International Journal of Multidisciplinary Research and Modern Education (IJMRME)
Impact Factor: 6.725, ISSN (Online): 2454 - 6119
(www.rdmodernresearch.org) Volume 3, Issue 1, 2017

AUTOMATIC INDICATION SYSTEM OF GLUCOSE LEVEL IN GLUCOSE TRIP BOTTLE

S. Gayathri* & C. S. Sundar Ganesh**

* UG Student, Department of Robotics and Automation Engineering, PSG College of Technology, Coimbatore, Tamilnadu

** Assistant Professor, Department of Robotics and Automation Engineering, PSG College of Technology, Coimbatore, Tamilnadu

Cite This Article: S. Gayathri & C. S. Sundar Ganesh, "Automatic Indication System of Glucose Level in Glucose Trip Bottle", International Journal of Multidisciplinary Research and Modern Education, Volume 3, Issue 1, Page Number 148-151, 2017.

Abstract:

In this paper, we found out the level of glucose in the glucose trip bottles that are used in the hospitals. When the glucose bottle is about to be emptied, an alert message is made to send to the nurses working there. Three important modules are used for this purpose. Load sensor that can measure weight is used as the input module which is used to send data to the controller. Arduino ATmega board is used as a controller module that process the data received from the sensor.GSM module is used as the output module which based on the control given by the controller, it sends alert message to the nurses' phone number.

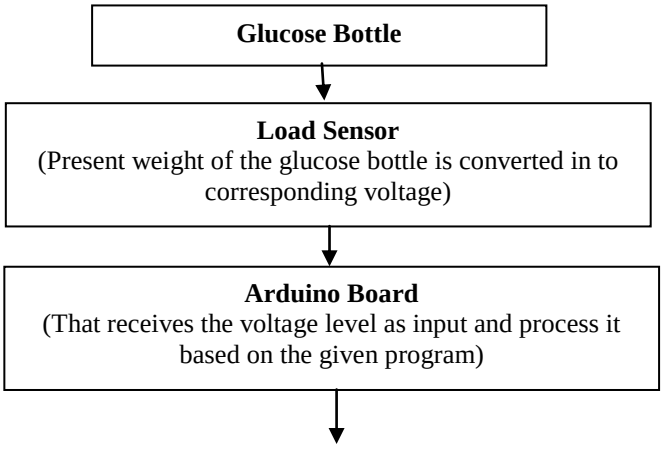
Key Words: Scaling, Atmega Controller, GSM Module & Glucose Trip Controller

1. Introduction:

Due to heavy workloads people even forget to do their most important work. If nurses working in hospitals forget to change the glucose trip bottle once it is emptied, it will bring a bad consequences to the patient. To remind the nurses, we can send an alert message to their phones saying that glucose bottle is going to be empty, as everyone will continuously use their mobile phones in this modern world. The system consists of micro controller which receives input from the load sensor and it send output to the GSM module. An alerting signal using sound alarms for replacing the glucose trip bottles is being used in very few hospitals. The sound alarms may not be heard by nurses if they are not too close enough and obviously sound cannot be increased as it is a hospital. So replacing the sound alarms with the alerting phone message may be still more efficient since everyone will use their phone always and thus using a GSM module is not so much costlier and it can be easily implemented. Coming to the setup, the overall system can be made to get stick on the glucose bottle and it is a temporary sticking and therefore they can be used for the next glucose bottles. The load sensor can also be made to hang in the bottle stand in which the glucose bottle can be made to hang. Therefore using a GSM module to send an alerting message is an efficient method. A wireless sensor system can be used for intravenous dripping system, which can detect when an intravenous liquid, provided to patients in hospitals, run out, as well as detecting obstructions in the catheter(1). IV DRIP is a simple, low-cost, mechanical automatic volume regulator to deliver intravenous fluid in low-resource settings. The device consists of two levers such as an IV bag hangs on the upper lever, while a counterweight hangs on the lower notched lever. The position of this counterweight dictates the volume of fluid dispensed. When the target volume is delivered, the levers tip and kink the IV tubing, stopping fluid flow and thus preventing over hydration (2). An automated system will be designed to detect the level of the Intra-Venous fluid and to send this critical data over a wireless transmitter. The data sent will be displayed in a dashboard (3).

2. Block Diagram:

The overall system consists of a input module (load sensor), a controller and an output module (GSM module) of Figure 1



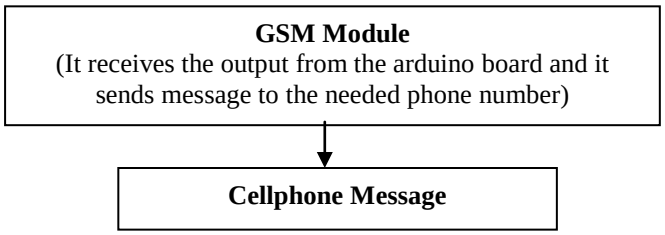


Figure 1: Flowchart of the system

The load sensor consists of two types of output such as inverting output and non-inverting output. The S-type load cell is hanged in the glucose hanger and the glucose bottle is hanged at the other end in the S-type load cell. Thus the weight of the bottle measured is converted into the corresponding voltage and hence the output from the load cell is in the form of voltage and the voltage level is magnified to the certain level and the non-inverting output of the load cell is given as input to the one of the PWM pins in ATMEGA board. The PWM pins output will be varying from 0-255. We use the load cell range from 0-750 gm. So the maximum value of PWM pins values will be 255 when the load cell can measure to its maximum range. This is called as scaling. But for our application the glucose bottle maximum weight is 500 which is equal to 170 in the PWM output. Thus once when the bottle is full the PWM pin output will be 170 at the time GSM module will send the message as "the bottle is full". The bottle's level is continuously measured, if there is any interrupt in measuring the glucose level, GSM module will send the message as "check the apparatus". And when the glucose level reaches half that is the PWM pin output is 85, the GSM module will send the message as "the bottle is reduced to 50%". And when the glucose level reaches the 10% that is the PWM output is 17, the emergency alert is generated and the alert message is sent as" emergency, the bottle is reduced to 10%".The power supply for the GSM module and the ATmega board is given using 12V adapter. Table 1 shows the Characteristics of glucose replacement schedule.

Table 1: Characteristics of glucose replacement schedule

S.No	Weight	PWM Values	Message
1	500 g	170	The bottle is full
2	250 g	85	The bottle is reduced to 50%
3	50 g	17	Emergency, the bottle is reduced to 10%

Thus this the circuit diagram of interfacing ATmega 2560 with the GSM module is shown in figure 2

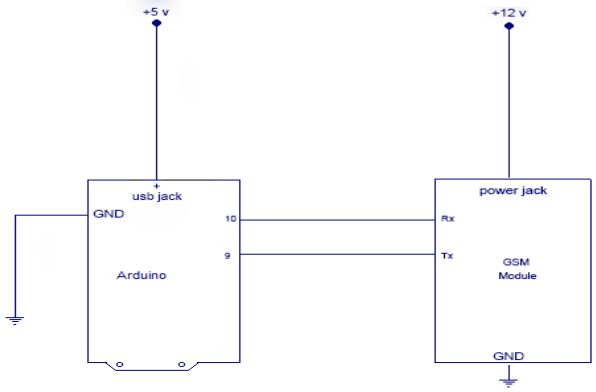


Figure 2: Circuit Diagram

3. Results and Discussion:

All the components that we are going to use for the glucose level indicator are bought and are tested separately to know whether each component is working correctly. We have a circuit using load sensor which gives the varied output that is from 0-10mV. Now the sensing part is over. Then the ARDUINO board is tested first by setting some inputs and output pins and doing programming for any other simple application. Then the output from the load sensor is made to be given as input of ARDUINO board and thus programming part it ready which was already inserted into the board so that as soon the input us given, the corresponding required output pin shows some great response and thus the ARDUINO board is also tested separately with the correct program. Then the GSM module is tested separately. First for any other simple application, it is checked and the three pins available in the module are connected to the supplies and to the output from the ARDUINO board. Based on the programming given and for the three stages it correctly sends the messages to the phone number given in the program and the GSM module is also tested. Thus the whole set up is tested and the output is got. A sample output is shown in Fig.3 in which an example for "bottle full" condition and "bottle level is half" condition is shown

3. Results and Discussion:

Figure 3: Sample Output

4.Conclusion:

Thus as the name indicates, the "glucose level indicator" is used to indicates the level or amount of glucose liquid present in the trip bottles in hospitals. Once this glucose level indicator comes into existence the people taking care of the patients and their in-charged nurses need not worry about the time that will take for the glucose bottle to be emptied. They can do the works whatever they want and thus the messages indicating the present situation of the glucose bottle is sent to the places where they are and it is sent as a warning message that the bottle will be emptied soon and thus they can go at the right time and replace the glucose bottle as soon as that gets empty. Thus the people in the hospital need not worry about anything and do their work without any tension. The entire concept can be achieved using "glucose level indicator".

5. References:

1. Paul Bustamante, Gonzalo Solas, Karol Grandez, Unai Bilbao," A new Wireless Sensor for Intravenous Dripping Detection", International journal on advances in networks and services, vol 3,no 1& 2,2012.
2. Bailey Flynn, Matthew Nojoomi, Michael Pan, Kamal Shah, Erica Skerrett, "IV DRIP: Intravenous Dehydration Relief in Pediatrics", instituteof global health technology.
3. Wireless Sensor Network for Intra-Venous fluid level indicator application an IEEE journal
4. K.J. Ayala, "The 8051 Microcontroller ", 3rd Edition, Microcontrollers Pros Corporation, 2004.
5. simon monk, "programming arduino".
6. Richard blam,"sams teach yourself arduino programming".
7. www.cicuitstoday.com

6. Appendix:

The program below is to communicate between the device and the phone number provided through the GSM module.

```
#include <SoftwareSerial.h>
SoftwareSerialmySerial(9, 10); // receiver and transmitter pins
constint positive = 13, ground = 12; // the pins are made as input supply
intconst invert = 11; //load sensor's inverting signal
intconstnon_invert = 10; //load sensor's non -inverting signal
voidSendMessage()
{
mySerial.println("AT+CMGF=1"); //Sets the GSM Module in Text Mode
delay(1000); // Delay of 1000 milli seconds or 1 sec
mySerial.println("AT+CMGS=\"+91xxxxxxxxxx\"\r"); //Enter mobile number
delay(1000);
mySerial.println(" SMS from GSM Module");//SMS text you want to send
delay(100);
mySerial.println((char)26); // ASCII code of CTRL+Z
delay(1000);
}
void setup()
{
```

```
digitalWrite(positive, HIGH);
digitalWrite(ground, LOW); //input supply
Serial.begin(9600); //GSM bandwidth
mySerial.begin(9600); // Setting the baud rate of GSM Module
delay(100);
pinMode(invert, INPUT);
pinMode(non_invert, INPUT);
pinMode(10,OUTPUT); //10 is transmitter and 9 is receiver
pinMode(9,INPUT)
}
void loop()
{
if (Serial.available() > 0)
{
while(analogRead(invert))
{
if(analogRead(invert)>170)
{
mySerial.println("AT+CMGF=1");//Sets the GSM Module in Text Mode
delay(1000); // Delay of 1000 milli seconds or 1 second mySerial.println("AT+CMGS=\"+91xxxxxxxxxx\"\r");
// type the mobile no delay(1000);
mySerial.println("The bottle is full");// The SMS text you want to send delay(100);
mySerial.println((char)26); // ASCII code of CTRL+Z  delay(1000);
}
else if(analogRead(invert)==85)
{
mySerial.println("AT+CMGF=1");//Sets the GSM Module in Text Mode
delay(1000);                          // Delay of 1000 milli seconds or 1 second
mySerial.println("AT+CMGS=\"+91xxxxxxxxxx\"\r"); // enter the mobile no
delay(1000);
mySerial.println("The bottle is reduced to 50%"); // text you want to send
delay(100);
mySerial.println((char)26); // ASCII code of CTRL+Z
delay(1000);
  }
else if (analogRead(invert)<=17)
mySerial.println("AT+CMGF=1");//Sets the GSM Module in Text Mode   delay(1000);  // Delay of 1000 milli
seconds or 1 second
mySerial.println("AT+CMGS=\"+91xxxxxxxxxx\"\r"); // enter the mobile no
delay(1000);
mySerial.println("Emergency the bottle is reduced to 10%") ; // The SMS text you want to send
delay(100);
mySerial.println((char)26); // ASCII code of CTRL+Z
delay(1000);
}
else
  {
mySerial.println("AT+CMGF=1");    //Sets the GSM Module in Text Mode
delay(1000);                          // Delay of 1000 milli seconds or 1 second
mySerial.println("AT+CMGS=\"+91xxxxxxxxxx\"\r");//enter the mobile no delay(1000);
mySerial.println("Check the apparatus"); // text you want to send
delay(100);
mySerial.println((char)26); // ASCII code of CTRL+Z
delay(1000);
  }
if (mySerial.available() > 0)
Serial.write(mySerial.read());
}
```