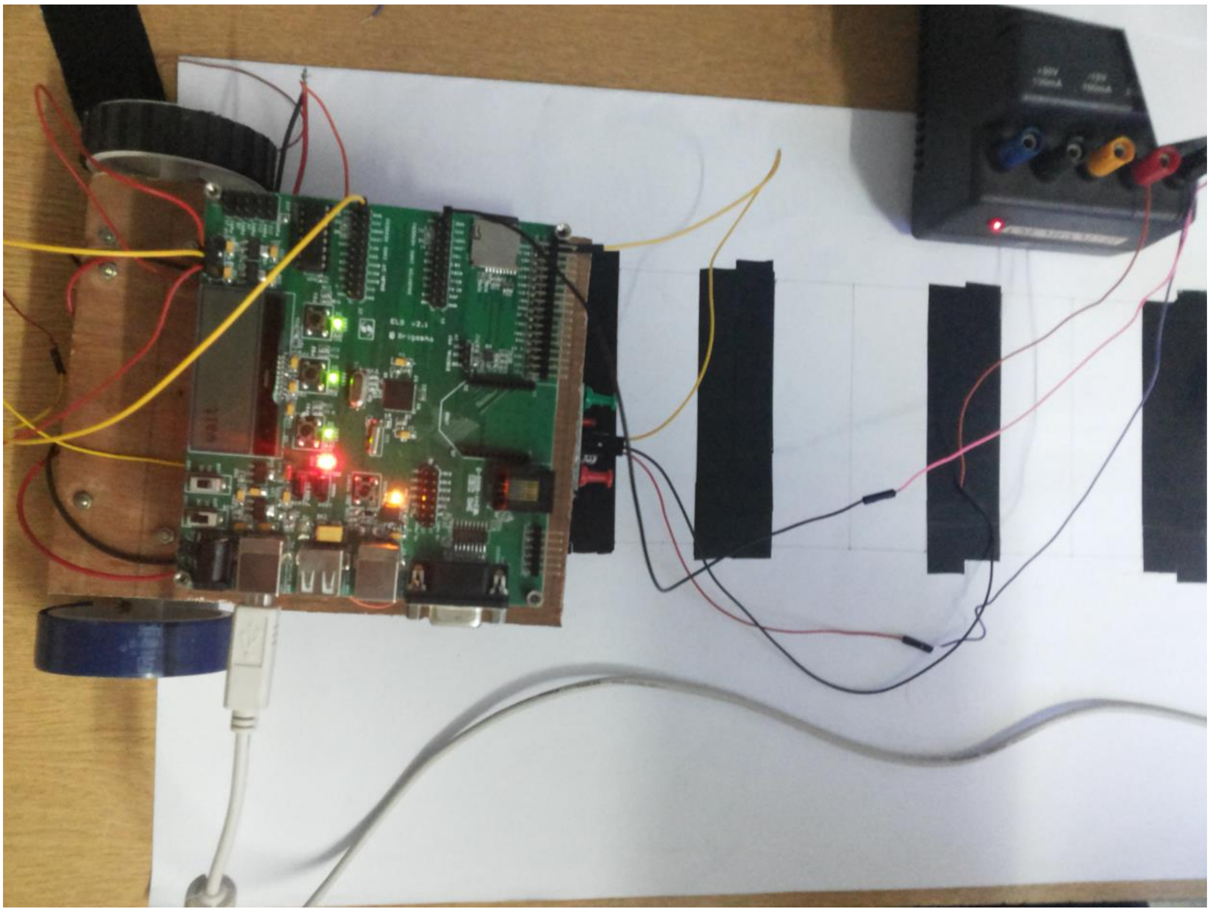
Barcode Scanner



Group 18:

Chandragiri Vinay	(120101018)
Kazipeta Sharath	(120101032)
Murali Raghu Babu B	(120101043)
Ravuru Anil Kumar	(120101059)

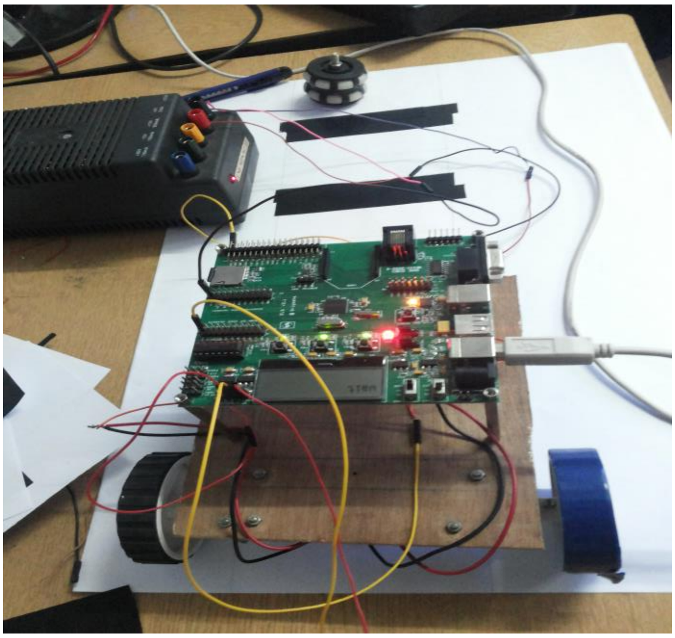
Index:

Introduction:

A barcode is an optical machine-readable representation of data, which shows certain data on certain products. A barcode consists of 2 parts – a human readable message, a machine readable mapping of this message. The mapping between messages and barcodes is called a symbology. The specification of a symbology includes the encoding of the single digits/characters of the message as well as the start and stop markers into bars and space, as well as the computation of a checksum.

Summary:

We are constructing a 1-d barcode scanner using an IR LED pair (consisting of IR emitter and IR Detector), which would enable us to scan the whole cross-section of Barcode sequentially. For the construction of the barcode scanner we have used a BOT which is indeed controlled by the PIC Controller.

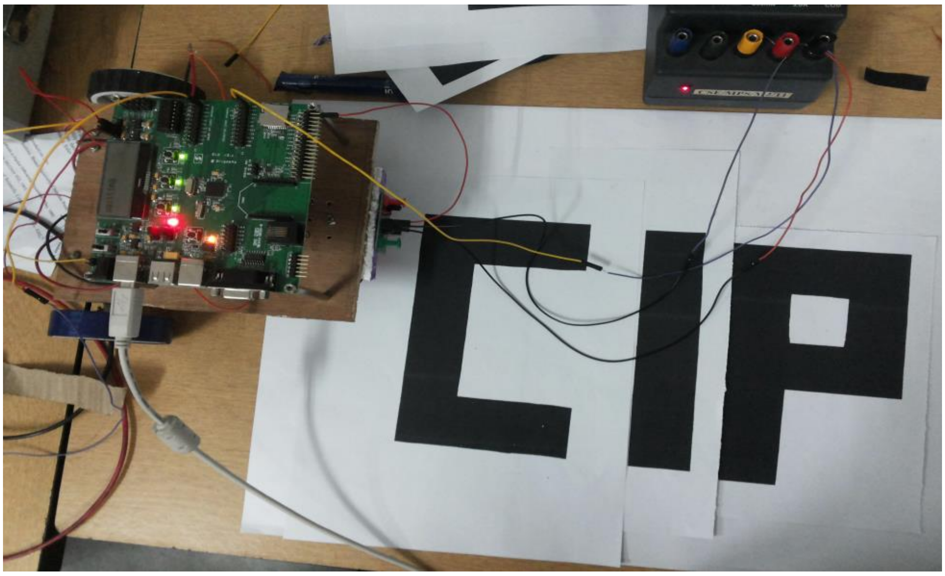


BAR CODE SCANNER scans the 1’s (Black bars) and 0’s (White Bars). The Barcode Scanner works in an edge triggered fashion. It distinguishes the Blacks and White’s Border maintains the trigger values for each length of the Bar and then decodes them and converts them into 1’s and 0’s. It senses the edge of Black and White and hence works through that idea. Even if you increase the speed of the motor, it shall distinguish the edge between black and white and hence here the barcode scanner is independent of the speed of the motor.

High Level Design:

Project Idea Source and Rationale:-

Our original goal was to come up with a final project that utilizes most, if not all the knowledge we have accumulated from other projects throughout the semester, and of course, it has to be doable with our current level of technical expertise. We naturally wanted to do something with barcode scanner hardware. Our idea was a construction of a bot which has an IR LED pair attached to it. It is indeed controlled by PIC controller and the information received by the IR LED pair is send to PIC to analyse the data. The Data received is checked whether the barcode is read from the front direction or the reverse direction. Even the received data is checked of any Error and if any Error has occurred it corrects the bits of the barcode. As a Part of Extension we have built a mechanism to recognise/ distinguish the characters of the ENGLISH language.



The BAR CODE SCANNER will be able to distinguish the characters C’, ‘I’, ‘P’ shown in the figure.

Project Background:

There are many types of commercial barcode standard but instead of 95 bits we implemented an 11 bit Barcode. This is done just to demonstrate. We have working code for 95 bits. The Description of Barcode and how we check for error has been implemented as follows.

Each Barcode contains 95 bars which are divided into 4 parts

1. Left band (101)
2.Center band (101)
3.Right band (01010)
4.12 sets of bands each containing 7 bars/bits. (6 on either sides of centre band)

Each band on left side starts with 1 and ends with 0 and contains odd number of 1's and right bands starts with 0 and ends with 1 and contains even number of 1's.

And the main feature of any barcode is it should output same results when it is scanned from opposite direction. The feature has been implemented in our project. It shall compare the first set of bits scanned to the set of barcodes (which are allowed to appear left) and the other way round and hence obtain the direction of scan of the barcode. After knowing the direction in which barcode should scan we check for error detection. We have some possible possibilities for a left band (since only odd number of 1's). If the barcode scans one error bit (0 instead of 1 or vice-versa), we subtract its decimal number with decimal values of possible bands. If it comes out to be a perfect power of 2 the band with error reading is replaced with subtracted band.

Our demo project contains 11 bits with width 3.6cm (length of 2 tapes). Left band (10), Right band (01), centre band (0) and a band of 3 bits on each side of centre band with similar properties of original barcode(parity, decoding). The Barcode has in-built correction method which corrects the barcode scanned based on the parity bit.

Logical Structure:

Essentially, the majority of this project focused on the conversion from the actual barcode strings to a series of binary data representing the thick/thin stripes/spaces of each particular barcode character. The barcode can be of any width (min threshold width is about 3cm). Once we have attained a reliable method of conversion between the actual barcodes and the barcode binary width data, we implemented parsing and decoding barcode binary width data, comparison of scanned code with pre-defined inventory barcodes and outputting the scanned result onto the LCD screen and display on the SD card. For the character Recognition we have scanned the input twice (one in the top and other in the bottom) and depending on the Trigger values for each character we'll distinguish the characters and display the sequence of characters scanned on the LCD screen and display on the SD card.

Program/Hardware Design:

Program Details:

After getting first batch of ADC data from the PIC, after our first barcode scan pass, we had some idea as to how we should go about interpreting these ADC data points. Even before we took a look at the data points, we knew that black stripe on the barcode string would yield a low ADC values and white space on the barcode string would yield a high ADC value since black surface absorbs more and reflect less IR than does the white surface.

While the data points we took confirmed our presumption of low ADC for black surface and high ADC for white surface, the resulting ADC graph was not what we were hoping for. Instead of an even amplitude oscillation between the a nearly constant high ADC value and a nearly constant low ADC value, we got a graph that has peaks and valleys at several different ADC values, a most undesirable outcome that threw us off of our initial plan using a threshold based binary assignment (With a clean threshold, we can convert any ADC value above the threshold as 1 and anything under as 0, where 0 and 1 represent black and white). It turned out that our IR detector's active scan area is much bigger than the thin stripe/space of our sample barcode and it was therefore taking the average of the neighbouring black and white stripe/space’s reflectance at any one time as its output (therefore, if we are at a thin black stripe neighbouring by thick white space, the ADC graphical conversion would yield a shallower valley than if we were scanning a thick black stripe neighbouring by thin white space).

The first side effect of up our detector’s active scanning area is a much smaller ADC difference between the black and white conversions. Then feed it to A/D conversion pin (PORTA.0). The second side effect is the shifting of the overall ADC conversion range depending on the effect of the adjacent Black/White Bars, due to which we lost most of the peak values i.e. If IR is in Black even then it shows a higher voltage because of the adjacent bars. So is the reason we are limiting the minimum thickness of the each bar width of the barcode. It clearly works beyond the limit width.

Infrared Sensor – IR SENSOR

Hardware Details:

Our barcode scanner consists of the following components:

i) PIC Controller

ii) DC motors (2)

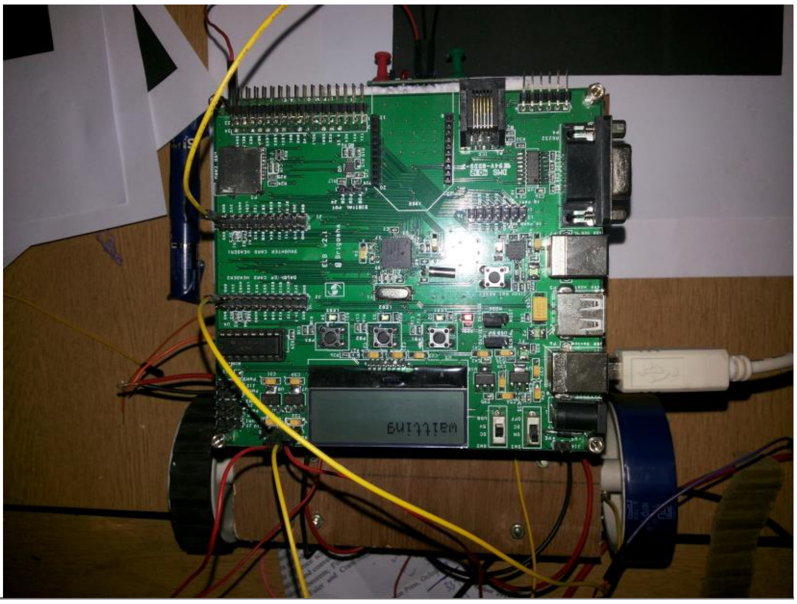
iii) IR LED (1)

iv) Cardboard (0.5 sq. ft.)

v) DC supply of 5v and 3v

vi) JUMP wires

vii) SD Card



Results:

Execution:

The basic operation of our barcode scanner involves the following steps:

1. Load the program to the Pic using the PIC kit-3.

2. On pressing the push button, the BOT starts going forward and scans the received analog signal is sent to PIC for ADC conversion.

3. After complete reading of the barcode the Bot completely stops and processes the data and displays if any error has occurred and copies it to the SD card.

For the character distinction part we need to execute it differently Load the program to the Pic using the PIC kit-3.

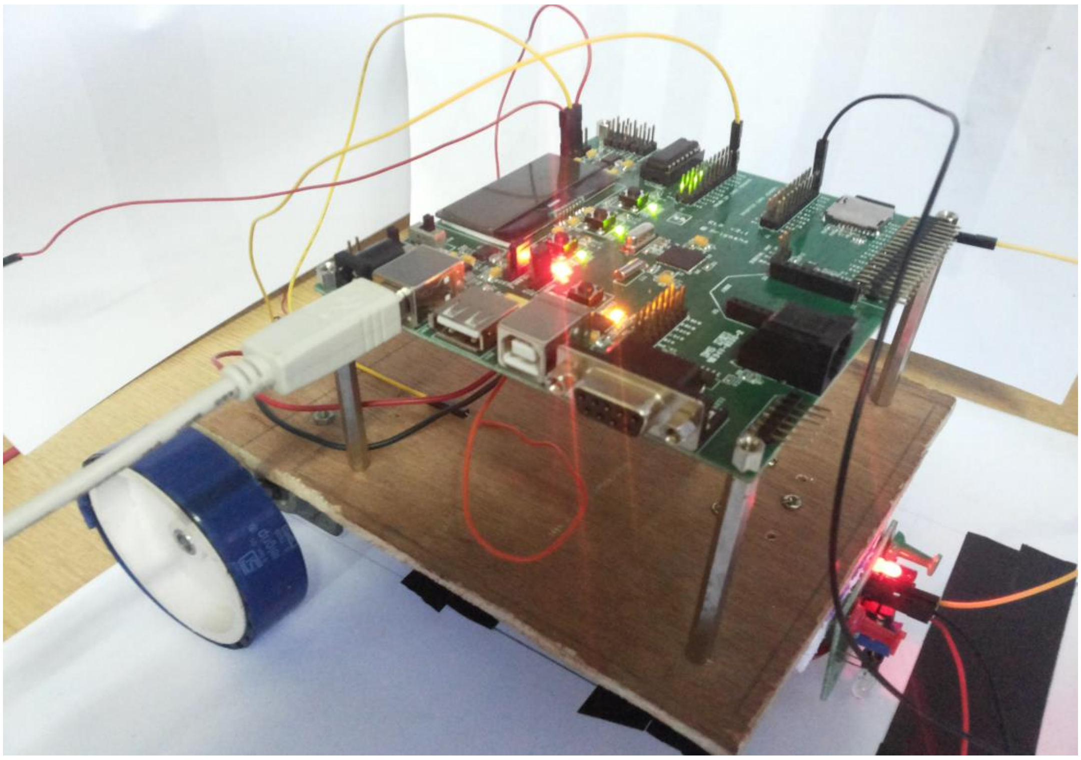
→On pressing the push button, the BOT starts going forward and scans and received analog signal is sent to PIC for ADC conversion

→After complete reading of the barcode the Bot stops.

→Now the bot is to be placed on the bottom side of the characters and then on clicking the push button it again starts reading.

→After complete reading of the barcode the Bot completely stops, and processes the data and displays if any error has occurred and copies it to the SD card.

Overall, the operation of our barcode scanner is very simple, however, it does require some practice before a first time operator can perform successful scans consistently.



Accuracy:

While our barcode scanner cannot scan a regular size, it puts a limit on the width of each barcode (around 2.5cm for each). Beyond that width it accurately works for varying lengths.

Safety in our design:

Our barcode scanner's user interface involves only 2 buttons and the wooden scanning apparatus. If setup correctly, our barcode scanner does not expose its user to any electrical connection at any time and therefore is very safe to operate.

Usability:

While a first timer user may not be able to achieve successful scans right away on our machine. Anyone with rudimentary hand-eye coordination and some scanning practice should be able to scan any of our specialized barcode samples correctly on our barcode scanner.

Applications:

Apart from the basic uses of the barcode we'll see the uses of the barcode scanner developed by us.

1. Width of each bar in Barcode can vary, even though our scanner works fine.

2. It can check if any error has occurred. If it has occurred then if can perform error correction.

3. Barcode scanner can also be used for the character distinction.

4. It can be extended to scan any length barcodes.

5. It can be used to scan letters on posters etc. and all that data is stored onto the SD Card.

6. Also the Barcode.txt file into which the code has been written can be connected to the computer and on running the python module written, it shall automatically do the google search of the barcode in Mozilla Firefox browser and display the results obtained.