5/15/2022

Logo, company name

Description automatically generated

Shahin Haque Previously known as Muhammed Azizul Haque https://www.linkedin.com/in/shahinhaque97/

Plymouth uNiversity

Health Logging Device

Module Code: PROJ324

Contents

[Introduction 2](#_Toc101452581)

[Sensors 3](#_Toc101452582)

[ADS1292R - ECG 3](#_Toc101452583)

[Setting up the ADS1292R 3](#_Toc101452584)

[What is an ECG and how do they work? 6](#_Toc101452585)

[Labelled ECG waveform 7](#_Toc101452586)

[ECG waveform formatting 9](#_Toc101452587)

[Electrical principles of an ECG 10](#_Toc101452588)

[AFE4490 – Pulse Oximeter 11](#_Toc101452589)

[What is Oxygen Saturation? 11](#_Toc101452590)

[Setting up the AFE4490 13](#_Toc101452591)

[MAX30205 – Body Temperature Monitor 14](#_Toc101452592)

[What classifies as a Fever (CDC)? 14](#_Toc101452593)

[What classifies as Hypothermia (CDC)? 15](#_Toc101452594)

[Interfacing the MAX30205 16](#_Toc101452595)

[SD Card 17](#_Toc101452596)

[Cloud Services 18](#_Toc101452597)

[LoRa 19](#_Toc101452598)

[What is LoRa? 19](#_Toc101452599)

[Where is LoRa used? 20](#_Toc101452600)

[Interfacing the SX1278 LoRa Module to the Microcontroller 21](#_Toc101452601)

[SX1278 Library/Code example 21](#_Toc101452602)

[Power Management 23](#_Toc101452603)

[PCB Design 24](#_Toc101452604)

[Bill of Material (BoM) 28](#_Toc101452605)

[Appendices 29](#_Toc101452606)

[Heart Sensor Code 33](#_Toc101452607)

[Lungs Sensor Code 38](#_Toc101452608)

[Body Temperature Sensor Code 42](#_Toc101452609)

# Introduction

# Sensors

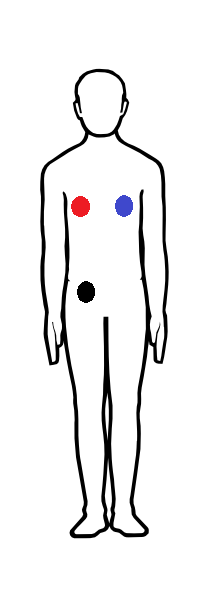
The focus of this project is to measure values from the heart, lungs, and body temperature. Each sensor microchip will be connected to a controller to set up the chip via SPI / I2C, followed by a USART communication to the logger microcontroller

Sensor chip

Sensor controller

Logger Microcontroller

LoRa to PC



## ADS1292R - ECG

The ADS1292R is a two channel 24-bit ADC with integrated respiration impedance and an ECG front end, having three electrodes connected to the chest. Two ECG electrodes and one DRL (Driven Right Leg) electrode for common mode noise reduction. Using impedance pneumography to measure respiration, this method works by measuring the change in chest impedance to calculate the respiration measurement.

For the prototyping stage, the ADS1292R breakout board is used. This board is provided by PROTOCENTRAL connected to an Arduino for testing. Upon completion a custom board will be created.

### Setting up the ADS1292R

The chip will be connected to a controller using SPI. Using the four main SPI connections (SCLK, MISO, MOSI, CS) as well as DRDY pin to assert when the data conversion is complete.

For a single device, the minimum speed needed for SCLK depends on the number of channels, number of bit resolutions and the output data rate. This can be calculated below:

A picture containing text

Description automatically generated

Let say, the mode we have set this chip to is 500SPS (samples per second), using two channels with a resolution of 24-bits, the minimum SCLK speed is approximately 36kHz.

Data retrieval can be done in two ways. Data can we read continuously by setting the chip in continuous mode upon start up. Alternatively, an opcode can be sent to read just one data output reading. The number of bits in the data output depends on the number of channels and the number of bits per channel. For the chip I will be using, there is twenty-four bits for status, twenty-four bits for ECG and twenty-four bits for the oxygen saturation. For the twenty-four bits of the status the signal is sent as follows:

1100 + LOFF\_STAT [4:0] + GPIO [1:0] + 13 ‘0’s

The table of commands for this chip is shown below:

Table

Description automatically generated with medium confidence

The chip breakout board that is connected as usual with the SPI pins such as MISO (Master in Slave Out), MOSI (Master Out Slave In), CS (Chip Select), SCK (Serial Clock) and Power (VDD & GND). There are a few extra pins such as reset, start and data ready. Data ready can be connected to an interrupt pin for thread like computation or put the microcontroller to sleep until the data is ready to save power. This is a two-channel device, however there is GPIO pins to attach a secondary ADS1292R to cascade them into a 4-channel device. The ADS1292R is shown in **figure X**.

Graphical user interface

Description automatically generated with medium confidenceDiagram, schematic

Description automatically generated

A picture containing electronics

Description automatically generatedThe upgraded version of this breakout board Is the ADS1293, shown in **figure X**. This has three high-resolution digital ECG channels with a simultaneous pace output. As each sensor is connected via its own dedicated microcontroller as states earlier for the reason of being swappable for newer chips, without affecting the other sensors on the board.

The ADS1292R chip does all the computations using impedance pneumography and outputs the data as states above. This data can be plotted as is, however if we would like to transmit the data. It will be split into three bytes per sample, which equates to twenty-four. How this is done is shown in the LoRa section of this report.

The break-out connections are shown in **Figure X**. This will be connected to the ATMEGA328-P microcontroller. This microcontroller will be dedicated to this sensor and its sole function will be to capture the data and send it to the host microcontroller to be analysed.

|  |  |  |
| --- | --- | --- |
| ads1292r pin label | Microcontroller Connection | Pin Function |
| VDD | +5V | Supply voltage |
| PWDN/RESET | D4 | Reset |
| START | D5 | Start Input |
| DRDY | D6 | Data Ready Output |
| CS | D7 | Chip Select |
| MOSI | D11 | Slave In |
| MISO | D12 | Slave Out |
| SCK | D13 | Serial Clock |
| GND | GND | GND |

The device is initiated the same way any SPI device is, setting the SPI clock speed (In this case 1MHz), followed by the SPI data mode (set to 1). Lastly, the bit order set to MSB first. A 24-bit data stream can be viewed on the Arduino plotter.

### What is an ECG and how do they work?

The NHS definition of an Electrocardiogram ECG is as follows:

**An electrocardiogram (ECG) is a simple test that can be used to check your heart's rhythm and electrical activity.**

Sensors attached to the skin are used to detect the electrical signals produced by your heart each time it beats.

These signals are recorded by a machine and are looked at by a doctor to see if they are unusual.

An ECG may be requested by a heart specialist (cardiologist) or any doctor who thinks you might have a problem with your heart, including your GP.

The test can be conducted by a specially trained healthcare professional at a hospital, a clinic or at your GP surgery.

An ECG is often used alongside other tests to help diagnose and monitor conditions affecting the heart.

It can be used to investigate symptoms of a possible heart problem, such as [chest pain](https://www.nhs.uk/conditions/chest-pain/), [palpitations](https://www.nhs.uk/conditions/heart-palpitations/) (suddenly noticeable heartbeats), [dizziness](https://www.nhs.uk/conditions/dizziness/) and [shortness of breath](https://www.nhs.uk/conditions/shortness-of-breath/).

An ECG can help detect:

* [**arrhythmias**](https://www.nhs.uk/conditions/arrhythmia/) – where the heart beats too slowly, too quickly, or irregularly
* [**coronary heart disease**](https://www.nhs.uk/conditions/coronary-heart-disease/) – where the heart's blood supply is blocked or interrupted by a build-up of fatty substances
* [**heart attacks**](https://www.nhs.uk/conditions/heart-attack/) – where the supply of blood to the heart is suddenly blocked
* [**cardiomyopathy**](https://www.nhs.uk/conditions/cardiomyopathy/) – where the heart walls become thickened or enlarged

A series of ECGs can also be taken over time to monitor a person already diagnosed with a heart condition or taking medication known to potentially affect the heart.

How an ECG is conducted

There are several separate ways an ECG can be conducted. The test involves attaching several small, sticky sensors called electrodes to your arms, legs, and chest. These are connected by wires to an ECG recording machine.

You do not need to do anything special to prepare for the test. You can eat and drink as normal beforehand.

Before the electrodes are attached, you will usually need to remove your upper clothing, and your chest may need to be shaved or cleaned. Once the electrodes are in place, you may be offered a hospital gown to cover yourself.

The test itself usually only lasts a few minutes, and you should be able to go home soon afterwards or return to the ward if you are already staying in hospital.

There are three main types of ECG:

* **a resting ECG** – conducted while you are lying down in a comfortable position
* **a stress or exercise ECG** – conducted while you are using an exercise bike or treadmill
* **an ambulatory ECG** **(sometimes called a Holter monitor)** – the electrodes are connected to a small portable machine worn at your waist so your heart can be monitored at home for 1 or more days

The type of ECG you have will depend on your symptoms and the heart problem suspected.

For example, an exercise ECG may be recommended if your symptoms are triggered by physical activity, whereas an ambulatory ECG may be more suitable if your symptoms are unpredictable and occur in random, short episodes.

As we can see from the types of ECGs my portable ECG is like the ambulatory ECG with additional sensors and data interpretation.

### Labelled ECG waveform

Diagram

Description automatically generated

|  |  |  |  |
| --- | --- | --- | --- |
| Breakdown for the ECG waveform | | | |
|  | The first and only wave is positive and thus an R wave |  | A large negative wave (Q), followed by a small positive wave (r) |
|  | The first wave is large and positive (R) followed by a small negative wave (s) |  | The negative wave manages to pass the bassline, and is therefore qualifies as an S wave |
|  | Initially a small positive wave (r), followed by a larger negative wave (S) |  | Initially a small negative wave (q), followed by a large positive wave (R) |
|  | The first wave is negative and small (q), followed by a large positive wave (R) and finally a small negative wave (s) |  | Notching on the upstoke of the R wave |
|  | Initially a larger negative (Q), then a large positive (R) |  | The negative deflection foes not manage to pass the baseline and can therefore qualify as an s wave. |
|  | A single negative wave is called a QS-complex |  |  |

### ECG waveform formatting

The ECG waves are recorded on paper with a 1mm2 grid. Voltage along the vertical axis against time on the horizontal axes. The ECG paper speed is 25mm/s as a result each 1mm horizontal box corresponds to 40ms. On some occasions the paper speed is increased to 50mm/s to better design the waveforms. For the vertical calibration, 10mm equates to 1mV with certain occasions having 20mm per 1mv.

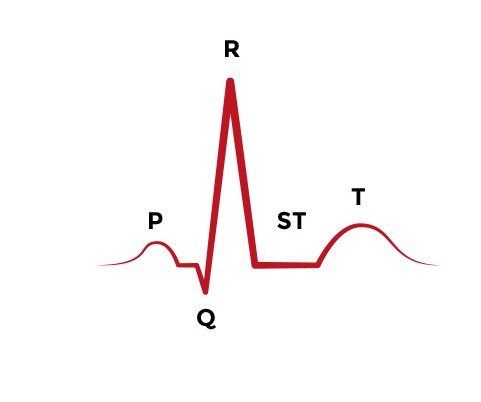
A picture containing text, hanger, document

Description automatically generated

Diagram

Description automatically generated

### Electrical principles of an ECG

The engineering definition from **XXX** of an ECG is a graph of the electric potential changed that occur between electrodes. The action potential created by contractions of the heart wall spreads electrical currents from the heart throughout the body. The spreading electrical currents creates different potentials at points throughout the body which can be measured through the electrodes.

The "P" wave corresponds to atrial contraction, and the "QRS" complex to the contraction of the ventricles. The "QRS' complex is much larger than the "P" wave due to the relative difference in muscle mass of the atria and ventricles, which masks the relaxation of the atria. The relaxation of the ventricles can be seen in the form of the "T" wave.

The electrocardiograph must be able to detect not only extremely weak signals ranging from 0.5 mV to 5.0 mV, but also a DC component of up to ±300 mV (resulting from the electrode-skin contact) and a common-mode component of up to 1.5 V, which results from the potential between the electrodes and the ground. The useful bandwidth of an ECG signal depends on the application and can range from 0.5-100 Hz, sometimes reaching up to 1 kHz. It is around 1 mV peak-to-peak in the presence of much larger external high frequency noise, 50 or 60 Hz interference, and DC electrode offset potential. Other sources of noise include movement that affects the skin-electrode interface, muscle contractions or electromyographic spikes, respiration (which may be rhythmic or sporadic), electromagnetic interference (EMI), and noise from other electronic devices that couple into the input.

## AFE4490 – Pulse Oximeter

A picture containing electronics, circuit

Description automatically generatedPulse Oximetry is an indirect method of measuring the oxygen levels in the blood. The sensor measures the amount of red and IR light wavelengths absorbed by blood to calculate the oxygen levels in blood. The measurement is done by a probe that clips on to a finger and contains emitters as well as a light sensor.

The breakout board I will be using is from ProtoCentral and shown in figure X. like the other ECG breakout a custom board can be created to integrate onto a single PCB. I have used this breakout board as a finger oximeter is non-evasive.

### What is Oxygen Saturation?

Pulse oximeters measure the haemoglobin in blood carrying oxygen i.e., oxygen saturation.

Haemoglobin without blood is called de-oxygenated haemoglobin (deoxy Hb), the hemoglobulin with oxygen is known as oxygenated haemoglobin (oxy Hb). Shape, icon

Description automatically generated

Icon

Description automatically generatedIcon

Description automatically generated

Application

Description automatically generated with low confidenceApplication

Description automatically generated

If a finger is placed between the light source and the light detector, part of the light will be absorbed by the finger and the unabsorbed light will reach the light detector. Hemoglobulin absorbs light and the amount of light absorbed is proportional to the concentration of haemoglobin in the blood vessel. This property is described in the law of physics as Beer’s Law. Having thicker veins means that the light will meet more hemoglobulin and thus with thicker veins there is a higher absorption rate.

Lamber’s Law: The amount of light absorbed is proportional to the length of the path that the light must travel in the absorbing substance.

Oxyhaemoglobin absorbs more infrared light than red light, however de-oxygenated haemoglobin absorbs more red light than infrared light. **Figure X** shows the graph showing each absorption rates.

A picture containing chart

Description automatically generatedDiagram

Description automatically generatedDiagram

Description automatically generatedDiagram

Description automatically generated

Diagram

Description automatically generatedDiagram

Description automatically generated

Using the two lights, infrared, and red light an algorithm can be created to deduce the oxygen saturation percentage.

### Setting up the AFE4490

The breakout board I will be using as stated is the AFE4490 breakout from ProtoCentral using the TI (Texas Instruments) chip. Using a DB7 connector to probe the finger. The breakout is connected to an ATMEGA328p microcontroller using the following pins as shown below in **figure X**. The SPI pins are required in those specific located dictated by the MCU SPI pins. The other pins can be changed, and very well may be for PCB routing.

|  |  |  |
| --- | --- | --- |
| AFE4490 pin label | Microcontroller Connection | Pin Function |
| GND | GND | GND |
| DRDY | D2 | Data ready(interrupt) |
| MISO | D12 | Slave out |
| SCK | D13 | SPI clock |
| MOSI | D11 | Slave in |
| CS0 | D7 | SLAVE SELECT |
| START | D5 | Conversion start Pin |
| PWDN | D4 | Power Down/ Reset |
| DIAG\_END | NC | Diagnostic output |
| LED\_ALM | NC | Cable fault indicator |
| PD\_ALM | NC | PD sensor fault indicator |
| VCC | +5v | Supply voltage |

The device is initiated the same way any SPI device is, setting the SPI clock speed (In this case 2MHz), followed by the SPI data mode (set to 0). Lastly, the bit order set to MSB first.

A twenty-two-bit data stream is produced via the chip and can be outputted directly using the UART communication protocol. As the data packet is larger than a byte, the data packet may need to be split into bytes and re-joined on the other end of the communication.

## MAX30205 – Body Temperature Monitor

MAX30205 is the new chip from Maxim which provides accurate human body temperature readings with an accuracy of 0.1 °C. This is a digital I2C temperature sensor, so an ADC would not be required to read this sensor. It also provides an over-temperature alarm as an interrupt to drive microcontroller interrupt lines. This sensor is tuned for the human body temperatures ranging from 37 Celsius to 39 Celsius.

Body temperature does not stay constant throughout the day and will vary throughout your lifetime too. Body temperature could be affected for example by  
- How active you are  
- What time of day it is  
- Your Age  
- Your Sex  
- What you have eaten or had to drink  
- Where you are in your menstrual cycle

Temperature can also differ depending on the location that is measured from. Under arm readings can be a degree lower than your mouth and rectal readings can be higher.

### What classifies as a Fever (CDC)?

From the definition from the Centre of Disease Control (CDC)

The 42 Code of Federal Regulations parts 70/71 Final Rule defines an ill person as someone who:

1. Has a fever (has a measured temperature of 100.4 °F [38 °C] † or greater, or feels warm to the touch, or gives a history of feeling feverish) **accompanied by one or more of the following**:
   * skin rash
   * difficulty breathing
   * persistent cough
   * decreased consciousness or confusion of recent onset
   * new unexplained bruising or bleeding (without previous injury)
   * persistent diarrhoea
   * persistent vomiting (other than air sickness)
   * headache with stiff neck, or
   * appears obviously unwell

**OR**

1. Has a fever that has persisted for more than 48 hours
2. Has symptoms or other indications of communicable disease, as the CDC may announce through posting of a notice in the Federal Register.
3. CDC considers a person to have a fever when he or she has a measured temperature of 100.4° F (38° C) or greater, or feels warm to the touch, or gives a history of feeling feverish. Note: Even though measured temperature is the preferred and most accurate method to determine fever, it is not always possible to take a person’s temperature. In certain situations, other methods of detecting a fever should be considered:
   * self-reported history of feeling feverish when a thermometer is not available, or the ill person has taken medication that would lower the measured temperature.
   * the person feels warm to the touch
   * appearance of a flushed face, glassy eyes, or chills if it is not feasible to touch the person or if the person does not report feeling feverish.

### What classifies as Hypothermia (CDC)?

* Hypothermia is caused by prolonged exposures to very cold temperatures. When exposed to cold temperatures, your body begins to lose heat faster than it is produced. Lengthy exposures will eventually use up your body’s stored energy, which leads to lower body temperature.
* Body temperature that is too low affects the brain, making the victim unable to think clearly or move well. This makes hypothermia especially dangerous, because a person may not know that it is happening and will not be able to do anything about it.
* While hypothermia is at very cold temperatures, it can occur even at cool temperatures (above 40°F) if a person becomes chilled from rain, sweat, or submersion in chilly water.

##### **Who is most at risk?**

Victims of hypothermia are often:

* Older adults with inadequate food, clothing, or heating
* Babies sleeping in cold bedrooms
* People who remain outdoors for extended periods—the homeless, hikers, hunters
* People who drink alcohol or use illicit drugs.

The following are warnings signs of hypothermia:

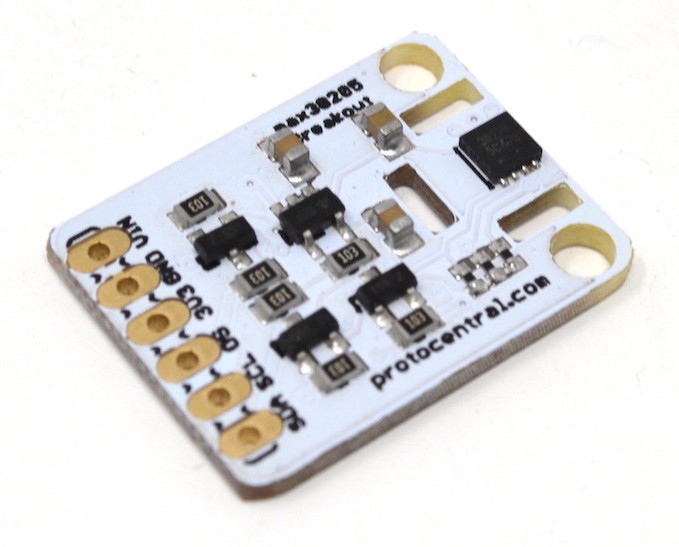
**Adults:**

* Shivering
* Exhaustion or feeling very tired
* Confusion
* Fumbling hands
* Memory loss
* Slurred speech
* Drowsiness

**Babies:**

* bright red, cold skin
* exceptionally low energy

### Interfacing the MAX30205

The MAX30205 board from ProtoCentral will be connected via I2C using the following connections in figure X. There will be two temperature sensors onboard but both devices will be connected to the same I2C bus as they operate with different I2C addresses.

|  |  |
| --- | --- |
| Micro-controller Pin Connection | MAX30205 Break-out |
| 5V | Vin |
| GND | GND |
| A4 | SDA |
| A5 | SCL |

Using the sample code provided via ProtoCentral, the Celsius value can be produces. All there is left is to send it to the host microcontroller to transmit the data via LoRa to be interpreted.

# SD Card

A close-up of a computer chip

Description automatically generated with low confidenceData will need to be logged for this project. The USB memory stick protocol will need additional time to implement and adds weight and bulk to this wearable device. An SD card is the most suitable choice for this project for the time being as the data will to be in real-time but for initial testing purposes, this will be used to log the data. The protocols for a micro-SD and a full-size SD card are the same. This will be communicated to the microcontroller via the SPI communication as explained earlier in the report.

The SD card operates at 3.3V logic levels and will need to be stepped down from 5V depending on the micro-controller.

A picture containing text, electronics, circuit

Description automatically generatedDepending on the SD card adapter used, some have a CD pin that shorts to ground when there is an SD card inserted. This is useful as trying to write to an SD card that is not present will crash the device.

|  |  |  |
| --- | --- | --- |
| SD pin label | Microcontroller Connection | Pin Function |
| GND | GND | GND |
| VCC | +5V | Supply Voltage |
| MISO | D12 | Slave out |
| SCK | D13 | SPI clock |
| MOSI | D11 | Slave in |
| CS0 | D10 | SLAVE SELECT |

There is a card-info sample code provided by ADAFRUIT that shows the information about the SD card, this is useful as settings will be known but also that the connection to the microcontroller is communicating.

Data can be saved to a CSV (Comma-separated values) file and later inputted into an excel document to create the graphs of the values for the heart, lungs, and temperature.

# Cloud Services

The top three competitors for cloud computing are Amazon Web Services (AWS). Microsoft Azure and Google Cloud Platform (GCP).

Amazon Web Services (AWS) is dominant in the 2020 report with a market share of 33%. With a vast set of tools and exponential growth, this would be suitable for an online platform for multi-user operation. Having the multiple user data uploaded to AWS for a qualified person to interpret the data. Unfortunately, the cost structure for AWS is confusing with different tiers depending on the services that is required. One year student membership can be obtained, however I have already used this for before the start of the module for hobbyist projects.

I have investigated AZURE and GCP with the free subscriptions I require more messages to be send in the given period. A message is defined as the data packet. Below is a list with a few pros and cons to compare for potential future iterations with additional funding.

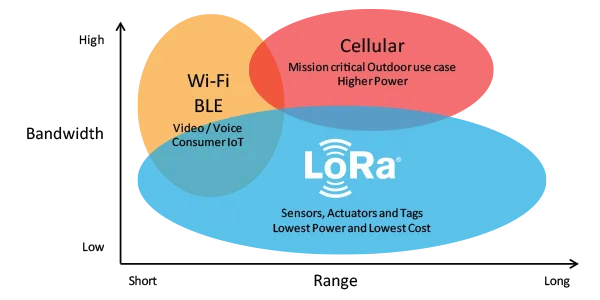
|  |  |
| --- | --- |
|  | |
| ****AWS Pros**** | **AWS Cons** |
| * Most services available, from networking to robotics * Most mature * Considered the gold standard in cloud reliability and security * More compute capacity vs Azure & GCP * All major software vendors make their programs available on AWS | * Dev/Enterprise support must be purchased * Can overwhelm newcomers with the sheer number of services and options * Comparatively limited options for hybrid cloud |
| **MICROSOFT AZURE** | |
| **Pros** | **Cons** |
| * Easy integration and migrations for existing Microsoft services * Many services available, including best-in-class AI, ML, and analytics services * Cheaper for most services vs AWS & GCP * Great support for hybrid cloud strategies | * Fewer service offerings vs AWS * Particularly geared towards enterprise customers |
| **GCP** | |
| **Pros** | **Cons** |
| * Plays nicely with other Google service and products * Excellent support for containerized workloads * Global fibre network | * Limited services vs AWS & Azure * Limited support for enterprise use cases |

# LoRa

## What is LoRa?

LoRa stands for long range. LoRa and LoRa-Wan together is defined as low power wide area network protocol. Connected with a battery to create an IOT (Internet of things) device. On a fundamental level LoRa is a radio protocol that is derived from the Chirp Spread Spectrum (CSS) technology. The data is encoded on radio waves using chirp pulses. This is like how dolphins and bats communicate. This modulated transmission is robust against disturbances and can be received at a great distance. As a wearable device this is perfect having a LoRa hub in the building receiving the signals from the body. If this product were designed for outdoor use, local storage would have been better or a bigger subscription to a cloud service.

This device can operate at a various of frequencies, from sub-gigahertz bands such as 915MHz, 868 MHz and 433MHz. Higher data rates can be achieved post-gigahertz for example on the 2.5GHz band.

Lora WAN is suitable for transmitting small size payloads (like sensor data) over long distances. LoRa modulation provides a significantly greater communication range with low bandwidths than other competing wireless data transmission technologies. The following **figure X** shows some access technologies that can be used for wireless data transmission and their expected transmission ranges vs. bandwidth.

A LoRa node usually operates on a battery and consists of a radio module and a microprocessor. A LoRa gateway is also a microprocessor with a radio module but is normally operated over AC mains.

A few key points to mention from The Things Network (**XX**):

* **Ultra-low power** - Lora WAN end devices are optimized to operate in low power mode and can last up to 10 years on a single coin cell battery.
* **Long range** - Lora WAN gateways can transmit and receive signals over ten kilometres in rural areas and up to three kilometres in dense urban areas.
* **Deep indoor penetration** - Lora WAN networks can provide deep indoor coverage, and easily cover multi floor buildings.
* **License free spectrum** - You do not have to pay expensive frequency spectrum license fees to deploy a Lora WAN network.
* **Geolocation**- A Lora WAN network can determine the location of end devices using triangulation without the need for GPS. A LoRa end device can be located if at least three gateways pick up its signal.
* **High capacity** - Lora WAN Network Servers handle millions of messages from thousands of gateways.
* **Public and private deployments** - It is easy to deploy public and private Lora WAN networks using the same hardware (gateways, end devices, antennas) and software (UDP packet forwarders, Basic Station software, Lora WAN stacks for end devices).
* **End-to-end security**- Lora WAN ensures secure communication between the end device and the application server using AES-128 encryption.
* **Firmware updates over the air** - You can remotely update firmware (applications and the Lora WAN stack) for a single end device or group of end devices.
* **Roaming**- Lora WAN end devices can perform seamless handovers from one network to another.
* **Low cost** - Minimal infrastructure, low-cost end nodes and open-source software.
* **Certification program**- The LoRa Alliance certification program certifies end devices and provides end-users with confidence that the devices are dependable and compliant with the Lora WAN specification.
* **Ecosystem**- Lora WAN has an exceptionally large ecosystem of device makers, gateway makers, antenna makers, network service providers, and application developers.

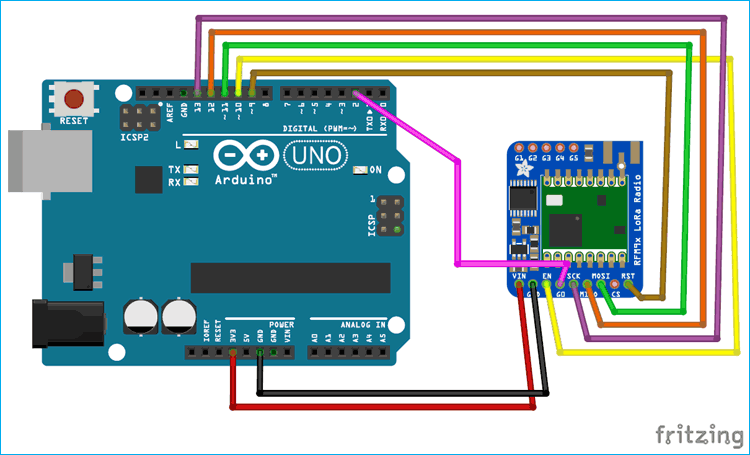
## Where is LoRa used?

* **Vaccine cold chain monitoring** - Lora WAN sensors are used to ensure vaccines are kept at appropriate temperatures in transit.
* **Animal conservation** - Tracking sensors manage endangered species such as Black Rhinos and Amur Leopards.
* **Dementia patients** - Wristband sensors provide fall detection and medication tracking.
* **Smart farms**- Real time insights into crop soil moisture and optimized irrigation schedule reduce water use up to 30%.
* **Water conservation**- Identification and faster repair of leaks in a city’s water network.
* **Food safety**- Temperature monitoring ensures food quality maintenance.
* **Smart waste bins** - Waste bin level alerts sent to staff optimize the pickup schedule.
* **Smart bikes**- Bike trackers track bikes in remote areas and dense buildings.
* **Airport tracking** - GPS-free tracking monitors vehicles, personnel, and luggage.
* **Efficient workspaces** - Room occupancy, temperature, energy usage and parking availability monitoring.
* **Cattle health** - Sensors monitor cattle health, detect diseases, and forecast calf’s’ delivery time.
* **LoRa in space** - Satellites to provide Lora WAN-based coverage worldwide.

## Interfacing the SX1278 LoRa Module to the Microcontroller

The LoRa module that will be used is the SX1278, more commonly known as the RA-02 LoRa board. This board operates at 433MHz. LPD433 (LPD = low power device) is an Ultra High Frequency (UHF) band which is a free communication is allowed to be operated in the UK as it meets the requirement for the Office of communications (OFCOM).

The LoRa module will be connected to an ATMEGA2560, the more powerful version of the ATMEGA328-P. This module also connects using the SPI protocol as shown in **table X and figure X**. For test purposes, The ATMEGA328-P will be used as the addition features will be used at a later stage.



Micro-controller

PC

LoRa Radio Module



Micro-controller

LoRa Radio Module

|  |  |  |
| --- | --- | --- |
| LoRa SX1278 Module | Microcontroller UNO Board | Pin Function |
| 3.3V | 3.3V | Power |
| GND | GND | Ground |
| EN/NSS | D10 | SPI Enable |
| G0/DIO0 | D2 | Interrupt Pin |
| SCK | D13 | SPI CLOCK |
| MISO | D12 | MASTER IN SLAVE OUT |
| MOSI | D11 | MASTER OUT SLAVE IN |
| RST | D9 | RESET |

Note that the power supply for this is 3.3V rather than the 5V from our sensors, this is critical as there is not a 5V version and the chip will be damaged. Secondly, connecting to the 3.3V is suitable for sending a few data packets, however the Arduino UNO pin only supplies 50mA, while the maximum working current of the RA-01/RA-02 is stated to be 105mA but, through self-testing can sink 200mA. In the receiving state the power characteristic is exceptionally low at a receiving current of 13mA and a standby current at 1.6mA. however, this does not matter too much as the receiving device is connected to mains, but for future updates where communication is bidirectional this is good to know.

## SX1278 Library/Code example

The library I will be using for this project, is the LoRa Radio library by Sandeep Mistry as this library is very well documented and compatible with different SX-XXXX chips. This allows an upgrade from the RA-01 to the RA-02 and so forth if needed.

Using the Arduino IDE Library Manager

* + Sketch – Include Library – Manage Libraries – Search LoRa Radio

Using Git

* + cd ~/Documents/Arduino/libraries/
  + git clone https://github.com/sandeepmistry/arduino-LoRa LoRa

This has the following libraries

* LoRa chip information, register values, set spread.
* LoRa Duplex
* LoRa Duplex Call-back
* LoRa Receiver
* LoRa Receiver call-back
* LoRa Sender
* LoRa Sender Non-Blocking
* LoRa Sync
* LoRa Simple Gateway
* LoRa Simple Node

Luckily, the subroutines have simplified the main code. A “ initiates the LoRa module with also a Boolean call-back if the device fails to initiate. The speed for 433 MHz will be written in the brackets as 433E6. A simple for the data transmission with a print statement in between to send the data. Below is **Figure X** being the flowchart showing the process.

Y

Initialise LoRa SPI

Is USART DATA Ready

Data = Read USART

LoRa begin  
transmission

Send Data Via LoRa

LoRa end transmission

START

r

Y

N

# Power Management

The Lo-Ion & LiPoly batteries that is considered is from ADAFRUIT and so the ADAFRUIT specification for battery management has been listed below:

Lithium ion/polymer batteries are extremely power dense. This makes them great for reducing size and weight of projects. However, they are not 'safe' batteries and require extra care. Charging or using the batteries incorrectly can cause explosion or fire.

There are five main things to watch for when charging and using batteries:

* Do not charge them **above** their maximum safe voltage (say 4.2V) - usually taken care of by any on-cell protection circuit
* Do not discharge them **below** their minimum safe voltage (say 3.0V) - usually taken care of by any on-cell protection circuit
* Do not **draw more current**than the battery can provide (say about 1-2**C**) - usually taken care of by any on-cell protection circuit
* Do not **charge them with more current** than the battery can take (say about 1**C**) - usually taken care of by any on-cell protection circuit but also set with the charger by adjusting the charge rate
* Do not charge the batteries **above or below**certain temperatures (usually about 0-50 degrees C) - sometimes managed by the charger, but often not an issue if the charge rate is reasonable.

A picture containing text, electronics, circuit

Description automatically generatedFor specifics on each battery, you must look at the datasheet to know what the safe voltages, currents and temperatures are - they can vary from cell to cell.

For the first three items, a circuit board attached to the battery can monitor the battery voltage and the current going out. These are often referred to simply as **protection circuits**. They are quite common on standard batteries, but **you must check the datasheet** **or product image** to verify that a protection circuit is attached.

Figure X shows a battery management system for a 1s LiPo from ADAFRUIT. This compact circuit is suitable for this project, but alternatively the schematic for this is only and so the entire project is possible to be on a single board.

Alternatively, A portable power bank can be used for this project, relieving the need for a power management system and issues with a LiPo as stated above.

# PCB Design

As stated in this report, a custom PCB will be made to house this entire project. Unfortunately, due to component shortages and delays in manufacturing and shipping I have decided on a different approach to this world-wide issue. Creating a break-out board for the sensors to plug into.

The board will have three microcontrollers using DIP-28 Sockets allowing me to source the microcontrollers from my past projects and rehouse them for this project. The microcontrollers run of an external clock using a 16MHz crystal oscillator and there will be a synchronised reset to reset all three controllers at once. An LCD has been added as an afterthought for data to be read from.

With no prior knowledge, the PCB has been created in KiCad and manufactured at JLCPCB. Alongside the core circuitry, additional pins have been attached to future additions. Also, should there be an error, this will not need to be remanufactured as the pins are swappable to a different pin should I need to. **Figure X** shows the sensor board. See appendix **XXX** for all the PCB Layers and schematics.

The dimension for this PCB is 100mm by 52mm, which allows it to fit nicely as a watch on the wrist or on the belt tucked away from sight.

A screenshot of a video game

Description automatically generated

A picture containing text, electronics, circuit

Description automatically generated

**A close-up of a circuit board

Description automatically generated with medium confidenceFigure XXX** shows the sensors and LCD connected to the custom sensor board. The DB-7 connector and audio connector have been positions there purposely for ease of connection when the 3D printed housing is attached.

A close-up of a circuit board

Description automatically generated with medium confidenceThe DIP-28 sockets you can see behind the LCD, are for the ATMEGA328-P microcontroller. For this version there is no onboard programmer as the IC is removable to be programmed separately.

See appendix **XXX** for all the PCB Layers and schematics.

The second PCB shown in **figure XX** is the mainboard PCB, this houses the more powerful microcontroller the ATMEGA2560 microcontroller. This microcontroller has additional USART communications and so can communicate with all the sensors. An SD CARD will be attached via a break-out board as the break-out board is cheaper than having the SD card soldered. The LoRa SX chip will also be on a breakout board as the chip I was planning to use is currently out of stock.

As this board uses USB, a differential pair has been created to match the trace lengths. The micro-USB is attached but also a breakout for the microSD just in case for any shortages.

A picture containing application

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

# Bill of Material (BoM)

\* Already owned.  
\*\* On loan from the University.

|  |  |  |  |
| --- | --- | --- | --- |
| Core Components | Price | Quantity | Notes |
| ADS1292R | £33.43 | 1x | <https://protocentral.com/product/ads1292r-ecg-respiration-breakout-kit/> |
| MAX30205 | £8.43 | 2x | <https://protocentral.com/product/protocentral-max30205-body-temperature-sensor-breakout-board/> |
| AFE44090 | £56.73 | 1x | <https://protocentral.com/product/protocentral-afe4490-pulse-oximeter-breakout-board-kit/> |
| ATMEGA2560 | N/A \*\* |  | <https://tinyurl.com/bddjpeap> |
| ATMEGA328-P | N/A \* | 3x | <https://tinyurl.com/38845t8u> |
|  |  |  |  |
| Power Bank | N/A \* | 1x | <https://tinyurl.com/4647stad> |

|  |  |  |  |
| --- | --- | --- | --- |
| SMD Components | Price | Quantity | Notes |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Services | Price | Quantity | Notes |
| 3D printing PLA |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Appendices

AFE4490 Datasheet –

<https://www.ti.com/lit/ds/symlink/afe4490.pdf>

haemoglobin gif –

<https://www.howequipmentworks.com/wp-content/uploads/2015/01/graph_animation.gif>

<https://www.howequipmentworks.com/pulse_oximeter/>

<https://github.com/Protocentral/protocentral-afe4490-arduino>

<https://github.com/Protocentral/AFE4490_Oximeter>

SD card

<https://cdn-learn.adafruit.com/downloads/pdf/adafruit-micro-sd-breakout-board-card-tutorial.pdf>

LoRa

<https://www.thethingsnetwork.org/docs/lorawan/what-is-lorawan/>

<https://en.wikipedia.org/wiki/Ultra_high_frequency>

<https://en.wikipedia.org/wiki/Ofcom>

<https://en.wikipedia.org/wiki/LPD433>

<https://github.com/Lora-net>

<https://github.com/sandeepmistry/arduino-LoRa/blob/master/API.md>

<https://circuitdigest.com/microcontroller-projects/arduino-lora-sx1278-interfacing-tutorial>

<https://github.com/sandeepmistry/arduino-LoRa>

ECG

<https://www.nhs.uk/conditions/electrocardiogram/>

<https://github.com/Protocentral/ADS1292rShield_Breakout>

<https://www.hopkinsmedicine.org/health/treatment-tests-and-therapies/electrocardiogram#:~:text=The%20electrodes%20are%20connected%20to,flowing%20the%20way%20it%20should>.

<https://github.com/Protocentral/protocentral-ads1293-arduino>

<https://protocentral.com/product/protocentral-ads1293-breakout-board/>

<https://www.ti.com/lit/ds/symlink/ads1292r.pdf?ts=1649935030076&ref_url=https%253A%252F%252Fwww.google.com%252F>

<https://github.com/Protocentral/ADS1292rShield_Breakout>

<https://protocentral.com/product/ads1292r-ecg-respiration-breakout-kit/>

<https://www.jove.com/v/10473/acquisition-and-analysis-of-an-ecg-electrocardiography-signal#:~:text=Electrical%20signals%20are%20produced%20by,electrical%20activity%20in%20an%20ECG>.

<https://www.uptodate.com/contents/ecg-tutorial-basic-principles-of-ecg-analysis#:~:text=The%20ECG%20paper%20speed%20is,sec%20(200%20ms)%20intervals>.

<https://www.aclsmedicaltraining.com/basics-of-ecg/>

CLOUD

<https://www.bmc.com/blogs/aws-vs-azure-vs-google-cloud-platforms/>

<https://intellipaat.com/blog/aws-vs-azure-vs-google-cloud/>

<https://www.datamation.com/cloud/aws-vs-azure-vs-google-cloud/>

temp

<https://www.cdc.gov/quarantine/air/reporting-deaths-illness/definitions-symptoms-reportable-illnesses.html>

<https://www.cdc.gov/disasters/winter/staysafe/hypothermia.html>

## Heart Sensor Code

**ADS1292R Sensor Code**

#include "protocentralAds1292r.h"

#include "ecgRespirationAlgo.h"

#include <SPI.h>

volatile uint8\_t globalHeartRate;

volatile uint8\_t globalRespirationRate=0;

const int ADS1292\_DRDY\_PIN = 26;

const int ADS1292\_CS\_PIN = 13;

const int ADS1292\_START\_PIN = 14;

const int ADS1292\_PWDN\_PIN = 27;

int16\_t ecgWaveBuff, ecgFilterout;

int16\_t resWaveBuff,respFilterout;

long timeElapsed=0;

ads1292r ADS1292R;

ecg\_respiration\_algorithm ECG\_RESPIRATION\_ALGORITHM;

void setup()

{

delay(2000);

SPI.begin();

SPI.setBitOrder(MSBFIRST);

//CPOL = 0, CPHA = 1

SPI.setDataMode(SPI\_MODE1);

// Selecting 1Mhz clock for SPI

SPI.setClockDivider(SPI\_CLOCK\_DIV16);

pinMode(ADS1292\_DRDY\_PIN, INPUT);

pinMode(ADS1292\_CS\_PIN, OUTPUT);

pinMode(ADS1292\_START\_PIN, OUTPUT);

pinMode(ADS1292\_PWDN\_PIN, OUTPUT);

Serial.begin(57600);

ADS1292R.ads1292Init(ADS1292\_CS\_PIN,ADS1292\_PWDN\_PIN,ADS1292\_START\_PIN); //initalize ADS1292 slave

Serial.println("Initiliziation is done");

}

void loop()

{

ads1292OutputValues ecgRespirationValues;

boolean ret = ADS1292R.getAds1292EcgAndRespirationSamples(ADS1292\_DRDY\_PIN,ADS1292\_CS\_PIN,&ecgRespirationValues);

if (ret == true)

{

ecgWaveBuff = (int16\_t)(ecgRespirationValues.sDaqVals[1] >> 8) ; // ignore the lower 8 bits out of 24bits

resWaveBuff = (int16\_t)(ecgRespirationValues.sresultTempResp>>8) ;

if(ecgRespirationValues.leadoffDetected == false)

{

ECG\_RESPIRATION\_ALGORITHM.ECG\_ProcessCurrSample(&ecgWaveBuff, &ecgFilterout); // filter out the line noise @40Hz cutoff 161 order

ECG\_RESPIRATION\_ALGORITHM.QRS\_Algorithm\_Interface(ecgFilterout,&globalHeartRate);// calculate

//disable below 2 lines if you want to run with arduino uno. (arduino uno does not have the memory to do all processing together)

respFilterout = ECG\_RESPIRATION\_ALGORITHM.Resp\_ProcessCurrSample(resWaveBuff);

ECG\_RESPIRATION\_ALGORITHM.RESP\_Algorithm\_Interface(respFilterout,&globalRespirationRate);

}else{

ecgFilterout = 0;

respFilterout = 0;

}

if(millis() > timeElapsed) // update every one second

{

if(ecgRespirationValues.leadoffDetected == true) // lead in not connected

{

Serial.println("ECG lead error!!! ensure the leads are properly connected");

}else{

Serial.print("Heart rate: ");

Serial.print(globalHeartRate);

Serial.println("BPM");

Serial.print("Respiration Rate :");

Serial.println(globalRespirationRate);

}

timeElapsed += 1000;

}

}

}

**ADS1292R Sensor Code**

#include "protocentralAds1292r.h"

#include "ecgRespirationAlgo.h"

#include <SPI.h>

volatile uint8\_t globalHeartRate = 0;

volatile uint8\_t globalRespirationRate=0;

const int ADS1292\_DRDY\_PIN = 6;

const int ADS1292\_CS\_PIN = 7;

const int ADS1292\_START\_PIN = 5;

const int ADS1292\_PWDN\_PIN = 4;

int16\_t ecgWaveBuff, ecgFilterout;

int16\_t resWaveBuff,respFilterout;

ads1292r ADS1292R;

ecg\_respiration\_algorithm ECG\_RESPIRATION\_ALGORITHM;

void setup()

{

delay(2000);

SPI.begin();

SPI.setBitOrder(MSBFIRST);

//CPOL = 0, CPHA = 1

SPI.setDataMode(SPI\_MODE1);

// Selecting 1Mhz clock for SPI

SPI.setClockDivider(SPI\_CLOCK\_DIV16);

pinMode(ADS1292\_DRDY\_PIN, INPUT);

pinMode(ADS1292\_CS\_PIN, OUTPUT);

pinMode(ADS1292\_START\_PIN, OUTPUT);

pinMode(ADS1292\_PWDN\_PIN, OUTPUT);

Serial.begin(57600);

ADS1292R.ads1292Init(ADS1292\_CS\_PIN,ADS1292\_PWDN\_PIN,ADS1292\_START\_PIN);

Serial.println("Initiliziation is done");

}

void loop()

{

ads1292OutputValues ecgRespirationValues;

boolean ret = ADS1292R.getAds1292EcgAndRespirationSamples(ADS1292\_DRDY\_PIN,ADS1292\_CS\_PIN,&ecgRespirationValues);

if (ret == true)

{

ecgWaveBuff = (int16\_t)(ecgRespirationValues.sDaqVals[1] >> 8) ; // ignore the lower 8 bits out of 24bits

resWaveBuff = (int16\_t)(ecgRespirationValues.sresultTempResp>>8) ;

if(ecgRespirationValues.leadoffDetected == false)

{

ECG\_RESPIRATION\_ALGORITHM.ECG\_ProcessCurrSample(&ecgWaveBuff, &ecgFilterout); // filter out the line noise @40Hz cutoff 161 order

ECG\_RESPIRATION\_ALGORITHM.QRS\_Algorithm\_Interface(ecgFilterout,&globalHeartRate); // calculate

//respFilterout = ECG\_RESPIRATION\_ALGORITHM.Resp\_ProcessCurrSample(resWaveBuff);

//ECG\_RESPIRATION\_ALGORITHM.RESP\_Algorithm\_Interface(respFilterout,&globalRespirationRate);

}else{

ecgFilterout = 0;

respFilterout = 0;

}

Serial.println(ecgFilterout);

//Serial.println(resWaveBuff);

}

}

## Lungs Sensor Code

**AFE4490 Sensor Code**

#include <SPI.h>

#include "Protocentral\_AFE4490\_Oximeter.h"

const int SPISTE = 7; // chip select

const int SPIDRDY = 2; // data ready pin

const int PWDN =4;

const int DRDY\_INTNUM =0; //digital pin2 interrupt num = 0. Please pass correct interrupt number if you are using any boards otherthan arduino uno

AFE4490 afe4490;

int32\_t heart\_rate\_prev=0;

int32\_t spo2\_prev=0;

void setup()

{

Serial.begin(57600);

Serial.println("Intilaziting AFE44xx.. ");

delay(2000) ; // pause for a moment

SPI.begin();

SPI.setClockDivider (SPI\_CLOCK\_DIV8); // set Speed as 2MHz , 16MHz/ClockDiv

SPI.setDataMode (SPI\_MODE0); //Set SPI mode as 0

SPI.setBitOrder (MSBFIRST); //MSB first

afe4490.afe44xxInit (SPISTE, SPIDRDY, DRDY\_INTNUM, PWDN);

Serial.println("intilazition is done");

}

void loop()

{

afe44xx\_output\_values afe4490Data;

boolean sampled\_value = afe4490.getDataIfAvailable(&afe4490Data,SPISTE);

if(sampled\_value == true)

{

if(afe4490Data.spo2 == -999){

Serial.println("Probe error!!!!");

}else if ((heart\_rate\_prev != afe4490Data.heart\_rate) || (spo2\_prev != afe4490Data.spo2)){

heart\_rate\_prev = afe4490Data.heart\_rate;

spo2\_prev = afe4490Data.spo2;

Serial.print("calculating sp02...");

Serial.print(" Sp02 : ");

Serial.print(afe4490Data.spo2);

Serial.print("% ,");

Serial.print("Pulse rate :");

Serial.print(afe4490Data.heart\_rate);

Serial.println(" bpm");

}

}

}

**AFE4490 Sensor Code**

#include <SPI.h>

#include "Protocentral\_AFE4490\_Oximeter.h"

const int SPISTE = 7; // chip select

const int SPIDRDY = 2; // data ready pin

const int PWDN =4;

const int DRDY\_INTNUM =0; //digital pin2 interrupt num = 0. Please pass correct interrupt number if you are using any boards otherthan arduino uno

AFE4490 afe4490;

void setup()

{

Serial.begin(57600);

Serial.println("Intilaziting AFE44xx.. ");

delay(2000) ; // pause for a moment

SPI.begin();

SPI.setClockDivider (SPI\_CLOCK\_DIV8); // set Speed as 2MHz , 16MHz/ClockDiv

SPI.setDataMode (SPI\_MODE0); //Set SPI mode as 0

SPI.setBitOrder (MSBFIRST); //MSB first

//NOTE: usually the DRDY\_INTNUM is same as the pin but the interrupt number for arduino uno pin2 is 0.

afe4490.afe44xxInit (SPISTE, SPIDRDY, DRDY\_INTNUM, PWDN);

Serial.println("intilazition done");

}

void loop()

{

afe44xx\_output\_values afe4490Data;

boolean sampled\_value = afe4490.getDataIfAvailable(&afe4490Data,SPISTE);

if (afe4490Data.calculated\_value == true)

{

Serial.println(afe4490Data.red);

//Serial.println(afe4490Data.ir);

}

}

#include <SPI.h>

#include "Protocentral\_AFE4490\_Oximeter.h"

const int SPISTE = 25; // chip select

const int SPIDRDY = 13; // data ready pin

const int PWDN =4;

AFE4490 afe4490;

void setup()

{

Serial.begin(57600);

Serial.println("Intilaziting AFE44xx.. ");

delay(2000) ; // pause for a moment

SPI.begin();

SPI.setClockDivider (SPI\_CLOCK\_DIV8); // set Speed as 2MHz , 16MHz/ClockDiv

SPI.setDataMode (SPI\_MODE0); //Set SPI mode as 0

SPI.setBitOrder (MSBFIRST); //MSB first

afe4490.afe44xxInit (SPISTE, SPIDRDY, SPIDRDY, PWDN);

Serial.println("intilazition done");

}

void loop()

{

afe44xx\_output\_values afe4490Data;

boolean sampled\_value = afe4490.getDataIfAvailable(&afe4490Data,SPISTE);

if (afe4490Data.calculated\_value == true)

{

Serial.println(afe4490Data.red);

//Serial.println(afe4490Data.ir);

}

}

## Body Temperature Sensor Code

**MAX30205 Sensor Code**

#include <Wire.h>

#include "Protocentral\_MAX30205.h"

MAX30205 tempSensor;

void setup() {

Serial.begin(9600);

Wire.begin();

//scan for temperature in every 30 sec untill a sensor is found. Scan for both addresses 0x48 and 0x49

while(!tempSensor.scanAvailableSensors()){

Serial.println("Couldn't find the temperature sensor, please connect the sensor." );

delay(30000);

}

tempSensor.begin(); // set continuos mode, active mode

}

void loop() {

  float temp = tempSensor.getTemperature(); // read temperature for every 100ms

  Serial.print(temp ,2);

  Serial.println("'c" );

  delay(100);

}

TEST