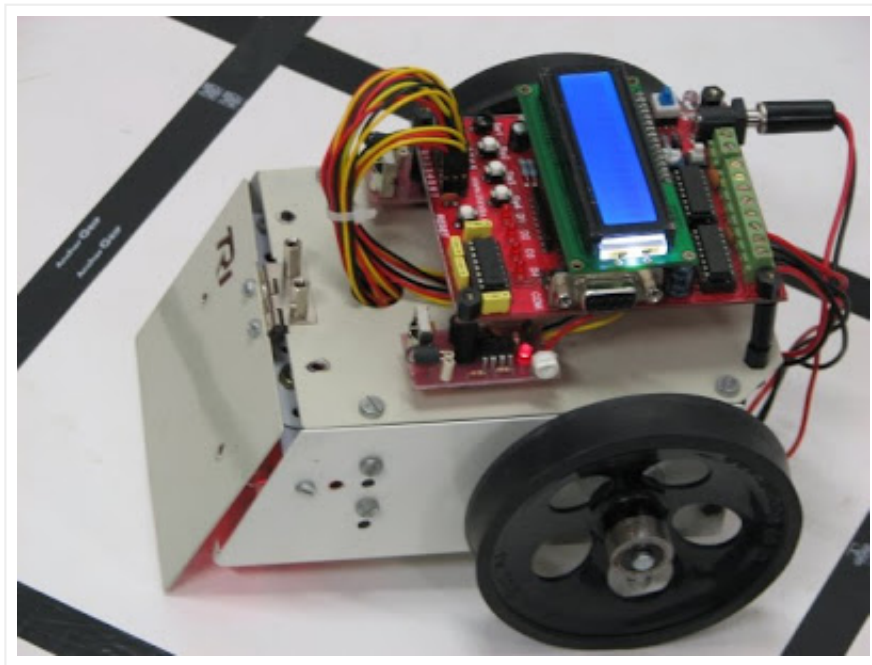


ROBOTICS news Robotic cool toys and robot making tutorials

Robotics news and the coolest robotic gadgets reviewed by a robot fan..

Friday, March 14, 2008

Build GRID Solving Robot tutorial

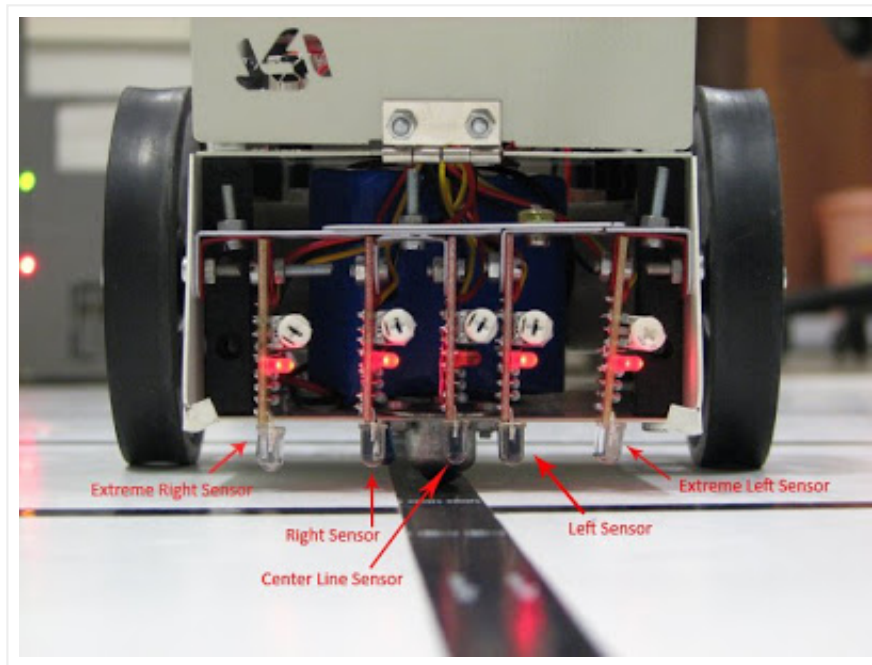


GRID solving **Robot** is useful for various robotics competitions. Here we have built a basic Grid solving robot using the iBOT by using 5 line sensors as shown in the image. The basic principle used for grid following is detecting the intersection and taking a turn accordingly. Here we have programmed the Robot for moving in a diagonal way from one end of the grid to another. As shown in the figure the five sensors connected of which the center one is for following the line and the sensors on the immediate left and right of the center sensor are for keeping the robot on the straight path. The extreme left and right sensors of robot are used for junction detection.

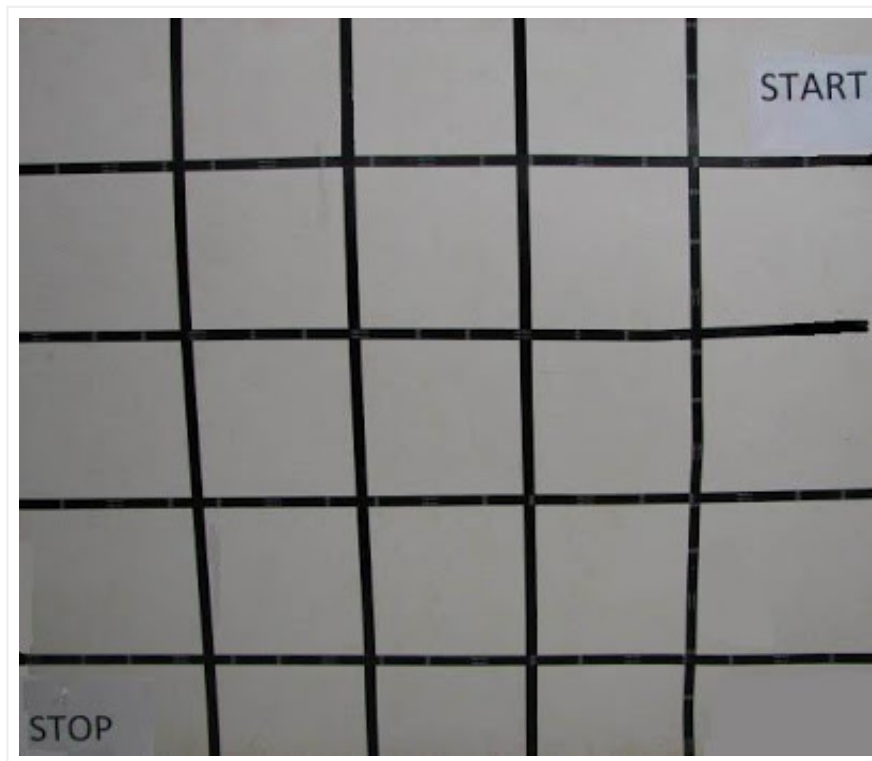
Here when a junction is detected robot takes a single wheel turn, thus aligning the robot on to the right angled path. Since we want the robot to travel diagonally we have programmed it to take alternate right and left turns at the junctions. A small amount of delay is included after each turn of robot to prevent false input to be taken by robot while turning. On experimenting with a lower speed (45rpm) motor the robot was more prone to such false signals and was not taking proper turns. Also the time taken for the robot to align back to the straight line was more because of which the robot missed to detect junction at some occasions. To avoid such situations while using low rpm motors the size of the Grid should be increased.

Here we have used 100rpm motors, which suited ideally for our program. Motors with higher rpm may cause problem as the junction can be missed due to speed.

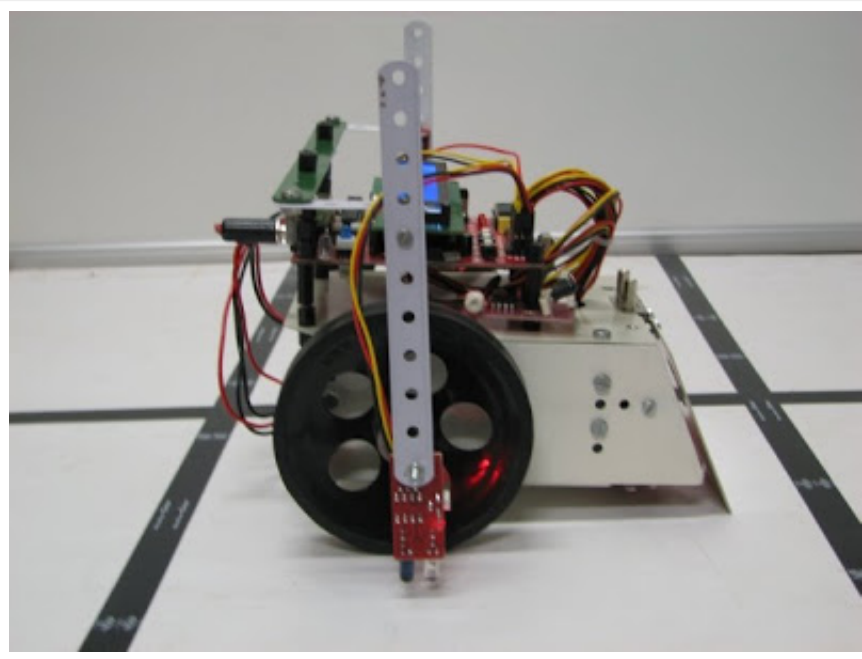
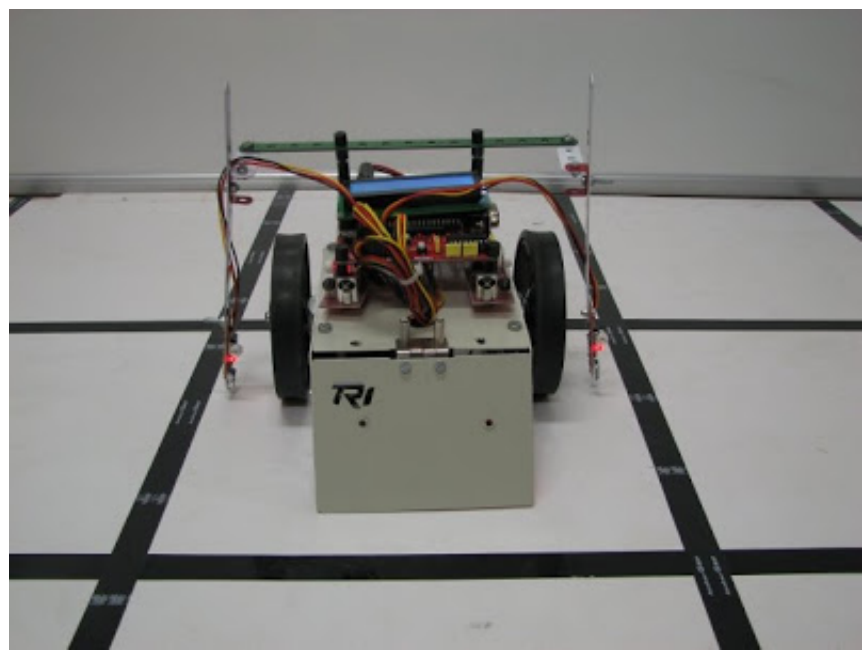
The sensors are connected as shown.



The GRID Arena is as shown. The size of each block is 10 inch x 10 inch.



We also experimented with various other combinations for sensor placements. The external left and right sensors can be aligned in the same line as the wheel. in such a case we need to take a differential turn. The placement of other three sensors will remain the same. The placement of the external sensors should be as shown below



PROGRAM FOR GRID FOLLOWER for Robot USING iBOT

/* Center sensor is connected at P1_4

Right sensor is connected at P1_1

Left sensor is connected at P1_2

Extreme Right sensor is connected at P1_0

Extreme Left sensor is connected at P1_3

Left Motor connected at P2_0 and P2_1

Right Motor connected at P2_2 and P2_3*/

```
#include
```

```
#include
```

```
#define ert P1_0
```

```
#define rt P1_1
```

```
#define lt P1_2
```

```
#define elt P1_3
```

```
#define cnt P1_4
```

```
#define fwd 0x0A
```

```
#define rev 0x05
```

```
#define lft 0x08
```

```
#define rht 0x02
```

```
void main()
```

```
{
```

```
int i=0;
```

```
P1=0xFF; //intialize P1 as input port
```

```
P2=0X00; //intialize P2 as output port
```

```
while(1)
```

```
{
```

```
if (cnt==0 && rt==1 && lt==1 && elt==1 && ert==1)
```

```
/*if center sensor is on black line and other sensors on white area*/
```

```
{
```

```
P2=fwd;
```

```
}
```

```
if (rt==0 && lt==1 && elt==1 && ert==1)
```

```
/*if right sensor is on black line and other sensors on white area continue to turn right till the  
center sensor is on black line*/
```

```
{
```

```
while(cnt==1)
```

```
{
```

```
P2=rht;
```

```
}
```

```
}
```

```
if (rt==1 && lt==0 && elt==1 && ert==1)
```

```
/*if left sensor is on black line and other sensors on white area continue to turn left till the  
center sensor is on black line*/
```

```
{
```

```
while(cnt==1)
```

```
{
```

```
P2=lft;
```

```
}
```

```
}
```

```
if (rt==0 && lt==0 && elt==0 && ert==1&& cnt==0)
```

```
/*if all other sensors except for extreme right sensor are on black line (i.e if the Bot is not  
aligned) align the iBOT such that all sensors come on black line ( i.e. move the Bot left)*/
```

```
{
```

```
while(ert==1)
```

```
{
```

```
P2=lft;
```

```
}
```

```
}
```

```
if (rt==0 && lt==0 && elt==1 && ert==0 && cnt==0)
```

```
/*if all other sensors except for extreme left sensor are on black line (i.e if the Bot is not aligned) align the iBOT such that all sensors come on black line ( i.e. move the Bot right)*/
```

```
{
```

```
while(elt==1)
```

```
{
```

```
P2=rht;
```

```
}
```

```
}
```

```
if (rt==0 && lt==0 && elt==0 && ert==0 && cnt==0)
```

```
/*if all sensors are on black i.e a junction is reached
```

```
increment a variable 'i' too determine the turning direction*/
```

```
{
```

```
i++;
```

```
/*if 'i' is even turn left until center sensor senses black line and other sensors sense white background*/
```

```
if (i%2==0)
```

```
{
```

```
if (elt==1 && ert==1 && cnt==0 && rt==1 && lt==1)
```

```
{
```

```
P2=fwd;
```

```
}
```

```
else
```

```
{
```

```
P2=left;
```

```
DELAY(500);
```

```
}
```

```
}
```

```
if (i%2!=0)
```

```
/*if 'i' is odd turn right until center sensor senses black line and other sensors sense white background*/
```

```
{
```

```
if (elt==1 && ert==1 && cnt==0 && rt==1 && lt==1)
```

```
{
```

```
P2=fwd;
```

```
}
```

```
else
```

```
{
```

```
P2=rht;
```

```
DELAY(500);
```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

NSensation at 5:03 AM



2 comments:



comp-tips March 20, 2009 at 11:40 AM

nice
thanks

[Reply](#)



darwin August 19, 2013 at 3:50 AM

thank u very much

i had some other doubts about it i once want to contact with you

[Reply](#)

Enter your comment...

Comment as: Google Accour ▼

[Publish](#)

[Preview](#)



[Home](#)



[View web version](#)

Powered by [Blogger](#).