

eYSIP2016

# EXPLORING WICED SENSE



Chinmay Patil  
Imran Khan  
Santosh  
Uttam

Duration of Internship: 10/06/2016 – 24/07/2016

*2016, e-Yantra Publication*

# Exploring Wiced Sense

## 1.1 Abstract

With hundreds of new ideas for applications of wireless devices it is critical that developers are provided a platform to quickly test their concepts and bring them to real use.

This project aims at exploring different aspects of WICED Sense (SIP Module), in order to start experimenting with sensor technology which will enable the developers to include physical quantities such as temperature , atmospheric pressure ,humidity ,direction, acceleration, yaw, pitch, roll, etc.

The aspects include understanding the wiced sense sdk inorder to program the device for user specific application , procedure to acquire the data from the wiced tag , saving the acquired data in a online database so that it can be accessible for distant users for processing , visualization of the sensor data on website , making automation related applicatons and path mapping of firebird V robot on a webpage using the sensor data.

The ultimate goal of the project is to create a basic platform inorder to enable future developers exploit the sensors in wiced sense.



## 1.2 Overview of tasks

- Downloading the SDK for WICED Sense, and getting familiar with it.
- Exploring different methods through which the data from wiced sense can be acquired and selection of one method for further project, taking into consideration all the aspects of project.
- Pushing the obtained data on desktop to the an online database.
- Creating GUI to display the obtained sensor data on the webpage.
- Developing a notification system so as to alert the user of a specific event on the webpage.
- Using the sensor data to perform path mapping of the firebird robot.
- Creating tutorials and documentation.

# Contents

<b>1</b>	<b>Exploring Wiced Sense</b>	<b>1</b>
1.1	Abstract . . . . .	1
1.2	Overview of tasks . . . . .	2
1.3	Hardware Components . . . . .	4
1.4	Software Components . . . . .	6
1.5	Data Flowchart . . . . .	9
1.6	Webpage Description . . . . .	11
1.6.1	Sensor GUI page . . . . .	11
1.6.2	Temperatutre notifier page . . . . .	12
1.6.3	Bot mapping page . . . . .	13
1.7	Future Work . . . . .	14
1.8	Bug report and Challenges . . . . .	14



## 1.3 Hardware Components

- Wiced Sense SIP Module



- WICED Sense is a BLE device manufactured by Broadcom\* which can provide wireless connectivity to wide range of embedded applications.
- The WICED Sense TAG is made up of the BCM20737S Bluetooth Low Energy SoC and five ST Microelectronics sensors: gyroscope, accelerometer, magnetometer, pressure, humidity and temperature. The BCM20737S connects directly to the sensors without the need for an external microprocessor. (\* 5 July, 2016 onwards Broadcom's IoT business is acquired by Cypress.)
- ST Microelectronics Devices used in the WICED Smart Kit:
  - \* Gyroscope (L3GD20)
  - \* Accelerometer (LIS3DSH)
  - \* Magnetometer (LSM303D)
  - \* Pressure sensor (LPS25H)
  - \* Humidity and Temperature sensor (HTS221)

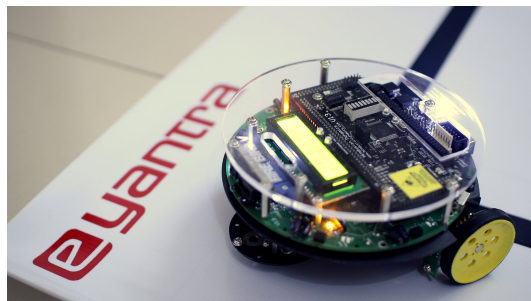
[Wiced Sense Vendor Link](#)  
[Wiced Sense Introduction](#)

### 1.3. HARDWARE COMPONENTS

---



- Zigbee Module
  - ZigBee is an IEEE 802.15.4-based specification for a suite of high-level communication protocols used to create personal area networks with small, low-power digital radios.
  - The technology defined by the ZigBee specification is intended to be simpler and less expensive than other wireless personal area networks (WPANs). Applications include wireless light switches, traffic management systems, and other consumer and industrial equipment that requires short-range low-rate wireless data transfer.
  - Its low power consumption limits transmission distances to 10100 meters line-of-sight. ZigBee is typically used in low data rate applications that require long battery life and secure networking (ZigBee networks are secured by 128 bit symmetric encryption keys.)
- Firebird V ( Atmega 2560 )



## 1.4 Software Components

- Linux Environment

- \* Node Js



- Node.js is a platform built on Chrome's JavaScript runtime for easily building fast and scalable network applications. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices.
- An important thing to realize is that Node is not a web-server. By itself it doesn't do anything. It doesn't work like Apache.

- \* Cylon Js



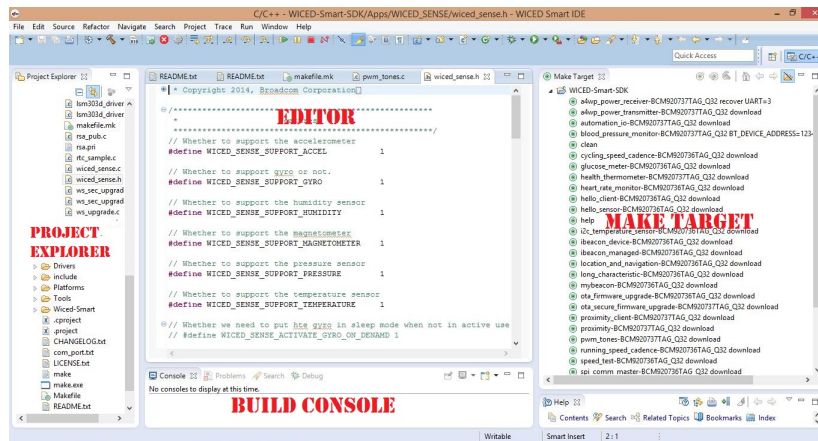
# CYLON.JS

- \* Cylon.js is a JavaScript framework for robotics, physical computing, and the Internet of Things. It makes it incredibly easy to command robots and devices.
    - \* Cylon.js has an extensible system for connecting to hardware devices. Compatibility of the hardware can be found at <https://cylonjs.com/>
    - \* Cylon js supports wiced sense. It has support to BLE (Bluetooth Low Energy) devices.
  - [Installation Procedure](#) for the softwares.



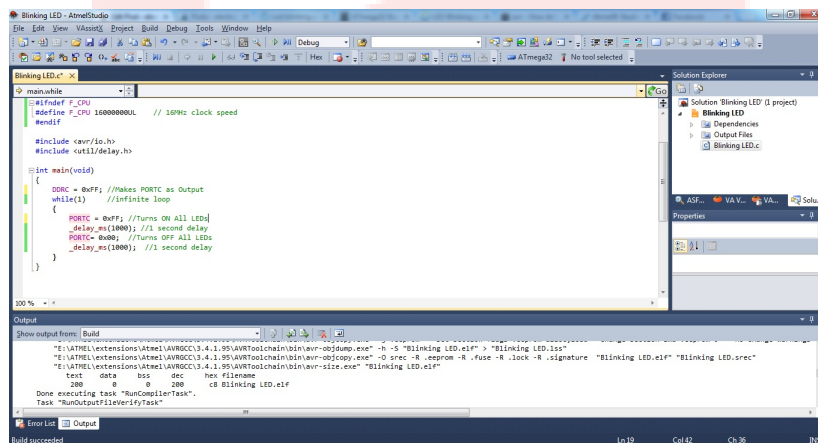
## 1.4. SOFTWARE COMPONENTS

- Windows Environment
  - Wiced Sense SDK



It is used to program the Wiced Sense Kit.

- Atmel Studio



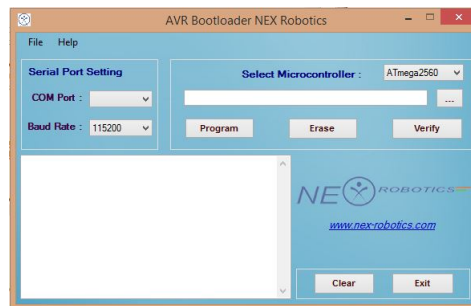
It is used to program Firebird V robot.





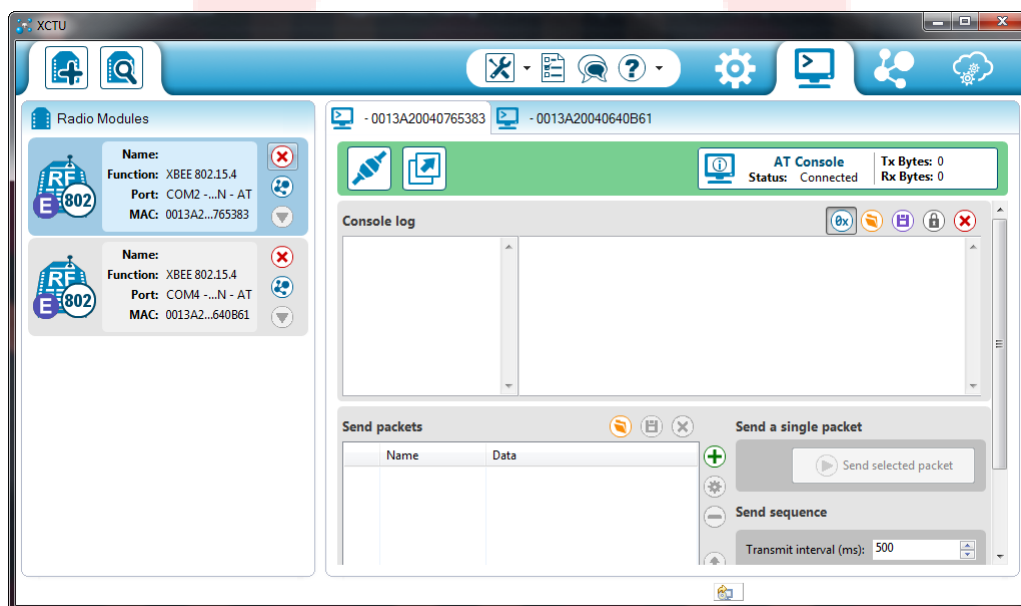
## 1.4. SOFTWARE COMPONENTS

### – AVR Bootloader



It is used to burn code into Firebird V.

### – XCTU

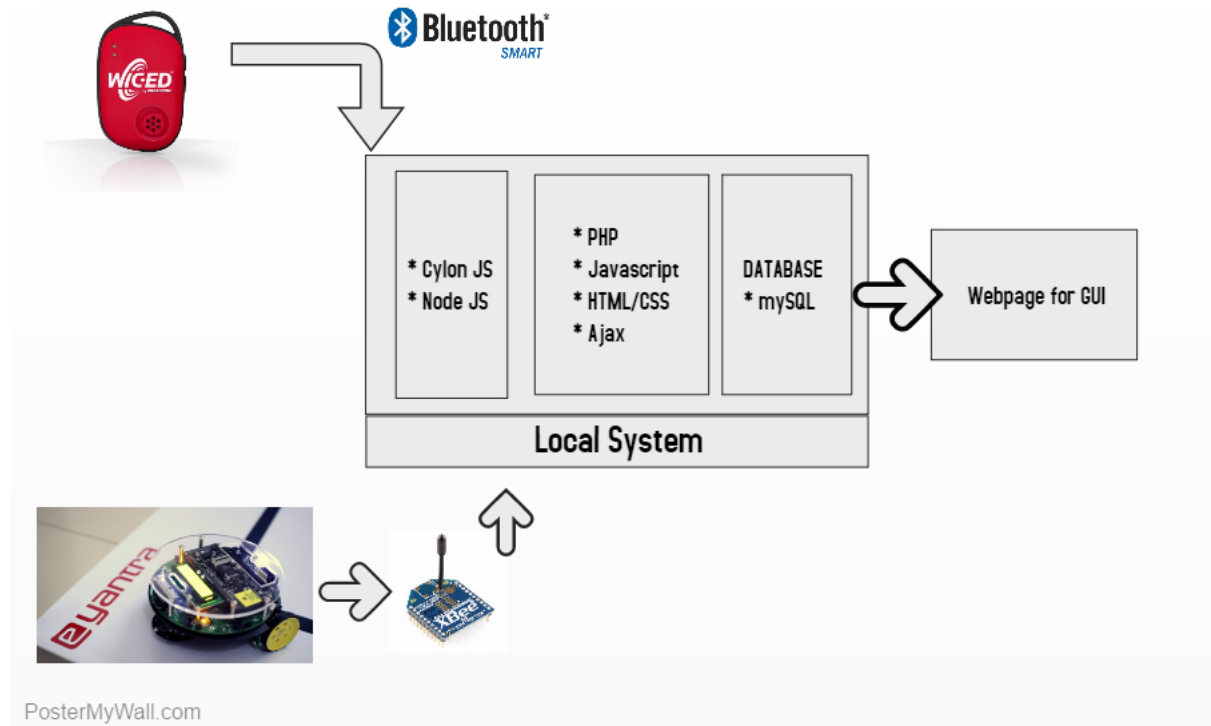


It is used to configure Zigbee Module.



## 1.5. DATA FLOWCHART

### 1.5 Data Flowchart



- The diagram represents the data flow diagram in our project.
- Wiced Sense is a BLE (Bluetooth Low Energy) device. It sends the data through Bluetooth to our local system, in our case ubuntu. We need to make sure that our system is BLE compatible.
- All the sensor data whose default packet format is explained in [Wiced Data Format](#) is sent to the local system in hex format.
- The data is taken into local system with the help of cylon-wiced-sense module. This is the link for cylon-wiced-sense codes. [wiced-sense.js driver.js](#) Its working is explained in codes.



## 1.5. DATA FLOWCHART

---

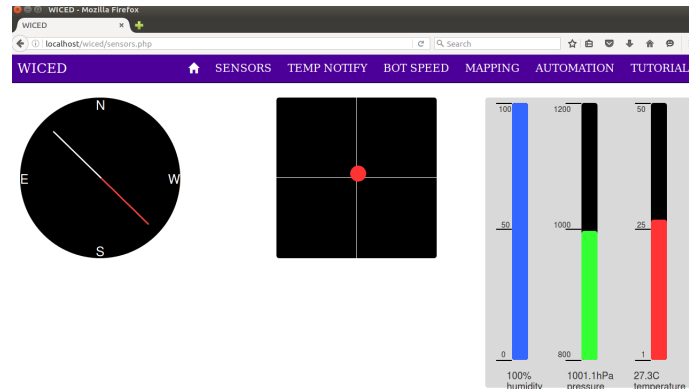
- Firebird is coded to follow the white line and send the distance travelled by the robot to the local system using zigbee. Distance travelled by the robot is given by the wheel encoders. The link to the code is given [here](#)
- The data of both the wheel encoders are sent.
- The node module serialport is used to take the data recieved on local machine through zigbee into cylon code.
- Combination of wiced data and encoder data is used for bot mapping which is explained later in the report.
- All the data recieved is then parsed and sent to php code using a node module querystring.
- It uses POST method to send data via URL to PHP.
- The PHP code whose link is provided [here](#) takes the data from URL and updates the tables in the SQL database using SQL query.
- Now the current webpage calls a PHP page asymmetrically at the server which gets the latest updated data from mySQL database and pass it to webpage via ajax. The link to php code is given [here](#)
- The data recieved on webpage is used for changing the GUI dynamically. The algorithms are created and the coding of calculations is done in javascript.
- The webpage for each application is different and is explained further.



## 1.6. WEBPAGE DESCRIPTION

### 1.6 Webpage Description

#### 1.6.1 Sensor GUI page



We have made this GUI inorder to display the sensor readings in human understandable form.

As seen in the image GUI consists of a compass , whose needle's direction is changed according to rotation of the wiced tag with respect to earth's north pole.

The Bubble level indicated the accelerometer readings. The bubble level shifts according to the acceleration in particular direction.

The 3 bars at the right of the page display the current temperature , pressure and humidity.

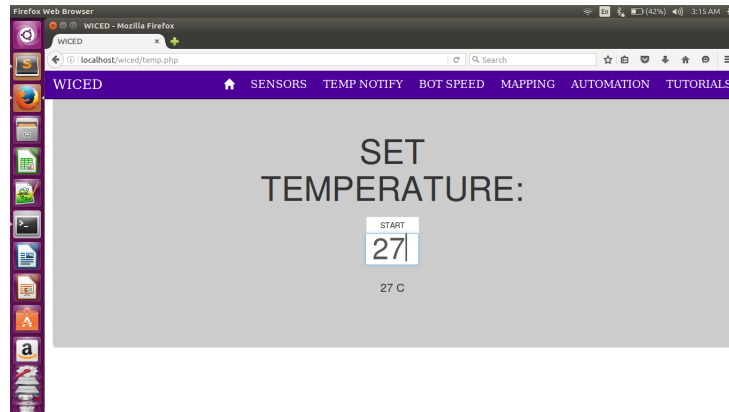
#### Code Description

- GUI for this page is made using CSS and HTML.
- This page continuously (10ms) makes a request to new.php page at server using an AJAX call.
- new.php page extracts all sensor data from mySQL tables using query and returns the extracted data to the sensor page after converting it into JASON format.
- This Jason format is parsed and all the sensor data are calculated separately to plot or change the gui dynamically.
- The link for the code is given [here](#)



## 1.6. WEBPAGE DESCRIPTION

### 1.6.2 Temperatutre notifier page



This page allows the user to set a particular temperature to trigger a event. The user sets the temperature by entering the temperature in the text box and hits start. Then this value is set as a threshold and a pop up appears if the temperature exceeds the limit.

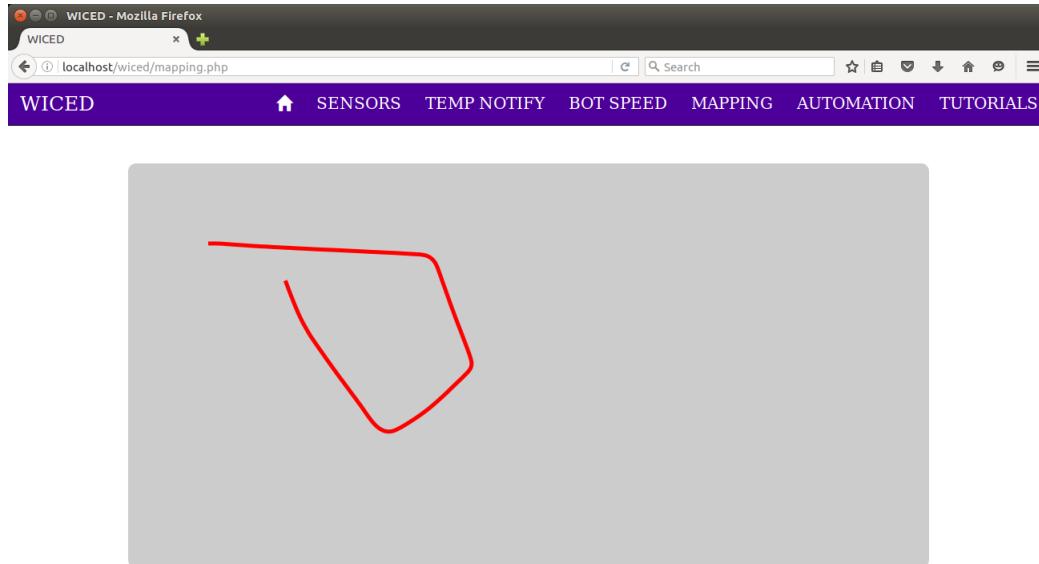
#### Code Description

- GUI for this page is made using CSS and HTML.
- This page sets the threshold temperature via HTML form and submit it to test.php
- test.php updates the threshold temperature value to mySQL database.
- This page continuously (10ms) makes a request to new.php page at server using an AJAX call.
- new.php page extracts temperature sensor data from mySQL tables using query and returns the extracted data to the temperature page after converting it into JASON format.
- Jason value of the sensor data is parsed and temp sensor data is calculated and compared with fixed/selected threshold temperature value
- As soon as the current temperature goes above fixed temperate, a popup alarm is raised using javascript alert.
- The link for the code is given [here](#)



## 1.6. WEBPAGE DESCRIPTION

### 1.6.3 Bot mapping page



This page provides a visualization of the real time movement of the robot. It uses SVG to draw the map. The path of the bot is traced on the screen as the bot proceeds. It uses data from the magnetometer to determine the angle of rotation and the value of encoder to give the distance travelled so as to facilitate path mapping.

#### Code Description

- GUI for this page is made using CSS and HTML.
- This page continuously (10ms) makes a request to new.php page at server using an AJAX call.
- new.php page extracts the sensor data of magnetometer and encoder of bot from mySQL tables using query and returns the extracted data to the mapping page after converting it into JASON format.
- This Jason format is parsed and all the sensor data are calculated separately to plot or change the gui dynamically.
- Magnetometer and Encoder readings are used to achieve bot path mapping using HTML SVG's
- The link for the code is given [here](#)



### 1.7 Future Work

- Large Scale Purpose
  - As the wiced contains accelerometer, gyroscope and magnetometer, it can be used for 3D mapping of an arena.
  - The wiced has total 5 sensors, so this tag can be used to create IOT applications
- Small Scale purpose
  - Wiced can be used as smart tags that can be attached to objects which are likely not to be found while in a hurry eg. keychain
  - It can be used to replicate the drawings made on the floor on the computer screen for further processing.

### 1.8 Bug report and Challenges

- Bugs
  1. Sometimes there is connection loss between wiced tag and PC (mostly the case of low battery)
  2. During mapping sometimes the point deviates from original position
- Challenges Faced
  1. Installing all the modules in cylon.js and node.js.
  2. Sending data from cylon code to MySQL database.
  3. Path mapping of firebird robot on webpage
- Work Halfdone
  1. Calculating velocity using wiced sense.

# Bibliography

- [1] SVG Tutorial, *SVG path*, *SVG line*, *Stroke* [Website link](#)
- [2] JavaScript Math Reference, *math.abs*, *math.ceil* , *math.round* , *math.atan2* [Website link](#)
- [3] Node js, *querystring*, *serialport* [Website link](#)