Railway Track Crack and Fault Detection System Using GPS GSM

P Jagadeesh [1], A Raju Kumar Reddy [2], M Vinay [3],

Affrose[4]

[1-4]Department Of Internet of Things, Malla Reddy University, Hyderabad, Telangana, India.
[5]Assistant Professor, Department of Internet of Things, Malla Reddy University, Hyderabad, Telangana, India.

2111cs050023@mallareddyuniversity.ac.in [1], 2111cs050036@mallareddyuniversity.ac.in [2], 2111cs050050@mallareddyuniversity.ac.in [3], roseaffrose557@gmail.com[4]

ABSTRACT

Railways are the most preferable transport system because of their reliability, passenger safety, and ease to travel. If any misalignment or crack occurs, it creates a loss of lives. If these cracks and misalignments are not taken care of early, they may result in derailments and eventually result in a significant loss of life. To overcome this issue, a railway track crack detection system is proposed. The components like Global Position System (GPS), BUZZER, IR SENSOR, ULTRASONIC SENSOR, ARDUINO, Global System for Mobile (GSM), BATTERY, and D.C. MOTOR will be used. The cracks are detected through sensor rays. GPS can send the position of the explicit area and GSM sends the message to the station premises. The sensor detects the crack and sends a message to the maintenance room, and all connections are made with ARDUINO. Here in the place of the train, a wheeled robot will be used with the help of the dc motor. This concept will give high accuracy and no problem will be occurred during detection and saves so many lives. It can be used for railway stations and used in metro trains, gaming systems, and any crack detection system**.**

1. INTRODUCTION

Depending on the fast developments in railway systems, high-speed trains are used for speed transportation, and rai1 transportation is increased day by day. The most of the people uses railway for transportation because it is essentia1 for transferring the goods and passengers from one place to another p1ace and low cost. And also, the railway system is providing faci1ity such as high speed, with economica1, environment friend1y, safety, and better characteristics of rai1ay systems.

These can be done by time to time maintenance and control measurements. But depending on different factors, deformations and derailment may occur on the superstructure of railways, because of improper maintenance and the currently irregular and manual track line monitoring mistake from workers. Such deformation is determining on time and taking precautions is very important for the safety of railway systems. Therefore, solution for this problem is introducing in this project. To provide the protection to the railway accident because of cracks occurs in the railway track. This system is used in-between two stations which will detect the cracks present on the track using IR sensors which transmit sine waves for an ideal track. If a crack is detected then this sensor will send a signal to the Arduino Uno board which will activate the GPS receiver. The GPS receiver will send the exact location which will then be messaged to the main authorities. Once the sensor sends a signal to the controller, the controller will activate the webcam. The webcam will provide the live video of the track. The live video and the data from the GPS will be updated in the designed application of the wireless camera. Using this smart technology will be a part of the brave new digitalized world which will be able to prevent the loss of precious life and property.

1.1 Introduction of Embedded System:

An Embedded System is a combination of computer hardware and software, and perhaps additional mechanical or other parts, designed to perform a specific function. A good example is the microwave oven. Almost every household has one, and tens of millions of them are used every day, but very few people realize that a processor and software are involved in the preparation of their lunch or dinner.
This is in direct contrast to the personal computer in the family room. It too is comprised of computer hardware and software and mechanical components (disk drives, for example).

1.2 History and Future:

Given the definition of embedded systems earlier is this chapter; the first such systems could not possibly have appeared before 1971. That was the year Intel introduced the world's first microprocessor. This chip, the 4004, was designed for use in a line of business calculators produced by the Japanese Company Busicom. In 1969, Busicom asked Intel to design a set of custom integrated circuits-one for each of their new calculator models. The 4004 was Intel's response rather than design custom hardware for each calculator, Intel proposed a general- purpose circuit that could be used throughout the entire line of calculators. Intel's idea was that the software would give each calculator its unique set of features. The microcontroller was an overnight success, and its use increased steadily over the next decade. Early embedded applications included unmanned space probes, computerized traffic lights, and aircraft flight control systems. In the 1980s, embedded systems quietly rode the waves of the microcomputer age and brought microprocessors into every part of our kitchens (bread machines, food processors, and microwave ovens), living rooms (televisions, stereos, and remote controls), and workplaces (fax machines, pagers, laser printers, cash registers, and credit card readers).

1.3 Real Time Systems:

One subclass of embedded is worthy of an introduction at this point. As commonly defined, a real-time system is a computer system that has timing constraints. In other words, a real-time system is partly specified in terms of its ability to make certain calculations or decisions in a timely manner. These important calculations are said to have deadlines for completion. And, for all practical purposes, a missed deadline is just as bad as a wrong answer.

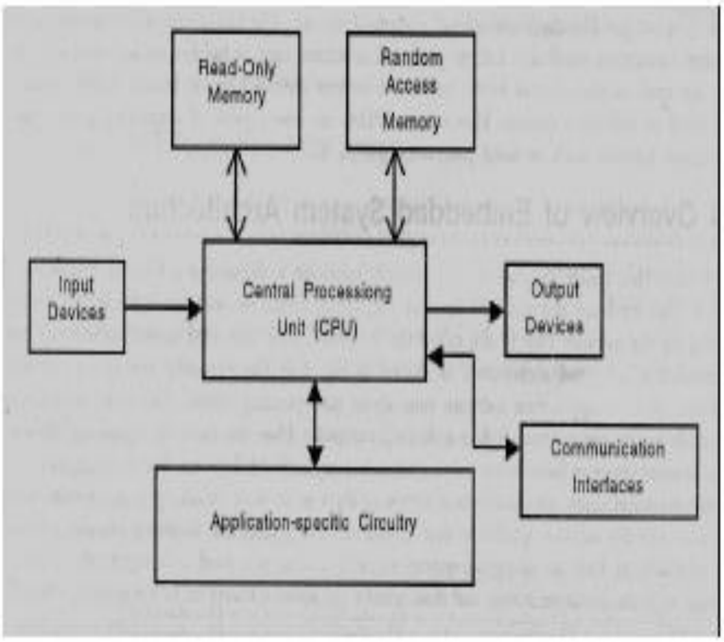
The issue of what if a deadline is missed is a crucial one. For example, if the real-time system is part of an airplane's flight control system, it is possible for the lives of the passengers and crew to be endangered by a single missed deadline. However, if instead the system is involved in satellite communication, the damage could be limited to a single corrupt data packet. The more severe the consequences, the more likely it will be said that the deadline is "hard" and thus, the system is a hard real-time system. Real-time systems at the other end of this discussion are said to have"soft"deadlines.

1.4 Overview of Embedded System Architecture

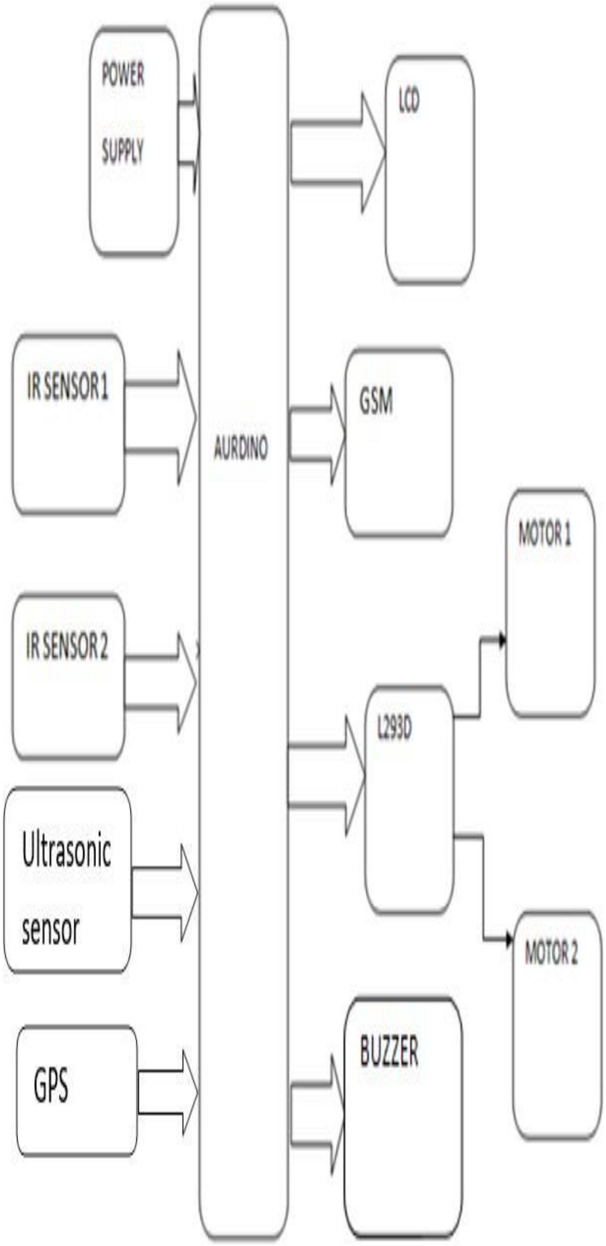
Every embedded system consists of custom-built hardware built around a Central Processing Unit (CPU). This hardware also contains memory chips onto which the software is loaded. The software residing on the memory chip is also called the 'firmware'. The embedded system architecture can be represented as a layered architecture as shown in Fig. The operating system runs above the hardware, and the application software runs above the operating system. The same architecture is applicable to any computer including a desktop computer. However, there are significant differences. It is not compulsory to have an operating system in every embedded system. For small appliances such as remote control units, air conditioners, toys etc., there is no need for an operating system and you can write only the software specific to that application. For applications involving complex processing, it is advisable to have an operating system. In such a case, you need to integrate the application software with the operating system and then transfer the entire software on to the memory chip. Once the software is transferred to the memory chip, the software will continue to run fora long time you don't need to reload new software.

Now, let us see the details of the various building blocks of the hardware of an embedded system. As shown in Fig. the building blocks are:

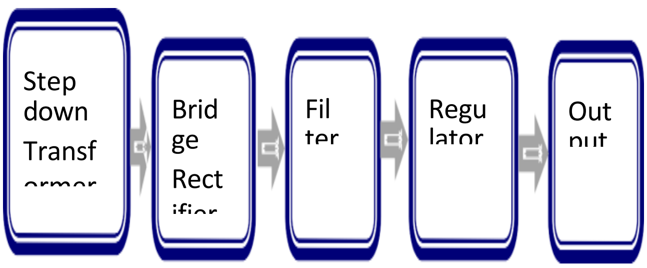
- Central Processing Unit (CPU)
- Memory (Read-only Memory and Random Access Memory)
- Input Devices
- Output devices
- Communication interfaces
- Application-specific circuitry



2.BLOCK DIAGRAM OF PROJECT AND FUNCTIONING



POWER SUPPLY BLOCKDIAGRAM:



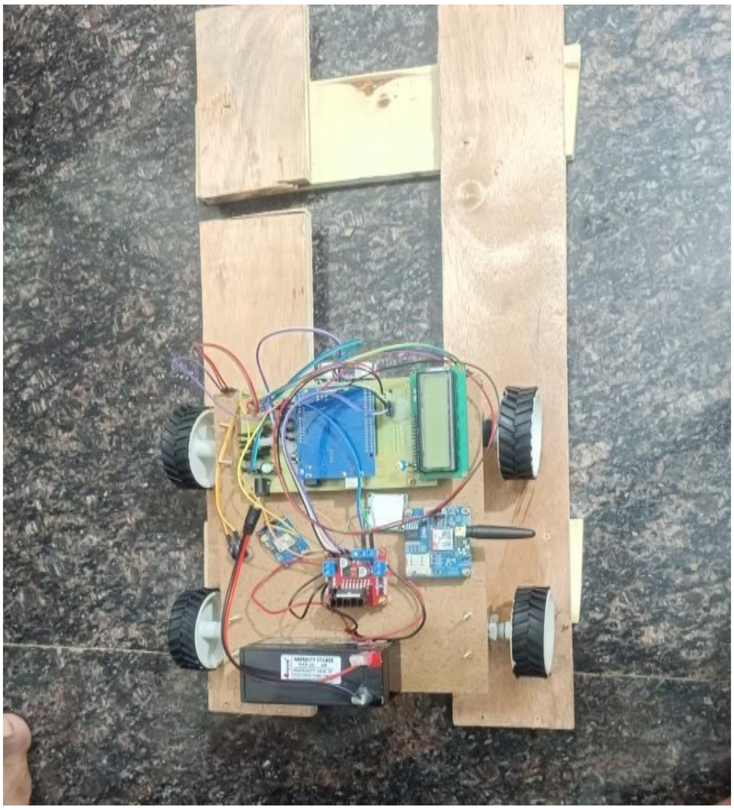
3. Code

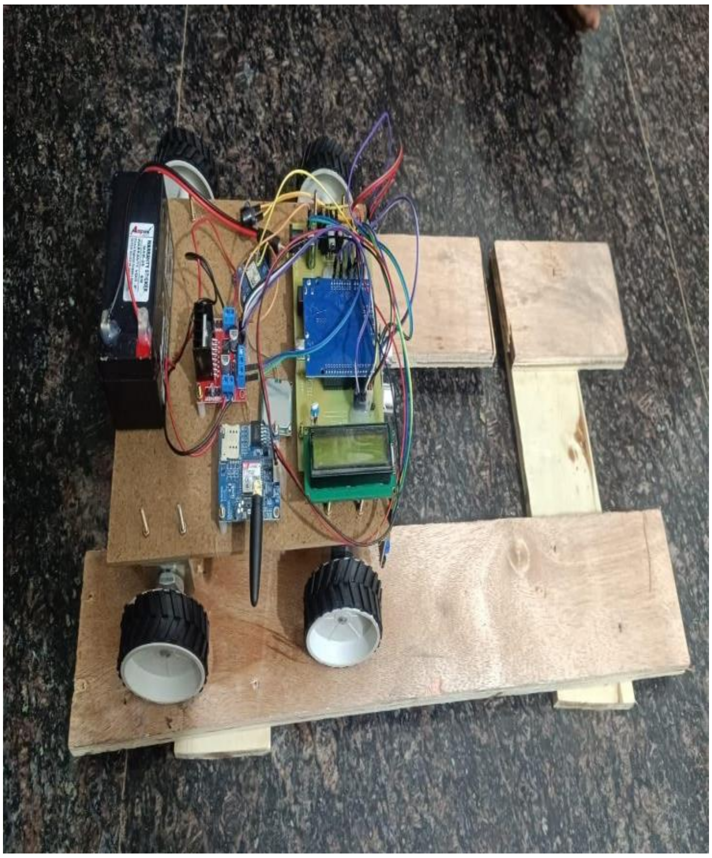
```
#include
#include
#include
LiquidCrystal
lcd(13, 12, 11, 10, 9, 8);
SoftwareSerial mySerial(2, 3);
String textMessage;
#define m11 4
#define m12 5
#define m21 6
#define pump1 7 const int trigPin = A0;
const int echoPin = A1;
long duration;
int distance;
int servoPin = A4;
Servo Servo1;
intmemsx=0,memsy=0;
int hbtc=0,hbtc1=0,rtrl=0;
unsigned char rcv,count,gchr='x',gchr1='x',robos='s';
char rcvmsg[10],pastnumber[11];
//char pastnumber1[11],pastnumber2[11];
//pastnumber3[11];
int ii=0,rchkr=0;
float tempc=0,weight=0;
float vout=0;
int sti=0;
String inputString = "";
// a string to hold incoming data
boolean stringComplete = false;
// whether the string is complete
{
 unsigned char rcr;
do{
 rcr = mySerial.read();
 }
while(rcr != 'K');
}
{
lcd.begin(16,2);
Serial.begin(115200);
mySerial.begin(9600);
Servo1.attach(servoPin);
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);
lcd.setCursor(0,0);
lcd.print(" WELCOME ");
pinMode(m11,OUTPUT);
pinMode(m12,OUTPUT);
pinMode(m21,OUTPUT);
pinMode(pump1,OUTPUT);
digitalWrite(m11,LOW);
digitalWrite(m12,LOW);
digitalWrite(m21,LOW);
digitalWrite(pump1,LOW);
Serial.println("Initializing...");
 Servo1.write(0); delay(1000);
```

```
gsminit();
serialEvent();
lcd.clear();
char memss='x';
{
int IR1 = digitalRead(A2);
int IR2 = digitalRead(A3);
digitalWrite(trigPin, LOW);
delayMicroseconds(2);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
int B= duration*0.034/2;
if(IR1 == LOW)
{
lcd.clear();
lcd.print("Train Entry");
lcd.setCursor(0, 1);
lcd.print("Gate Closed");
digitalWrite(pump1,HIGH);
delay(2000);
digitalWrite(m11,HIGH);
digitalWrite(m12,LOW);
digitalWrite(m21,LOW);
digitalWrite(pump1,LOW);
Servo1.write(160);
delay(1000);
mySerial.write("AT+CMGS=\"");
mySerial.write(pastnumber);
mySerial.write("\"\r\n");
delay(3000);
mySerial.write(" Train Entry ");
mySerial.write("\"\r\n");
delay(300);
mySerial.write(0x1A);delay(4000);delay(4000);
}
if(IR2 == LOW)
{
lcd.clear();
lcd.print("Train Exit");
lcd.setCursor(0, 1);
lcd.print("Gate Open");
digitalWrite(pump1,HIGH);
delay(2000);
digitalWrite(m11,LOW);
digitalWrite(m12,HIGH);
digitalWrite(m21,LOW);
digitalWrite(pump1,LOW);
Servo1.write(0);
delay(1000);
mySerial.write("AT+CMGS=\"");
mySerial.write(pastnumber);
mySerial.write("\"\r\n");
delay(3000);
mySerial.write(" Train Exit ");
mySerial.write("\"\r\n"); delay(300);
mySerial.write(0x1A);delay(4000);delay(4000);
}
}
if(IR1 == HIGH && IR2 == HIGH && B0)
{
textMessage = mySerial.readString();
Serial.print(textMessage);
delay(10);
}
}
void serialEvent()
{
while (mySerial.available())
{
char inChar = (char)mySerial.read();
//sti++;
//inputString += inChar;
if(inChar == '*')
{
sti=1;
inputString += inChar;
// stringComplete = true;
// gchr = inputString[sti-1]
}
if(sti == 1)
{
inputString += inChar;
}
if(inChar == '#')
{
sti=0;
stringComplete = true;
}
}
}
int readSerial(char result[])
{
int i = 0;
while (1)
{
while (mySerial.available() > 0)
{
char inChar = mySerial.read();
if (inChar == '\n')
{
result[i] = '\0';
mySerial.flush();
return 0;
}
if (inChar != '\r')
{
result[i] = inChar; i++;
}
}
}
}
int readSerial1(char result[])
{
int i = 0;
while (1)
{
```

```
while
(mySerial.available() > 0)
{
char inChar = mySerial.read();
if (inChar == '*')
{
result[i] = '\0';
mySerial.flush();
return 0;
}
if (inChar != '*')
{
result[i] = inChar;
i++;
}
}
}
}
void gsminit()
{
mySerial.write("AT\r\n");
mySerial.write("ATE0\r\n");
mySerial.write("AT+CMGF=1\r\n");
okcheck();
okcheck();
okcheck();
mySerial.write("AT+CNMI=2,2,0,0,0\r");
okcheck();
mySerial.write("AT+CSMP=17,167,0,0\r\n");
okcheck();
Serial.print("SEND MSG STORE");
Serial.print("MOBILE NUMBER");
lcd.clear();
lcd.print("SEND MSG STORE");
lcd.setCursor(0, 1);
lcd.print("MOBILE NUMBER");
do{
rcv = mySerial.read();
}
while(rcv != '*');
readSerial(pastnumber);
pastnumber[10]='\0';
Serial.print(pastnumber);
lcd.clear();
lcd.print(pastnumber);
delay(1000);
mySerial.write("AT+CMGS=\"");
mySerial.write(pastnumber);
mySerial.write("\"\r\n");
delay(3000);
mySerial.write("registered\r\n");
mySerial.write(0x1A);
delay(4000);
delay(5000);
//delay(1000);
}
```

4 Results





5. Conclusion

The proposed system has the ability to detect the cracks and obstacles if any on the track. There are many advantages with the proposed system as compared with the traditional detection techniques which include low cost, low power consumption, fast detecting system without human intervention and less analysis time. By this model we can easily avoid train accidents and derailments so that many human lives can be saved.

6. References

[1] Vikas Desai, "Design and Implementation of GSM and GPS Based Vehicle Accident Detection System", IJIT, Vol 01, Issue 03,pp. 1- 4,2013.

[2] C.Prabha, R.Sunitha, R.Anitha, "Automatic Vehicle Accident Detection and Messaging System Using GSM And GPS Modem", IJAREEIE, Vol. 3, Issue 7, pp. 1-5, 2014.

[3] Vikram Singh Kushwaha, Deepa Yadav, "Car Accident Detection System using Gps, Gsm and Bluetooth" in IJERGS May-June 2015.

[4]AboliRavindraWakure,ApurvaRajendraPatkar, "Vehicle Accident Detection and Reporting System using Gps And Gsm.", IJERGS, Vol 10,Issue 4,pp.- 25- 28, April 2014. [5] N. Watthanawisuth, "Wireless Black Box using MEMS Accelerometer and GPS Tracking for Accidental Monitoring of Vehicles", IEEE conference in Jan, 2012.

[6] Hoang Dat Pham, "Development of vehicle tracking system using GPS and GSM modem" IEEE conference in Dec, 2013.

[7] Rashida Nazir, Ayesha Tariq, Sadia Murawwat, Sajjad Rabbani,"Accident Prevention and Reporting System using GSM (SIM 900D) and GPS (NMEA 0183)", Int. J. Communications, Network and System Sciences, 2014, 7, 286-293 Published Online August 2014.

[8] Ravi Shankar Shekhar, Purushottam Shekhar, Ganesan P, "Automatic Detection of Squats in Railway Track", Innovations in Information Embedded and Communication Systems, 2015

[9] Sanket Salvi, Shashank Shetty , "AI based Solar Powered Railway Track Crack Detection and Notification System with Chatbot Support", Third International Conference on I-SMAC, 2019

[10] B. S. Sathish, Ganesan P, A. Ranganayakulu, Dola Sanjay S, Jagan Mohan Rao, "Advanced Automatic Detection of Cracks in Railway Tracks", International Conference on Advanced Computing & Communication Systems (ICACCS), 2019

[11] Nilisha Patil , Dipakkumar Shahare , Shreya Hanwate , Pranali Bagde , Karuna Kamble, Prof. Manoj Titre, "Designing of Improved Monitoring System for Crack Detection on Railway Tracks", d International Conference on Intelligent Communication Technologies and Virtual Mobile Networks, 2021.