Embedded Programming Hackaton 2023-2024

v1.1 Jan 18, 2023

Case: Advanced Life Support Patient Monitor Simulation together with UtrechtUMC Wilhelmina Children's Hospital

- **Click here to enroll**

Introduction

Wilhelmina Children's Hospital (WKZ) in Utrecht (part of UtrechtUMC) is always looking for new ways to improve skills of the nurses and physicians. Together with the HAN Health Concept Lab they are developing tools for life support training including a manikin to practice high quality CardioPulmonary Resuscitation (CPR) .

Case

Wilhelmina Children's Hospital wants to improve their training by adding vital signs monitoring. In the past we already developed the box to trick real patient monitors using Arduino that gives an idea of the end goal of the hackaton (see Fig. 1).

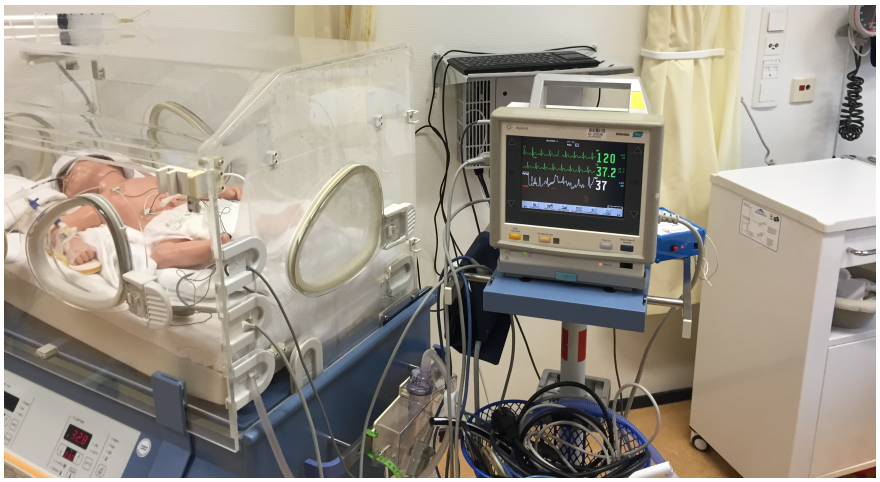

Figure 1: The end goal: a simulated vital signs monitor for training advanced life support.

Figure 2 shows a simulated vital signs monitor. The monitor shown is already available for life support training. WKZ wants to be able to use such a monitor with a manikin to increase realism. This means real vital signs sensors will need to be connected. When a particular sensor (e.g. Electrocardiogram (ECG)) is

connected, the simulated vital signs should show on the monitor (in case of the ECG , heart rythm, heart beat, i.e. the green signal in from Fig. 2).

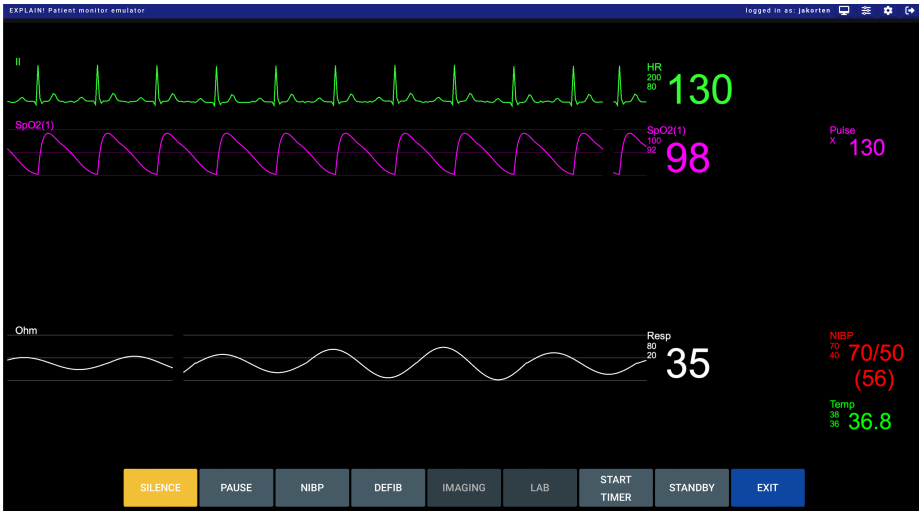

Figure 2: Example of a vital signs monitor. We see the version of RadboudUMC which is available to use as reference during the hackaton.

Requirements

Based on the same principle WKZ wants to: - Only show particular vital signs when the appropriate (simulated) vital sign leads are attached. - Run the simulator on a Raspberry Pi, also when there is no external network connection is available (hotspot mode).

The Challenge

Your assignment is to program (parts) of this simulator. You will need to combine both high and in low-level C/C++ code. At the Raspberry Pi you will need to use C++ and the Object Oriented Paradigm (optimized for code quality) possibly together lower level code (optimized for speed to generate the vital signs). On the simulation modules (see Fig. **??**) you will need to use low-level C-code as well.

The following vital signs

- ECG : difficult (Must)
- Blood oxygenation (SpO2): difficult (Should)
- Non-invasive blood-pressure (NIBP) (sensor + actuators): moderate (Should)
- Temperature (cavity and surface/skin): easy (Should)

- Breathing (via the other signals): difficult (Could)

Note: the ECG is mandatory and therefor has MoSCoW priority Must, the other items are Should. Real patient monitors determine breathing via the other signals. Breathing patterns are a Could. Code quality and vital signs quantity all add up to the final result.

Hardware and Software

In order to be able to support this project also after the hackaton, we decided to prescribe the hardware and parts of the software. The Advanced Life Support simulation modules use SAMD21/SAMD51 microcontrollers. The modules are in essence all i2c peripherals. The next section lists the contents of the simulation kit.

There is already a basic Hardware Abstraction Layer (HAL) that you should use: Universal HAL.

Simulation Kit content

There are modules available for all the vital signs described: they each have their characteristics and (estimated) difficulty level.

ALS Simulation Kit content

Fig. 3 shows the simulation kit with the modules listed above.

Figure 3: The simulation modules as interface to connect the vital sign sensors to the simulated monitor.

Fig. 4 shows the SensorHub Module.
The entire kit contents is listed below:

- 2x SensorHub Allows to connect multiple sensors and uses a SAMD21

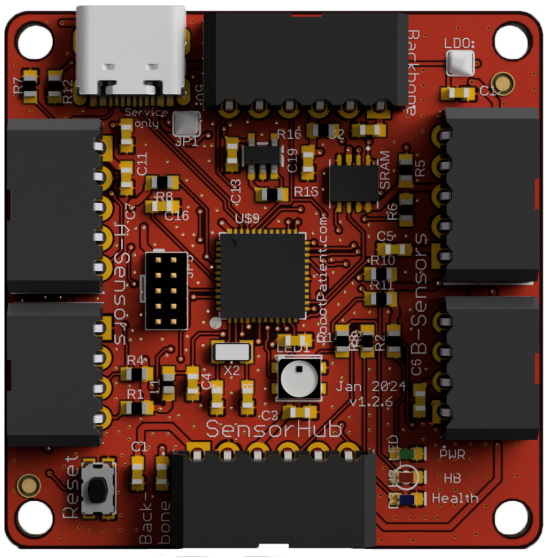

Figure 4: The SensorHub module acts as a communication hub between the sensors and the simulated monitor.

microcontroller: has a USB-C interface to program.

- 1x SimModule ECG + 3 ECG leads and Lead detector board.
- 1x SimModule SpO2 + blood oxygen saturation sensor.
- 2x SimModule Temp + 1x surface + 1x cavity probe.
- 1x SimModule NIBP + neonatal cuff.
- 1x SimModule Single Row adapter (i2c multiplexer)
- 1x SimModule Adapter (i2c multiplexer)
- 1x SimModule PiInterface with SAMD51 (communication with the Raspberry Pi)
- 1x Raspberry Pi 3/4 + Power Supply + SD-card
- 1x HDMI-screen

Practical Information

Schedule

Date: Wednesday Jan, 24 2024 - 8.30 - 21.00 R29/H2.12

8.30	Arrival of the hackers. Let's fill our cup with Coffee/Tea
8.45	Introduction on the subject
9.00 - 12.30	Programming
2.30 - 13.15	Lunch break / Walk in the park (optionally)
13.15 - 18.00	More programmming
18.00 - 18.30	Pizza break
18.30 - 20.30	More programmming
20.30 - 21.00	Drinks

Thursday Jan, 25 2024 - 8.30 - 16.30

R29/H2.12:	
8.30	Arrival of the hackers. Let's fill our cup with Coffee/Tea
8.45	Stand-up meeting
9.00 - 12.30	Programming
12.30 - 13.15	Lunch break / Walk in another park (optionally)
R29/H0.36:	
13.15 - 16.30	Finalizing the programming + assessments
16.30	Drinks, snacks and Award ceremony

Table 1: (Preliminary) schedule.

Note: Please bring your own lunch and cups. Coffee/tea/hot water will be provided.

Awards

Apart from the grading there will be rewards:

1. First place: Raspberry Pi 5 kit (4Gb RAM, PSU, housing, etc) for the best group (limited availability so 1x Pi to raffle within the group, JLink debuggers that retail at 60 bucks or more for each member).
2. Second place: Feather M4 boards
3. Third place: Feather M0 boards
4. For all participants: Debug ducks!

And of course you get credits for Prog 6 (if you pass, see Rubrics).

Please use the Open-Closed principle for this! - The Raspberry Pi software can be higher level C++, preferably object oriented. You should generate vital signs (e.g. an electrocardiogram) to be displayed on a screen via one of the HDMI-ports. The characteristics should be configurable. For the communication between possible control devices and the Pi, MQTT is preferable.

Reference materials

Note: more reference material will be added before the hackaton.

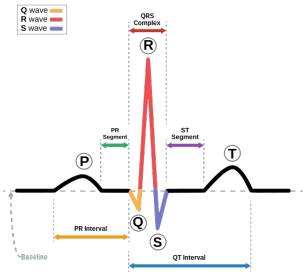

Figure 5: Electrocardiogram.

Fig. 5 shows the basic morphology of an ECG : reference.

The sinus rythm is the standard rythm without anomalies. There are also many anomalies (arrythmia). These could be simulated as well. You can see this in the simulated cloud-based monitor Explain Monitor.