The Chinese University of Hong Kong
Department of Computer Science and Engineering
CENG2030 Fundamentals of Embedded System Design

Project 1: Pulse Detector

Submission Instructions:

- You are required to submit **demo videos**, and **Arduino codes** to Blackboard.
- Create each Arduino project with a project name based on the project and question numbers, e.g. "**ceng2030_pro1**".
- Compress all the files to one single zip/rar file named with your student ID number, e.g. "**1155123456.zip**".
- **Upload the zip/rar file before the deadline as stated in Blackboard**, **10 marks** will be deducted **per hour**

For each question below, you are required to record a short mp4 **video** to demonstrate the answers. In the video, the following elements are required:

A. Show your full name and SID on a paper next to your circuit [5 marks]
B. Voice descriptions in English/Cantonese/Mandarin on what you are doing [5 marks]
C. Demonstrate your works by presenting your system and its outputs including LED and raw data [10 marks]

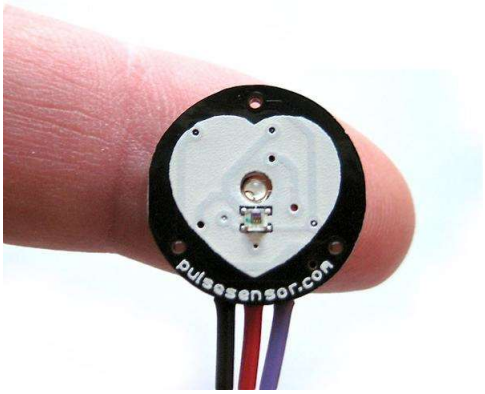
List of components and equipment

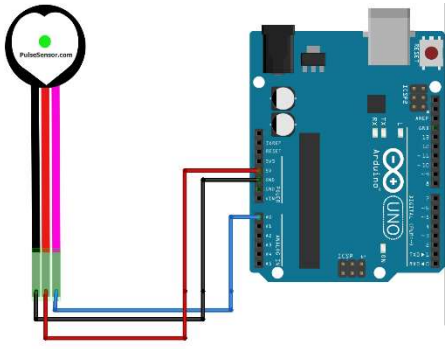
- Arduino UNO board
- Micro SD card module
- IR pulse sensor
- Resistor: 1x 100kΩ
- LED: 1x LED
- Breadboard

1. Pulse Detector

In this project, a pulse detector will be built to detect the pulse rate of human being. An IR pulse sensor can be attached on the finger to detector the pulse signal. The pulse signal will be processed by the Arduino processor to determine the actual value of the pulse rate. The pulse rate data will be stored in the Micro SD card module. For visualization of the detected pulses, a resister and a LED can be used.

The pulse sensor and the connection diagram with Arduino UNO board are shown below. As shown at the back of the sensor, the pins are labelled as "**S**", "**+**", and "**–**" for Signal, 5V, and Ground respectively. The sensor output signal is an analog output, so it is connected to pin A0 on the Arduino UNO board for analog signal input.



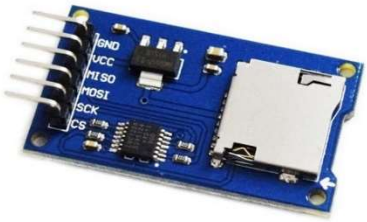


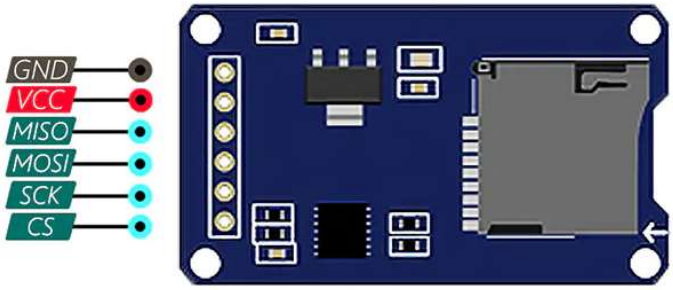
2. Pulse Indicator

To visualize the detected pulse signal, connect the resistor and the LED in series from one of the digital output pins to the ground. Refers to the previous lab sheets for details.

3. Micro SD Card Module [Optional]

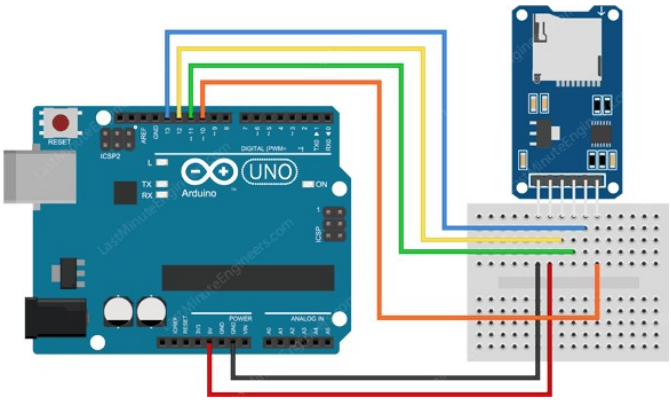
The Micro SD card module (optional) is shown below. It can be used to store the data onto the micro SD card. The pins GND, VCC, MISO, MOSI, SCK, and CS stand for Ground, 5V, Master In Slave Out, Master Out Slave In, Serial Clock, and Chip Select respectively.



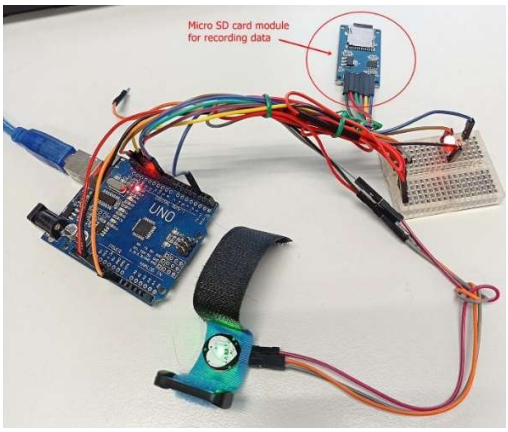


- GND is Ground (0V)
- VCC is Power (5V)
- MISO is SPI output from the SD card.
- MOSI is SPI input to the SD card.
- SCK pin accepts clock pulses which synchronize data transmission generated by Arduino.
- CS pin is used by Arduino (Master) to enable and disable specific devices on SPI bus.

The connection between Arduino and Micro SD Card Module is shown below.



The resistor and LED can be connected **in series** to a digital output pin of the Arduino to show the detected pulses. After connecting the Arduino UNO board, pulse sensor, micro SD card module, resistor and LED together, the circuit will be similar to the following picture.



4. Data Acquisition and Pulse Rate Estimation

After the hardware connections above, it is ready for data acquisition and pulse rate estimation. In your Arduino program, implement the following tasks with the hints below:

a. Set a reference timer for pulse counting [10 marks]

To get the time information, use millis() which returns the number of milliseconds passed since the Arduino board began running the current program.
millis(): https://www.arduino.cc/reference/en/language/functions/time/millis/

b. Input analog raw data from IR sensor [20 marks]

To read input from analog input pin, use analogRead() which reads the value from the specified analog pin.
analogRead(): https://www.arduino.cc/reference/en/language/functions/analog-io/analogread/

[Optional] You may plot the analog raw data by printing the data using Serial.begin() and Serial.print(), and plotting the data as a graph continuously using Serial Plotter for checking.
Serial.begin(): https://www.arduino.cc/reference/en/language/functions/communication/serial/begin/
Serial.print(): https://www.arduino.cc/reference/en/language/functions/communication/serial/print/
Serial Plotter: https://arduinogetstarted.com/tutorials/arduino-serial-plotter

c. Set a threshold for the analog raw data in order to indicate a detected pulse [10 marks]

For both Arduino Uno and Nano, the maximum resolution of analog input is 10 bits. This means that the analog input is in the range of 0 to 1023 (i.e. 2^{10}-1). You should set a threshold to determine the detected pulse yourself for correct pulse detection. If the analog raw data is larger than the threshold, pulse is detected, and light up the LED by digital output using digitalWrite().
digitalWrite(): https://www.arduino.cc/reference/en/language/functions/digital-io/digitalwrite/

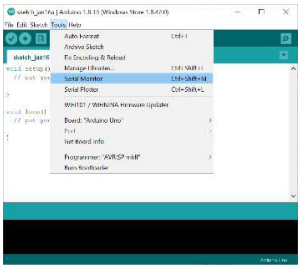
d. Count the number of pulses per minute (i.e. estimated pulse rate) [20 marks]

By counting the detected pulse in one minute, you should be able to estimate the pulse rate. Please note that the estimated pulse rate may not be correct initially since not enough data is processed. However, it will be steady when the pulse is being counted for over one minute.

e. Show the analog raw data and the estimated pulse rate at the Serial Monitor window [20 marks]

To output the raw data and the pulse rate, use Serial.begin() and Serial.println() which prints data to the serial port as human-readable ASCII text followed by a carriage return character (ASCII 13, or '\r') and a newline character (ASCII 10, or '\n').
Serial.println(): https://www.arduino.cc/reference/en/language/functions/communication/serial/println/

To start up the Serial Monitor, from the menu of Arduino IDE, click Tools ➔ Serial Monitor. The printed texts by using Serial.println() will be shown in the Serial Monitor window.
Serial Monitor: https://arduinogetstarted.com/tutorials/arduino-serial-monitor



f. [Optional] If you have a micro SD card module, you may use it with the following reference. [0 marks]
https://randomnerdtutorials.com/guide-to-sd-card-module-with-arduino/

THE END