Automatic Railway Gate Control & Track Switching

ABSTRACT:

Present project is designed using 8051 microcontroller to avoid railway accidents happening at unattended railway gates, if implemented in spirit. This project utilizes two powerful IR transmitters and two receivers; one pair of transmitter and receiver is fixed at up side (from where the train comes) at a level higher than a human being in exact alignment and similarly the other pair is fixed at down side of the train direction. Sensor activation time is so adjusted by calculating the time taken at a certain speed to cross at least one compartment of standard minimum size of the Indian railway. We have considered 5 seconds for this project. Sensors are fixed at 1km on both sides of the gate. We call the sensor along the train direction as 'foreside sensor' and the other as 'aft side sensor'. When foreside receiver gets activated, the gate motor is turned on in one direction and the gate is closed and stays closed until the train crosses the gate and reaches aft side sensors. When aft side receiver gets activated motor turns in opposite direction and gate opens and motor stops. Buzzer will immediately sound at the fore side receiver activation and gate will close after 5 seconds, so giving time to drivers to clear gate area in order to avoid trapping between the gates and stop sound after the train has crossed.

The same principle is applied for track switching. Considering a situation wherein an express train and a local train are traveling in opposite directions on the same track; the express train is allowed to travel on the same track and the local train has to switch on to the other track. Two sensors are placed at the either sides of the junction where the track switches. If there's a train approaching from the other side, then another sensor placed along that direction gets activated and will send an interrupt to the controller. The interrupt service routine switches the track. Indicator lights have been provided to avoid collisions. Here the switching operation is performed using a stepper motor. Assuming that within a certain delay, the train has passed the track is switched back to its original position, allowing the first train to pass without any interruption. This concept of track switching can be applied at 1km distance from the stations.

The project is simple to implement and subject to further improvement.

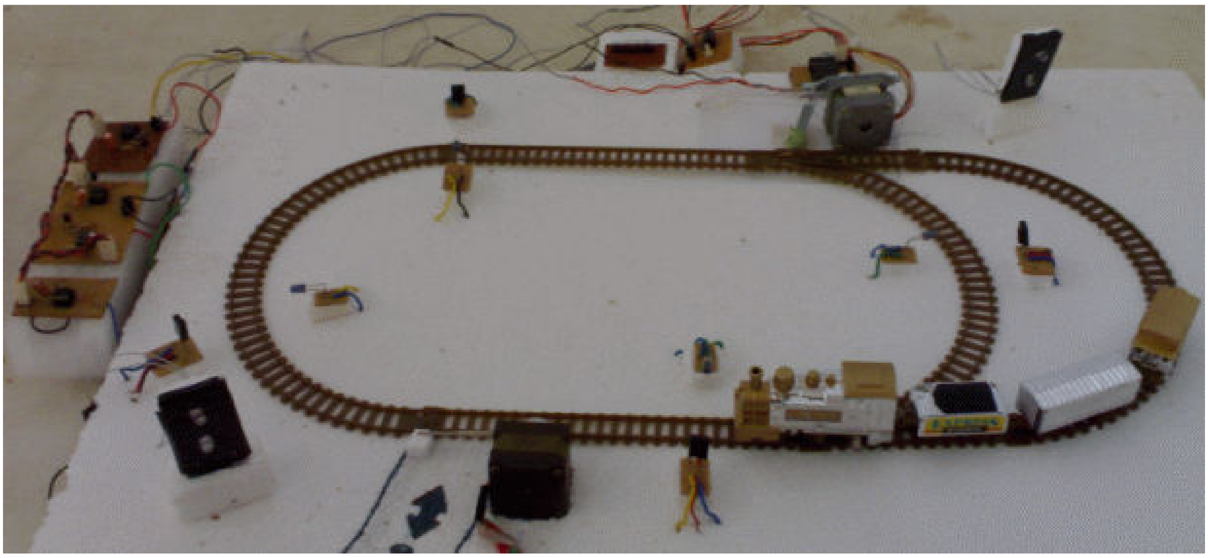
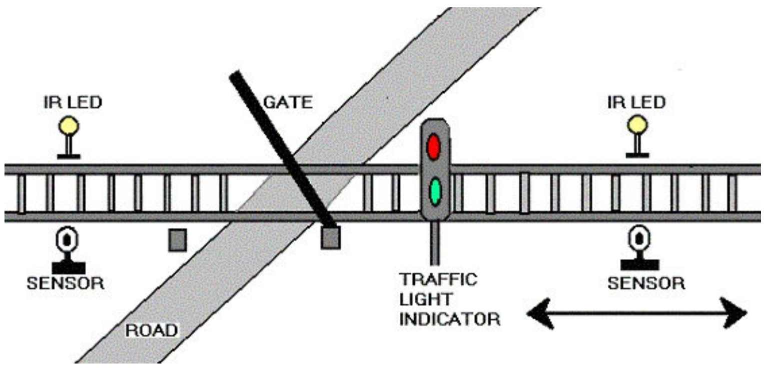


Fig: Model of Automatic Railway Gate Control & Track Switching

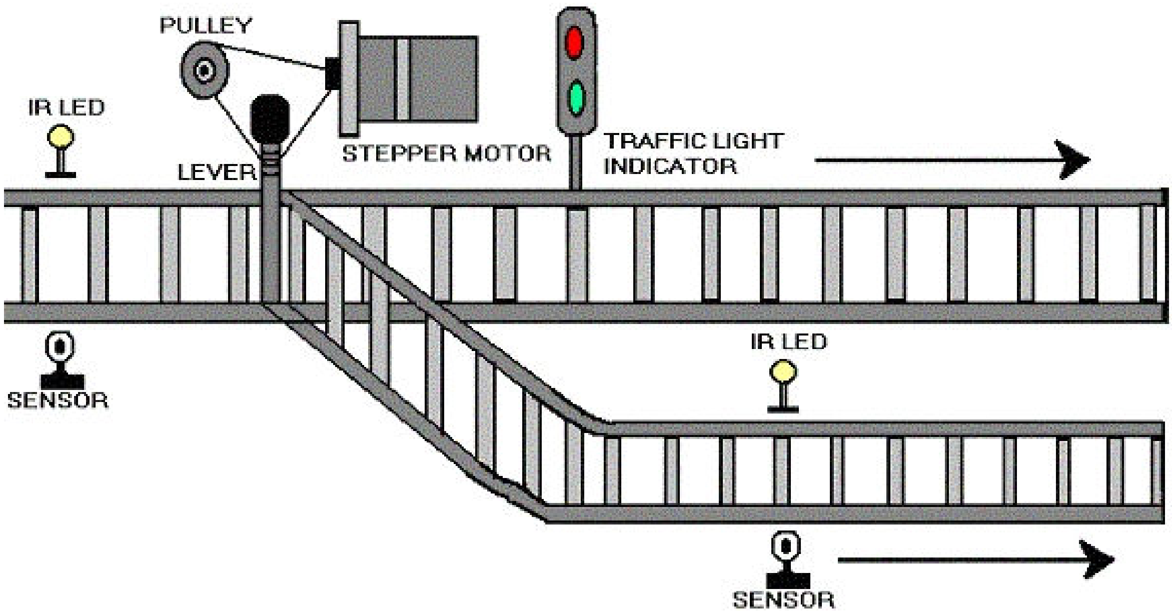
Gate Control:

Railways being the cheapest mode of transportation are preferred over all the other means .When we go through the daily newspapers we come across many railway accidents occurring at unmanned railway crossings. This is mainly due to the carelessness in manual operations or lack of workers. We, in this project has come up with a solution for the same. Using simple electronic components we have tried to automate the control of railway gates. As a train approaches the railway crossing from either side, the sensors placed at a certain distance from the gate detects the approaching train and accordingly controls the operation of the gate. Also an indicator light has been provided to alert the motorists about the approaching train.



Track Switching

Using the same principle as that for gate control, we have developed a concept of automatic track switching. Considering a situation wherein an express train and a local train are travelling in opposite directions on the same track; the express train is allowed to travel on the same track and the local train has to switch on to the other track. Indicator lights have been provided to avoid collisions .Here the switching operation is performed using a stepper motor. In practical purposes this can be achieved using electromagnets.



Hardware Description

The project consists of four main parts:

1. 8051 microcontroller
2. IR Transmitter
3. IR Receiver
4. Stepper Motor Circuit

8051 Microcontroller

The I/O ports of the 8051 are expanded by connecting it to an 8255 chip. The 8255 is programmed as a simple I/O port for connection with devices such as LEDs, stepper motors and sensors. More details of the 8255 are given later.

The following block diagram shows the various devices connected to the different ports of an 8255. The ports are each 8-bit and are named A, B and C. The individual ports of the 8255 can be programmed to be input or output, and can be changed dynamically. The control register is programmed in simple I/O mode with port A, port B and port C (upper) as output ports and port C (lower) as an input port.

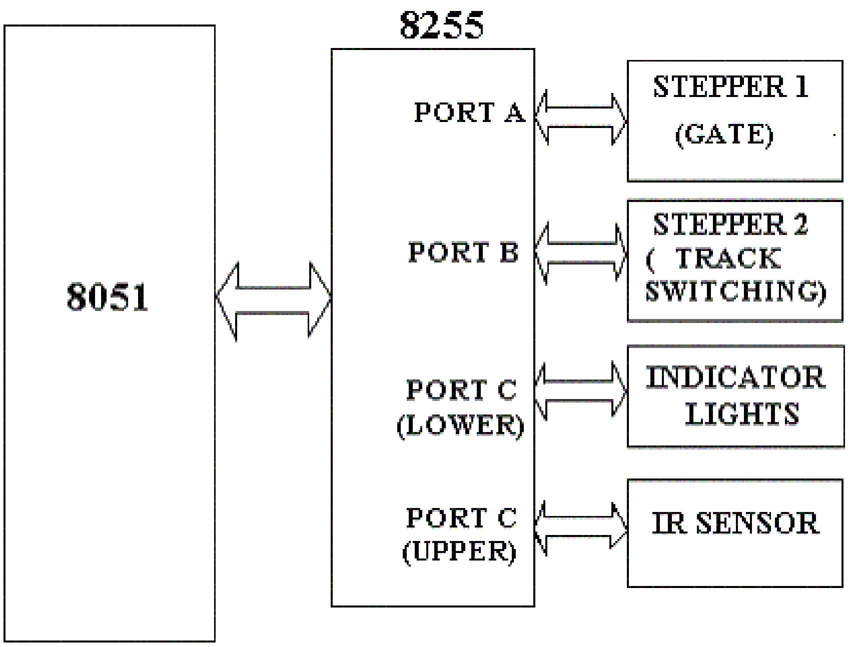


Fig: Block diagram of 8051 Microcontroller

IR Circuits

This circuit has two stages: a transmitter unit and a receiver unit. The transmitter unit consists of an infrared LED and its associated circuitry.

IR Transitter

The transmitter circuit consists of the following components:

1. IC 555
2. Resistors
3. Capacitors
4. IR LED

The IR LED emitting infrared light is put on in the transmitting unit. To generate IR signal, 555 IC based astable multivibrator is used. Infrared LED is driven through transistor BC 548.

IC 555 is used to construct an astable multivibrator which has two quasi-stable states. It generates a square wave of frequency 38kHz and amplitude 5Volts. It is required to switch 'ON' the IR LED.

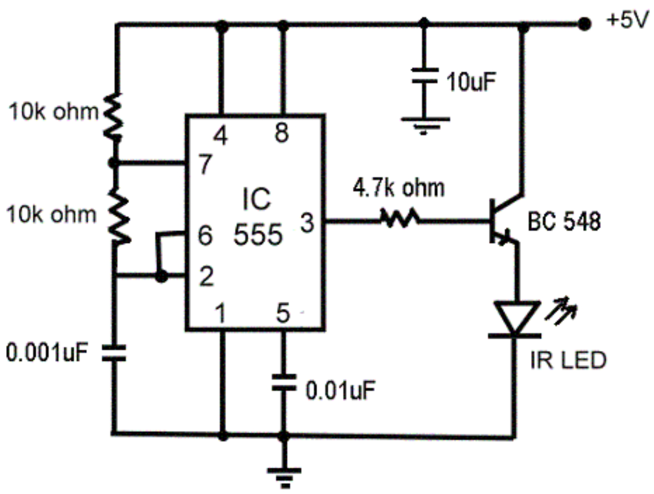


Fig: IR Transmitter

IR Receiver

The receiver circuit consists of the following components:

1. TSOP1738 (sensor)
2. IC 555
3. Resistors
4. Capacitors

The receiver unit consists of a sensor and its associated circuitry. In receiver section, the first part is a sensor, which detects IR pulses transmitted by IR-LED. Whenever a train crosses the sensor, the output of IR sensor momentarily transits through a low state. As a result the monostable is triggered and a short pulse is applied to the port pin of the 8051 microcontroller. On receiving a pulse from the sensor circuit, the controller activates the circuitry required for closing and opening of the gates and for track switching. The IR receiver circuit is shown in the figure below.

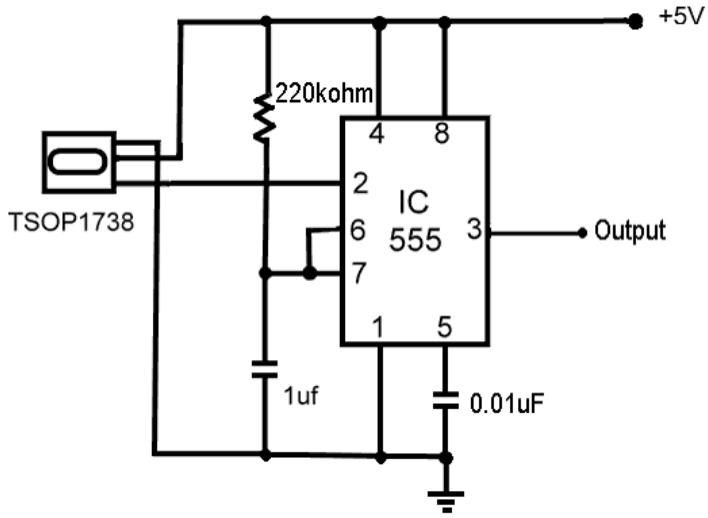


Fig: IR Receiver

Stepper motor circuit

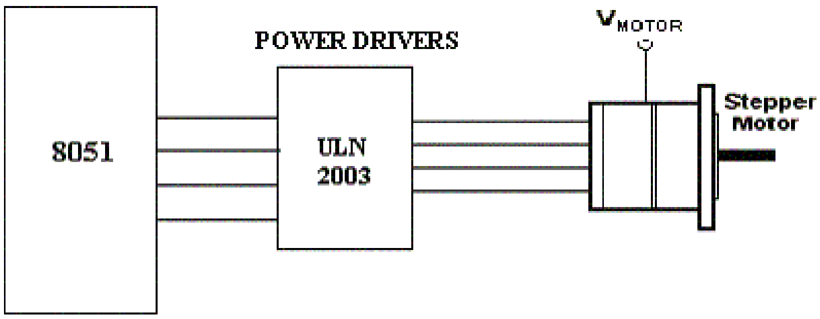


Fig: Stepper motor circuit

Here a stepper motor is used for controlling the gates. A stepper motor is a widely used device that translates electrical pulses into mechanical movement. They function as their name suggests – they "step" a little bit at a time. Steppers don't simply respond to a clock signal. They have several windings which need to be energized in the correct sequence before the motor's shaft will rotate. Reversing the order of the sequence will cause the motor to rotate the other way.

Stepper motor control board circuit:-

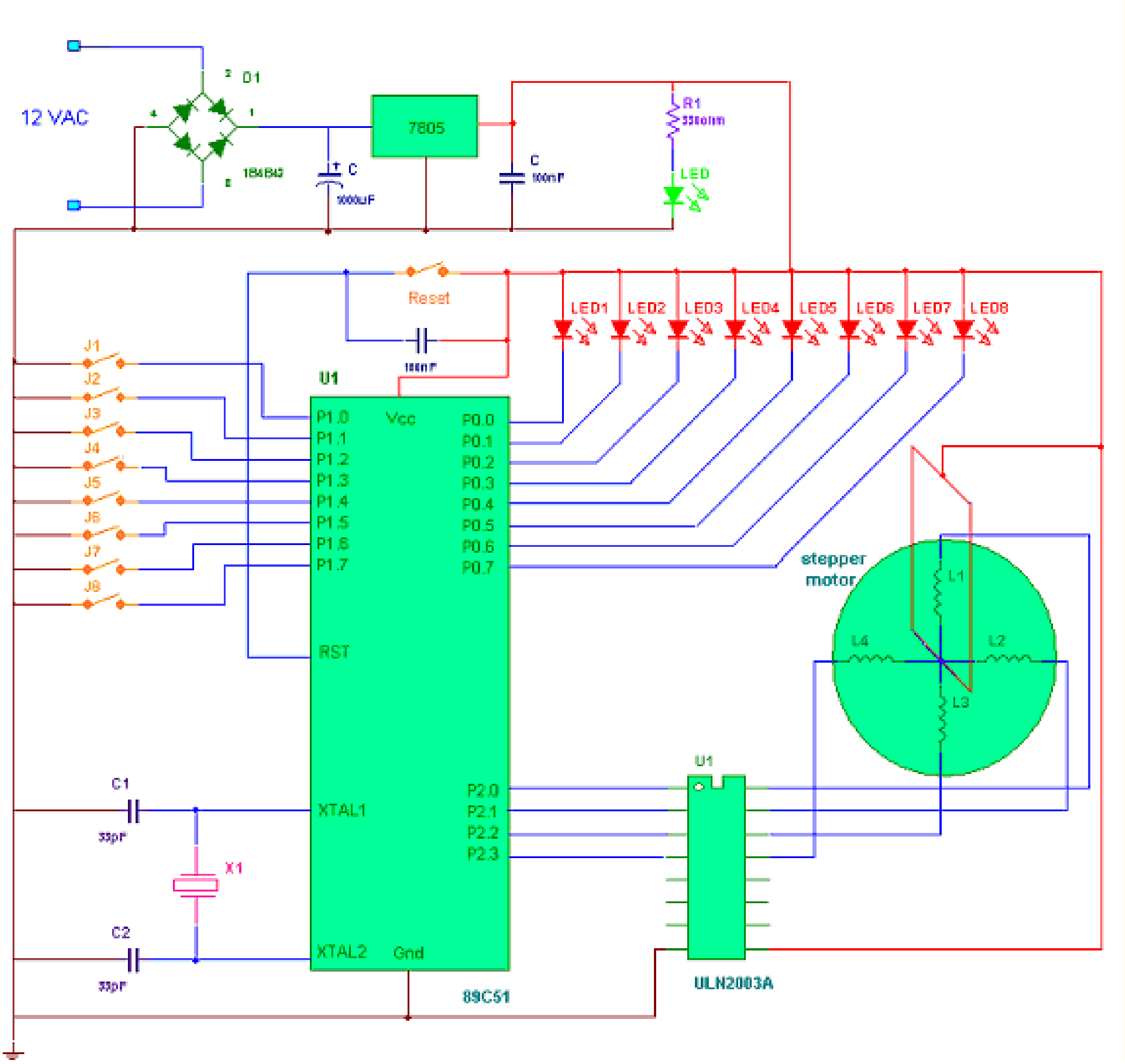


Fig: Stepper motor control board circuit

The circuit consists of very few components. The major components are 7805, 89C51 and ULN2003A.

Connections:-

1. The transformer terminals are given to bridge rectifier to generate rectified DC.

2. It is filtered and given to regulator IC 7805 to generate 5 V pure DC. LED indicates supply is ON.
3. All the push button micro switches J1 to J8 are connected with port P1 as shown to form serial keyboard.
4. 12 MHz crystal is connected to oscillator terminals of 89C51 with two biasing capacitors.
5. All the LEDs are connected to port P0 as shown
6. Port P2 drives stepper motor through current driver chip ULN2003A.
7. The common terminal of motor is connected to Vcc and rest all four terminals are connected to port P2 pins in sequence through ULN chip.

Programming for Microcontroller:

In this program we shall see minimal functionality. Only first two switches and first three LEDs will be used. When you press one switch the motor will start rotating in clockwise direction and stops when completes one revolution. Pressing second switch will do the same job but in anticlockwise direction. The speed will be 10 RPM fixed. Motor runs in single coil excitation mode. First LED indicates key press event. Second blinks when motor rotates clockwise and third blinks when motor rotates anticlockwise.

```
Org 00h
mov r0,#01h      ; initileze key count
mov p1,#0ffh      ; P1 as input port
lop:    mov a,p1
cjne a,#0ffh,jmp
ajmp lop            ; loop until any key
; is pressed
jmp:  clr p0.0              ; indicates keypress
loop:  rrc a
jnc num             ; get key no.
inc r0
sjmp loop
num: acall dely          ; key debounce delay
setb p0.0

cjne r0,#01h,nxt      ; for 1st key
acall clkwise            ; rotate motor
```

```
sjmp over                 ; clock wise
nxt:    cjne r0,#02h,over ; for 2nd key
acall aclkwise           ; rotate anticlock

over:  mov p2,#00h         ; restore initial
mov p1,#0ffh             ; data and
mov r0,#01h
sjmp lop                  ; jump to loop again

clkwise:
mov r1,#32h              ; load count 50d
go: clr  p0.1
mov p2,#01h             ;give 4 pulses in loop
acall delay              ; total 200 pulses
mov p2,#02h             ; each coil energized
acall delay              ; one by on in single
setb p0.1
mov p2,#04h             ; coil excitation
acall delay              ; 30 ms delay in betwn
mov p2,#08h             ; means speed is
acall delay              ; 10 RPM
djnz r1,go
ret
aclkwise:
mov r1,#32h
go1:clr p0.2
mov p2,#01h
acall delay
mov p2,#08h         ; revert the pulse
acall delay          ; sequence for
setb p0.2
mov p2,#04h         ; anticlockwise
acall delay
mov p2,#02h
acall delay
djnz r1,go1
ret

delay:
mov r5,#1Eh         ; load count 30
```

```
lop2: mov r6,#FAh     ; give 1 ms delay
lop1:nop              ; so it gives full
nop                   ; 30 ms delay
djnz r6,lop1
djnz r5,lop2
ret

dely:
mov r5,#0C8h      ; load count 200
lop2: mov r6,#0FAh   ; for 200ms delay
lop1: nop
nop
djnz r6,lop1
djnz r5,lop2
ret
end
```