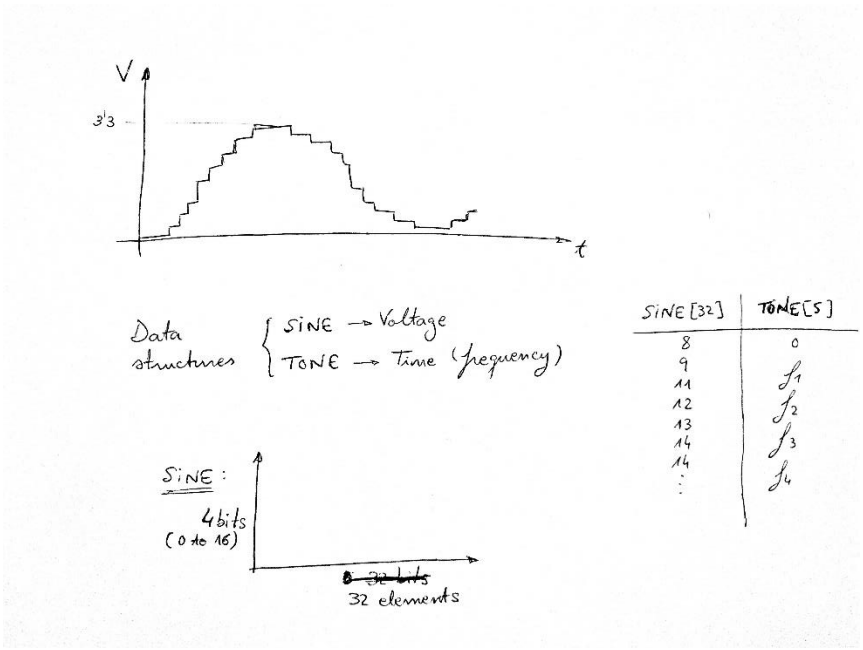
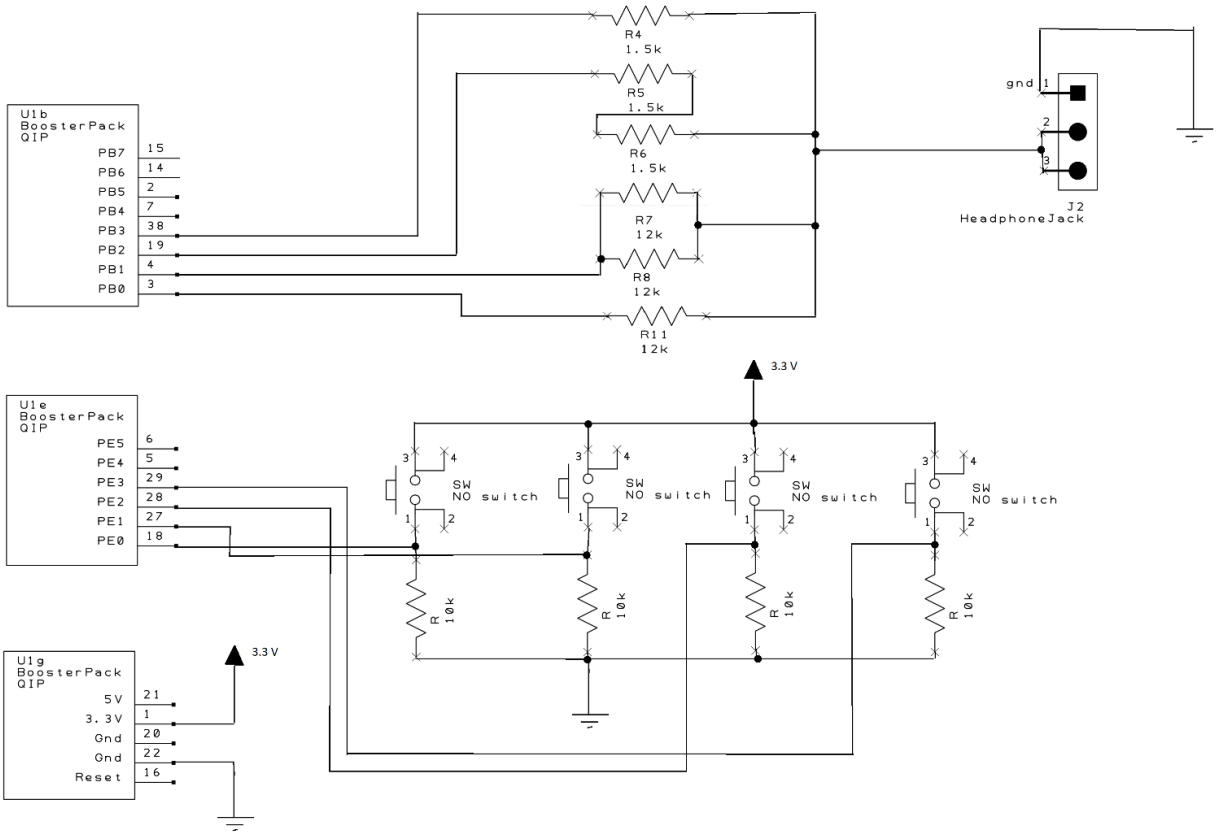
Lab 6 – Deliverables

Data structures:



Circuit diagram:



Bit3 bit2 bit1 bit0	Theoretical DAC voltage	Measured DAC voltage
0	0	0.0003
1	0.22	0.229
2	0.44	0.455
3	0.66	0.684
4	0.88	0.894
5	1.1	1.123
6	1.32	1.349
7	1.54	1.578
8	1.76	1.726
9	1.98	1.955
10	2.2	2.178
11	2.42	2.405
12	2.64	2.612
13	2.86	2.835
14	3.08	3.062
15	3.3	3.291

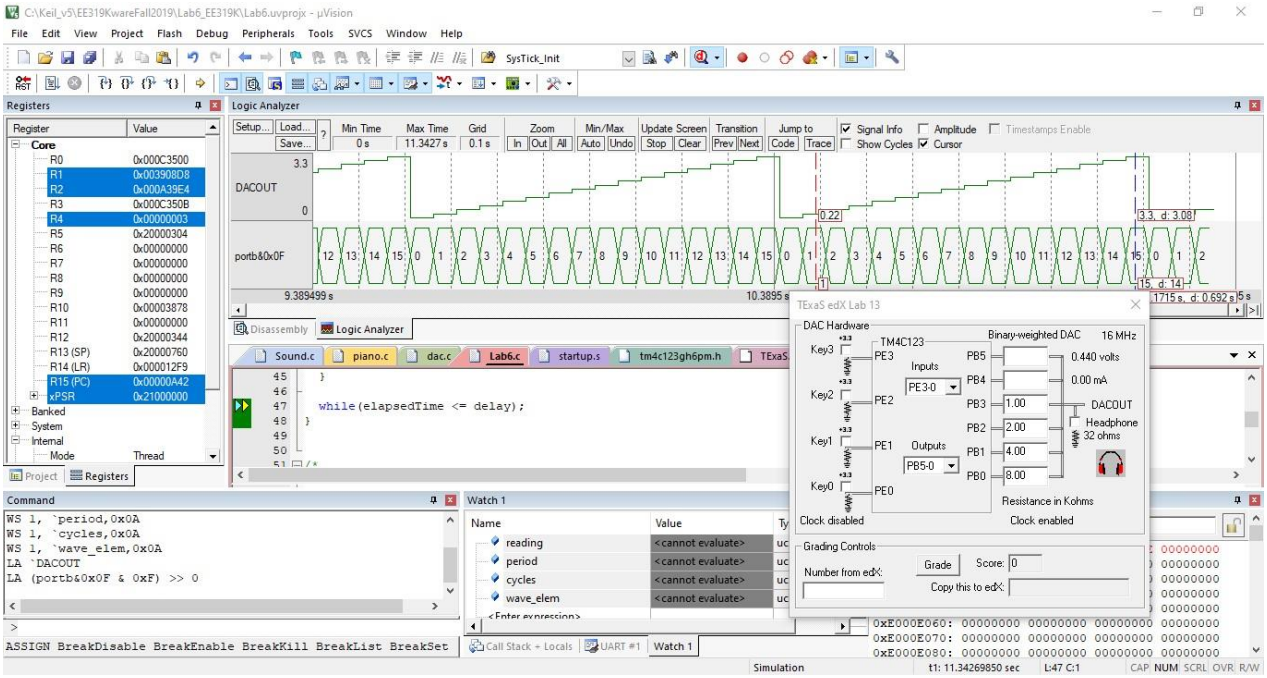
Resolution: 0.22 V

Range: 3.3 V

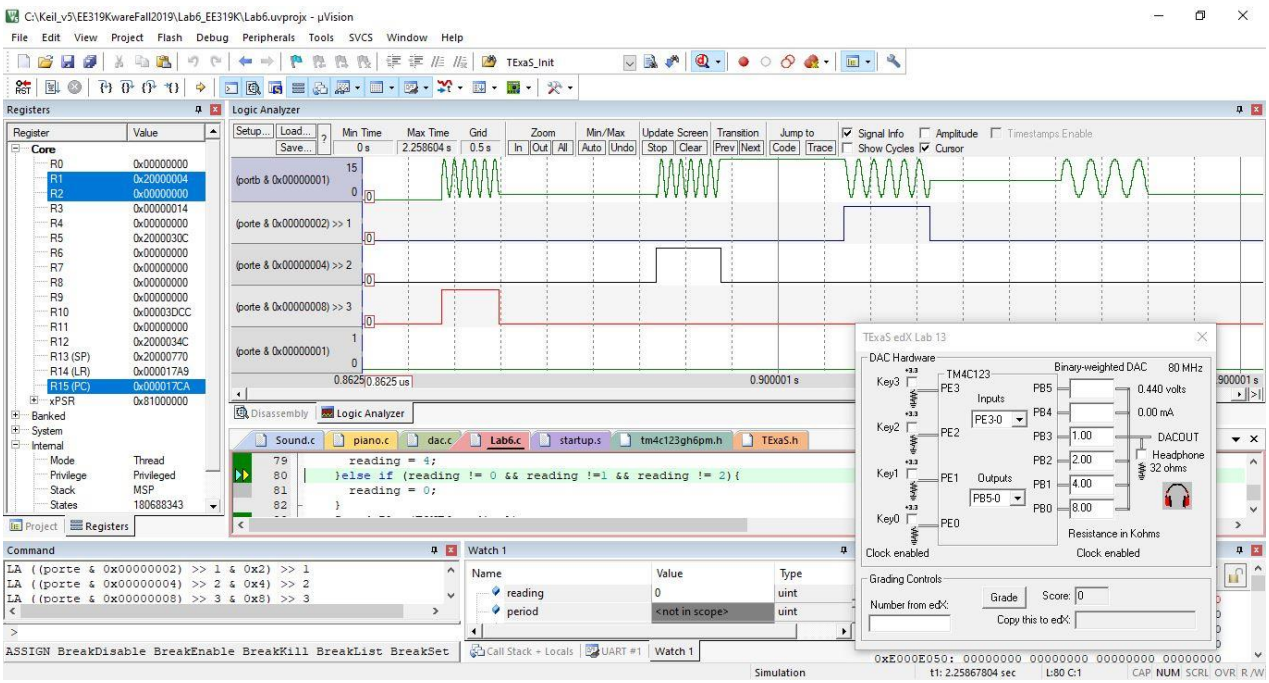
Precision: 16 alternatives (4 bits)

Accuracy: ± 0.11 V

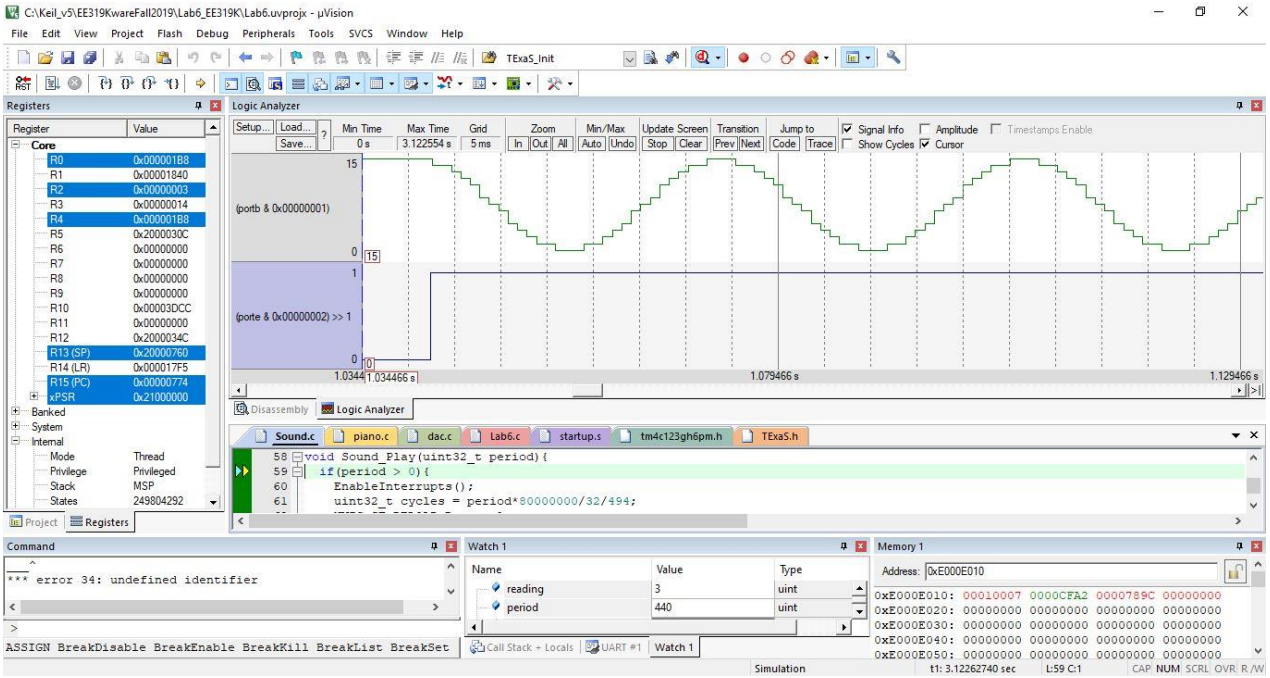
Simulation of the DAC:



Simulation of the whole program:



Detail of the wave:



When does the interrupt trigger occur?

When the CURRENT value of the timer changes from 1 to 0.

In which file is the interrupt vector?

In Startup.s

List the steps that occur after the trigger occurs and before the processor executes the handler.

- The current function is finished
- Eight registers are pushed into the Stack with R0 on top
- The vector address is loaded into the PC
- The IPSR register is set to the interrupt number (15 for SysTick)
- The top 24 bits of LR are set to 0xFFFFFF. The bottom eight bits specify how to return from the interrupt.

It looks like BX LR instruction simply moves LR into PC, how does this return from interrupt?

The program knows that it is returning from an interrupt because the top 24 bits of LR are set to 0xFFFFFF. It follows these steps:

- Pop the eight registers off the stack
- Return to thread mode using MSP (if the bottom eight bits of the LR are 0xF9)
- The IPSR is automatically reset to its previous state.