# Search And Assist Robot Prototype

*By,*
*Abhinav M. (National Institute of Technology Trichy)*
*Siddhartha Kumar (National Institute of Technology Karnataka)*

*Under the Guidance of*
*Professor Bharadwaj Amrutur*

Electrical Communication Engineering
Indian Institute of Science
Bangalore - 560 012
July 2012

# Acknowledgement

I would first like to thank Professor Bharadwaj Amrutur for giving me this opportunity to work under his guidance and support

I would also like to extend my gratitude to Mr. Anand S. and Mr. Nagaraj Naik for their help throughout the project

I also would like to thank Mr. Abhinav M. for his help with this project

# Introduction:

Natural disasters and man-made disasters lead to a lot of loss of life, destruction of property and it is the first responders who are responsible to enter these disaster zones, rescue the survivors and salvage any important artifacts documents that might buried in the rubble. But a disaster of a large scale can lead to many hazards, for example a leaking gas pipeline, weakened structures, fire hazards, live power lines, etc. and all of these can be hidden and lead to an unnecessary loss of life to the rescue workers. That is why it has been the goal for many institutions and researchers to assist these rescue workers with technology that is already available all around us.

# Objectives:

The main aim of our project is to build a prototype of a robot that can be sent in to a disaster zone before any humans are allowed to enter, identify all hazards and map out safe routes along the way to be used by rescue workers. To be able to achieve this our robot must be mobile, send back a live video feed so the rescuers can visualize the entire zone, must be able to run simple image processing and signal conditioning algorithms so as to reduce the amount of data sent back and also allow for a semi autonomous control.

# Development:

The robot is powered by a beagleboard chosen specifically for its open source community support and its capabilities to perform complex image and audio processing. We have utilized the beagleboard to interface with the brushed DC motors via motor drivers to drive the robot, and control two servo motors that make up the camera mount, allowing for 360 degree field of vision. The beagleboard is also directly interfaced with a camera that streams a live feed back to the host computer via wifi, image processing can be performed on this feed on the beagleboard itself to identify markers, survivors etc.

The host PC sets up an "ad-hoc" network that the wifi dongle on the beagleboard connects to. The data collected by the robot is sent to the host PC via this wireless network. Also to control the robot the commands are sent from the host system to the robot via the same network.

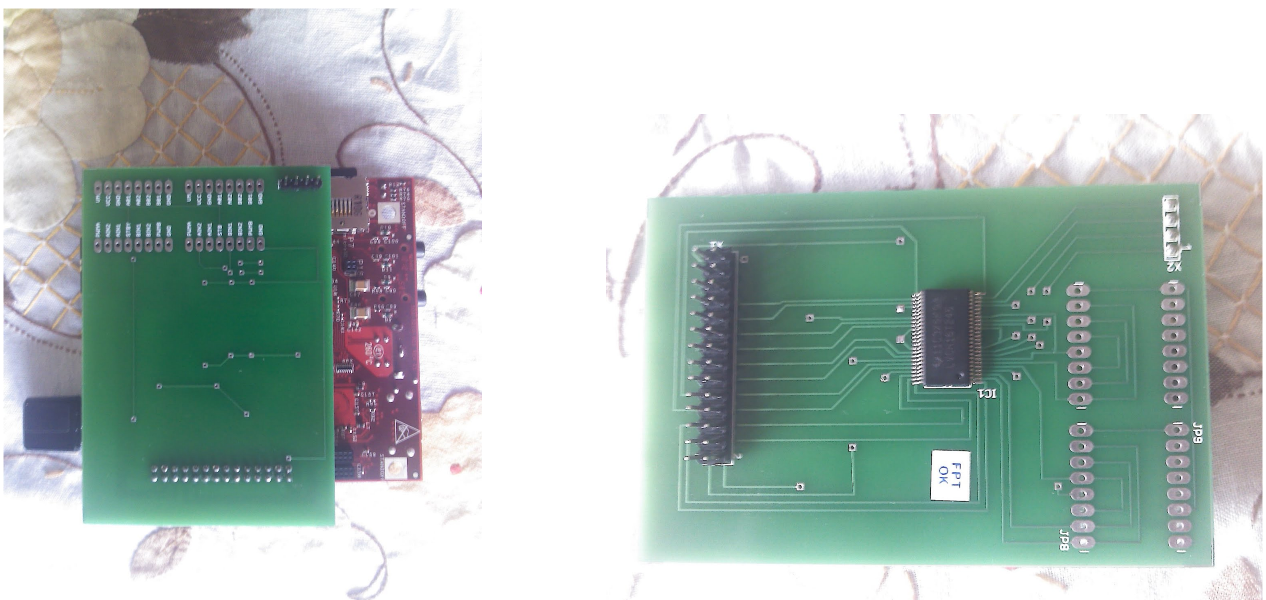
An extra provision has been made to allow a simple serial communication with the TIUSB3410 chip via the USB 2.0 interface provided on the beagleboard, this allows easy expandability of the entire platform to incorporate TI's low power MSP430 microcontroller based platforms which can act either as wireless dongles supporting PAN networks like zigbee and bluetooth, or else analog to digital converters that can take inputs from various sensors depending on the types of hazards expected.

# Hardware:

Due to the 1.8 V output of the general purpose input output drivers for the beagleboard we had to design a printed circuit board that had a logic level translator (SN74LVCH16T245) that could convert the 1.8 V output to 5 V output so as to be able drive the TTL logic motor drivers and the Servo motors.

The Motor Driver (TB6612FNG) takes three inputs for each motor a PWM signal, input 1 and input 2. The Input 1 and 2 control the direction of the motors and the PWM signal controls the speed of the Motor. The PWM signal is optional if no speed control is required. These DC motors will be the basic drive system for the robot. Movement will be controlled by the polarity of the DC motors, and rotation will be controlled by opposite rotation of either side, allowing for an almost zero turning radius.

The servo motors are controlled directly by a PWM signal, depending on the duty cycle of the PWM we  can control the angle of rotation. Therefore by cumulatively adding or subtracting "delta" from the duty cycle we can adjust the inclination and angle of the camera to get our required view.



The PCB placed on the beagleboard and seperately.

# Software:

## Beagleboard:

On the beagleboard we will have two processes running simultaneously, the camera gstreamer pipeline, on which image processing can be performed, which is being streamed across the wireless network constantly and a server waiting for a request from the host PC.

Once the host PC makes a connection with the beagleboard, the beagleboard sends an acknowledgement back to the host system. On receiving any input from the host the beagleboard identifies the tag (the first letter of the command) and sends the data to the corresponding function.

There are three main functions:

1. DC motor control: Which initializes all GPIOs, controls direction of each motor. Speed control has also been provided via a driver named “servodriver” that drives a normal GPIO pin with a PWM signal by simply toggling it with a timer.

2. Servomotor control: Keeps a cumulative count of the current duty cycle and increments or decrements depending on the input passed to it hence controlling the field of view of the camera.

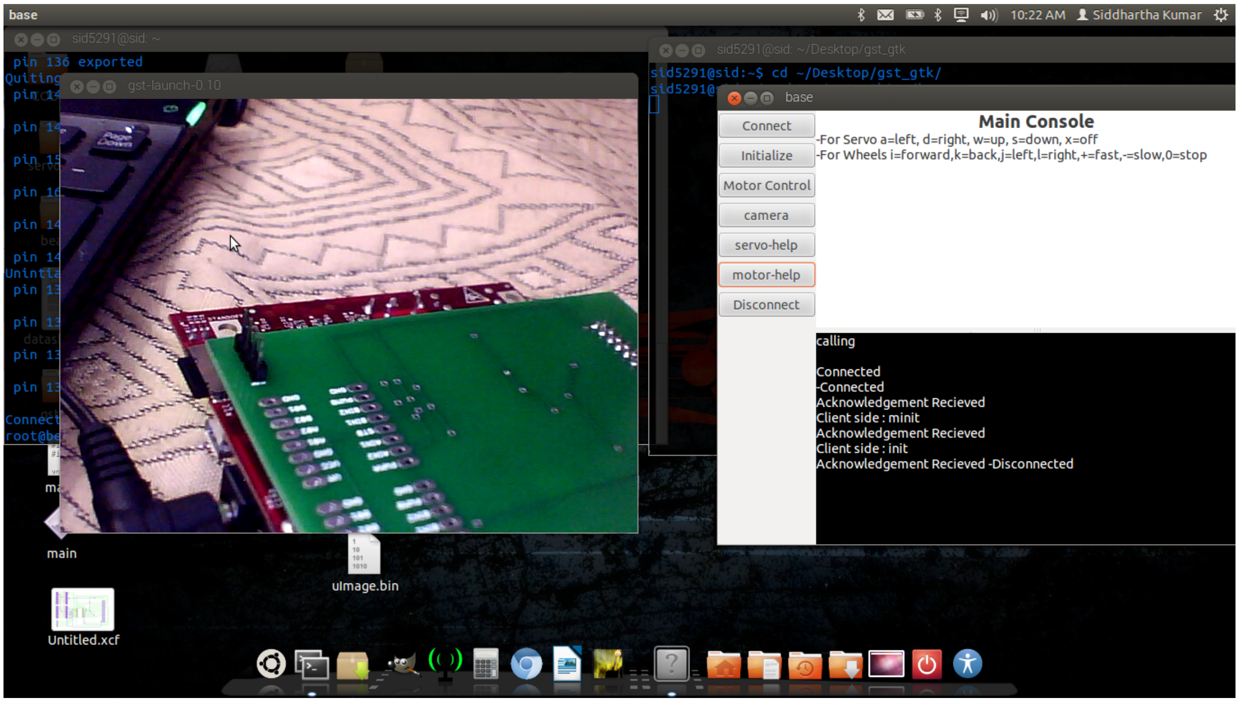
3.Serial Communication: This is a simple half duplex UART communication to with TIUSB3410 chip which in our case talks to the MSP430G2553, on the launchpad. This function either can constantly read ADC values sent from the microcontroller and send it back to the host PC or it can communicate with the microcontroller to get any specific data required.

## Host System:

The host system runs a crude graphical user interface which allows for all the basic tasks to be accomplished easily and with no learning curve. It runs the client side of both the gstreamer pipeline showing the output on a simple window and a function that takes in keyboard inputs to control the direction of motion and the camera angels.

By converting simple one-key keystrokes from the user to tagged strings which are sent to the beagleboard and appropriately executed it is very easy for the user to carry out the basic tasks therefore not requiring any special training. Also this system allows easy expandability for new functions to be applied in the background.

(All software is available at https://github.com/sid5291/search-assist-robot-iisc (git clone for all updates) )




Screenshot of the GUI ( The video feed on the left and the console on the right)

# Conclusion and Looking Forward:

With this project we have laid out the basic software and hardware framework for the first prototype of the robot. The robot is currently a data collecting system which is controlled remotely.

The next steps in this project is to assemble the complete robot and do some basic field tests . Then look into semi autonomous control and incorporation of the IMU device allowing for mapping functionality. Also on the host system the graphical user interface could be more polished, more intuitive and better designed.