Measurement of water's weight and height with Arduino Mega

Overview

This is a system that has the main purpose to use for measuring and monitoring 3 things every 10 minutes including:

1. Distance
2. Weight of water
3. Height of water level

In addition, the data is sent to mobile phone via SMS and also written to MicroSD card.

This system is designed to work for Arduino mega with 5 external IoT devices which are ultrasonic sensor called HC-SR04, weight scale called Load Cell, HX711 amplifier broad, MicroSD card adapter, and GPRS+GPS module called A7 Ai-Thinker.

Library

This system is required 3 libraries consisted of 2 Arduino libraries and 1 contributed library including:
1. SD – this library allows reading from and writing to SD cards.
2. SPI (Serial Peripheral Interface) – this library allows communicating with SPI devices.
3. HX711 – this library allows HX711 24-bit analog-to-digital converter for load cell to work. (src: https://github.com/bogde/HX711)

Materials

1. Arduino Mega
2. Ultrasonic sensor (HC-SR04)
3. Weight scale (Load Cell)
4. HX711 amplifier board
5. MicroSD card adapter
6. GPRS+GPS module (A7 Ai-Thinker)
7. MicroSD card
8. Sim card
9. Arduino IDE

Installation

How to install Arduino IDE?

For Linux: https://www.arduino.cc/en/Guide/Linux (see also the Arduino playground page https://playground.arduino.cc/Learning/Linux)

For macOS X: https://www.arduino.cc/en/Guide/MacOSX

For Windows: https://www.arduino.cc/en/Guide/Windows

<u>How to install IoT devices?</u>

- Arduino Mega and HX711 amplifier

```
Arduino Mega pin 6 -> HX711 CLK
Arduino Mega pin 5 -> HX711 DOUT
Arduino Mega pin 5V -> HX711 VCC
Arduino Mega pin GND -> HX711 GND
```

Figure 1: Connection between Arduino Mega and HX711

- Arduino Mega and A7 Ai-Thinker

```
Arduino Mega TX3 -> A7 Ai-Thinker RX
Arduino Mega RX3 -> A7 Ai-Thinker TX
A7 Ai-Thinker 5V -> A7 Ai-Thinker PWR
```

Figure 2: Connection between Arduino Mega and A7 Ai-Thinker

- HX711 amplifier and Load Cell

```
HX711 E+ -> Load Cell RED
HX711 E- -> Load Cell BLACK
HX711 A- -> Load Cell WHITE
HX711 A+ -> Load Cell GREEN
```

Figure 3: Connection between Load Cell and HX711

- Arduino Mega and MicroSD card adapter

```
Arduino Mega pin 53 -> MicroSD card adapter SS
Arduino Mega pin 51 -> MicroSD card adapter MOSI
Arduino Mega pin 50 -> MicroSD card adapter MISO
Arduino Mega pin 52 -> MicroSD card adapter SCK
```

Figure 4: Connection between Arduino Mega and MicroSD card adapter

<u>How to calibrate the scale</u>

The Load cell is calibrated with a "2 known weight method", a method suggested by David. This method will need to take 2 weight and height at 2 points of the scale, and then create a formula to turn any weight read by the Load cell into the height of the water.

$$currentHeight = Height1 - (\frac{Weight1 - currentWeight}{Weight1 - Weight2} \times (Height1 - Height2))$$

The formula can reliably calculate the height of the water, even though there's some deviation.

Description of code

This program is understandably separated into 4 parts including:

1. Libraries' import and variables' declaration part – importing 3 libraries that is stated above, defining constant variables and normal variables, declaring objects, and setting some pins. In addition, output mode to boot, sleep, and wake up the GSM board is set here. This method will put the device into a low power mode, sleep, to save power between read, which can be 12 hours or a day apart from another. And back to normal state when it needs to read and send data. This will also help the device to continue to work after a power loss happen by putting a code that will boot up after a power loss.

2. Setup part (setup()) – this is the main part that is used to initialize and set the initial value of variables. Moreover, it is used to set pin mode, calibrate the scale, initialize the SD library, and set the maximum baud rate of Arduino Mega as 115200.

3. Execution part (loop()) – this is another main part that is used to actively control the Arduino Mega board for execution by calling the external functions.

4. Addition function part
   This part consists of 4 functions including:
   - calculateDistance() is used to calculate the actual distance by applying the raw data that stems from HC-SR04 with this formula; distance = time x speed of sound.

```
Speed of sound = 340 m/s = 34000 cm/s = 34 cm/ms = 0.034 cm/µs
time = distance / speed
distance = time x speed
distance apart (cm) = time x speed / 2 = time x 0.034 / 2
therefore, d = duration*0.034/2.0
```

Figure 5: Distance Formula's Proof

   - calculateWeightHeight() is used to calculate the weight and height of water by applying the raw data that stems from Load Cell and HX711 amplifier board with two-known-weight method. This method needs to calibrate the Load cell at 2 points in the tube.
   For example.
     - Fill the water and record the depth and value read by the Load cell and calls it "height1" and "weight1".
     - Fill more water to record the second known weight, and calls it "height2" and "weight2".
     - After this we have all the variable to calculate the height of water by knowing just the weight read by Load cell.
     - In order to get the currentHeight, we use the formula as follow.

$$currentHeight = Height1 - (\frac{Weight1 - currentWeight}{Weight1 - Weight2} \times (Height1 - Height2))$$

Figure 6: Two-known-weight calibration method

- writeSD() is used to record the data into MicroSD card by using SD library.
- SendSMS() is the function that uses AT command to send the data to mobile phone via SMS.

```
Serial3.println("AT+CMGF=1");    // set input and output format of messages to Text mode.
delay(5000);                     // Delay to let the system finish the work
Serial3.print("AT+CMGS=\"");     // Begin to send SMS to given number
Serial3.print(phone_no);         // Phone number here
Serial3.write(0x22);             // "
Serial3.write(0x0D);             // /r
Serial3.write(0x0A);             // /n              /r/n = new line in Windows
```

Figure 7: Example of AT command