Autonomous Sentry Robot

Group 9
Brian Dodge - EE
Nicholas Musco - EE
Trevor Roman - CpE

Motivation

- We wanted to do a project that applied, and expanded, our knowledge of robotics and computer vision.
- Since we are two electrical and one computer engineering students, we wanted to do a project that had both hardware and software components and a robot is a perfect choice.
- We thought it would be an interesting idea to have an autonomous mobile security system for your home.

Goals and Objectives

- **Autonomous Control** - must be able to perform its tasks without user control
- **Remote Control** - user must be able to take control of the robot
- **Mapping and Localization** - must map an enclosed area and determine its position
- **Object Avoidance** - must avoid obstacles while moving
- **Motion Detection** - must be able to detect motion and then alert the user

Requirements and Specifications

Form Factor

Requirement ID	Requirement Description
FF1	The chassis must be low profile, no more than 1ft high, and 1.5ft wide.
FF2	The chassis must be able to hold all of the electronics, battery, and sensors.

Wheels and Motion

Requirement ID	Requirement Description
WM1	The robot must be able to move forward, backwards, and turn using wheels.

Requirements and Specifications

Control

Requirement ID	Requirement Description
C1	The robot must move smoothly with low vibration to keep the camera steady
C2	The robot must make turns smoothly in autonomous and user control modes.
C3	The robot must be able to be controlled autonomously.
C4	The robot must be able to be controlled by an user.

Requirements and Specifications

Sensors

Requirement ID	Requirement Description
S1	The robot’s sensors must be able to detect collisions, distance, and depth.
S2	The robot’s sensors must be equipped with sufficient sensors to build a reliable map of its environment.
S3	The robot’s sensors must be able to detect motion or changes in the environment instantaneously.
S4	The robot must be able to operate reliably in light or darkness.
S5	The robot’s sensors must be able to detect pitfalls, stairs, and other obstacles from which it cannot escape.

Requirements and Specifications

Processing

Requirement ID	Requirement Description
P1	The robot must be able to autonomously navigate and map in real time.
P2	The robot must reliably operate, react, and make decisions within 1-3 seconds.
P3	The robot must be able to find its docking station and successfully dock to charge.
P4	The robot must have 75% certainty of detections before alerting its user.
P5	The user must receive alert notifications from ASR within 5 seconds of detection.
P6	The robot must alert user if damaged or stuck.
P6.1	If stuck, must attempt extricate itself for at least 10 seconds before alerting user, then continuing to try and escape
P7	If successful at extrication, the robot must alert user within 10 seconds.

Requirements and Specifications

Power

Requirement ID	Requirement Description
PW1	The robot must be able to operate for at least 5 hours on a full charge.

Design Constraints

- Low Cost
- Low Power
- Lightweight
- Small Form Factor
- Less Time

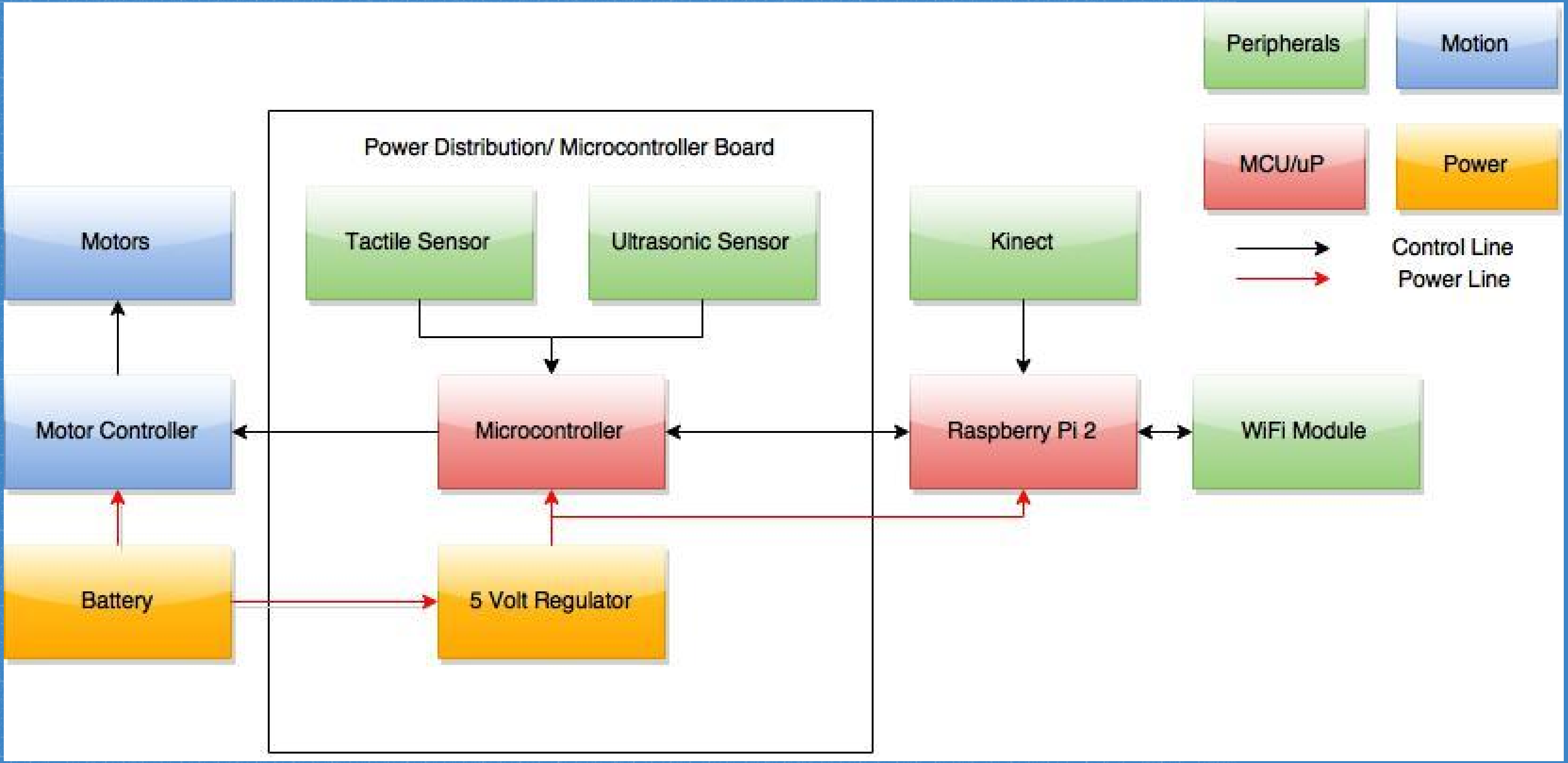
Related Standards

Standard Number	Scope	Title
IEEE 802.11n-2009	WiFi	IEEE Standard for Information technology -Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput
IEC 62680-1 Ed. 1.0 b:2013	USB	Universal serial bus interfaces for data and power - Part 1: Universal serial bus specification, revision 2.0
IEC 62680-2 Ed. 1.0 b:2013	USB	Univ. serial bus interfaces for data and power - Part 2: Universal serial bus - Micro-USB cables and connectors specification, revision 1.01

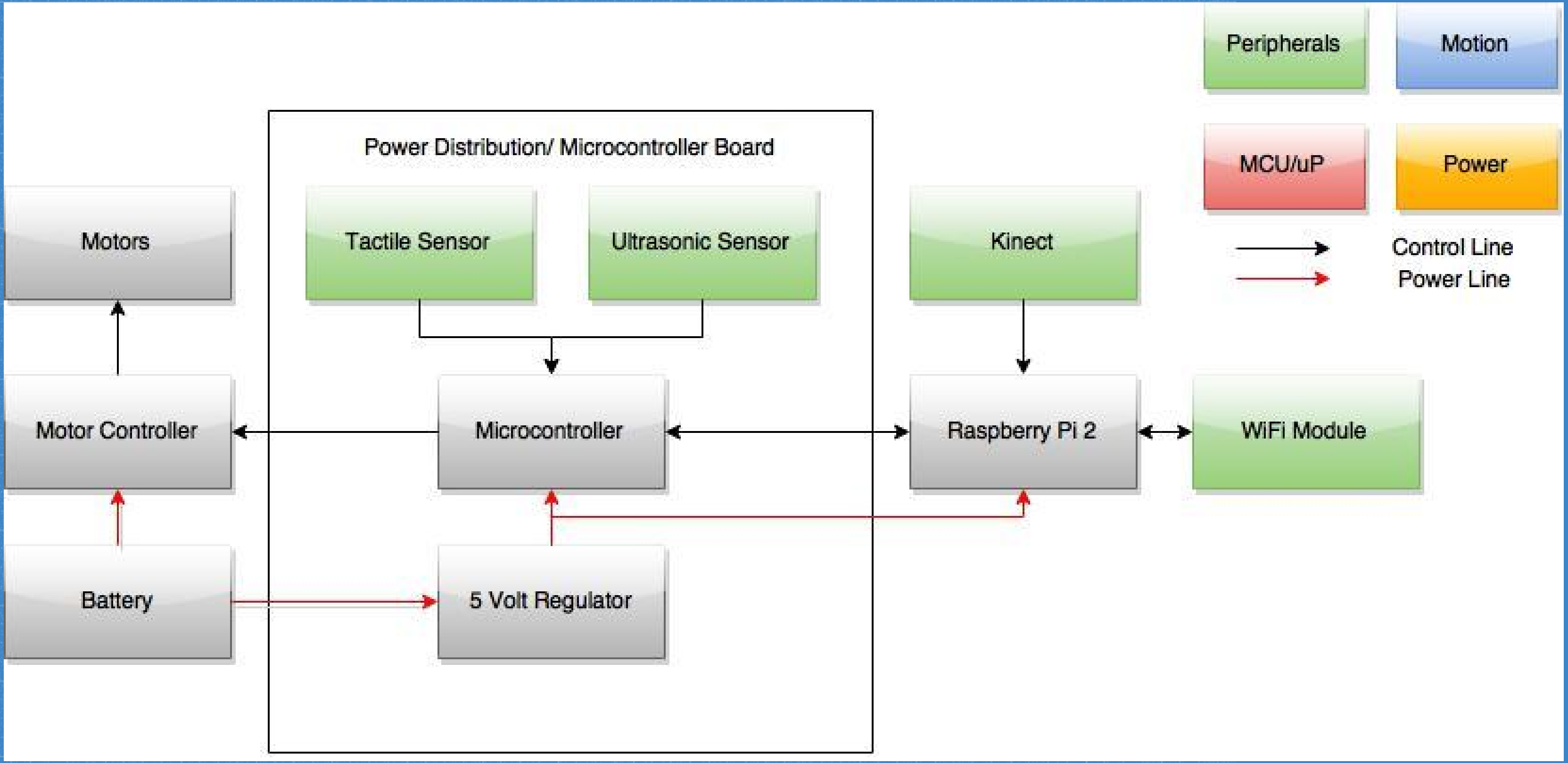
BSR/IEEE 1873-201x	Mapping	Standard for Robot Map Data Representation for Navigation
IEC 60335-2-29 Ed. 4.2 b:2010	Battery Charger	Household and similar electrical appliances - Safety - Part 2-29: Particular requirements for battery chargers
IEC 62676-1-1 Ed. 1.0 b:2013	Video Surveillance	Video surveillance systems for use in security applications - Part 1-1: System requirements - General
IEC 62676-2-2 Ed. 1.0 b:2013	IP video	Video surveillance systems for use in security applications - Part 2-2: Video transmission protocols - IP interoperability implementation based on HTTP and REST services

Hardware Design

Overall Block Diagram



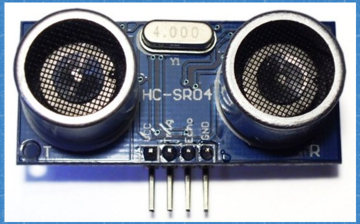
Peripherals



Ultrasonic Sensor - HC-SR04

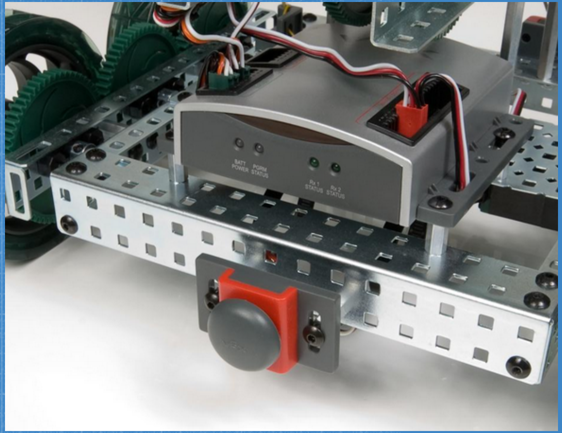
- Middle layer of obstacle avoidance
- Four Ultrasonic Sensors
  - Two on the front corners
  - Two on the back corners

Working Voltage (V)	5
Working Current (mA)	15
Minimum Range (cm)	2
Maximum Range (M)	4
Measuring Angle	15 degree
Price ($)	8.99/2



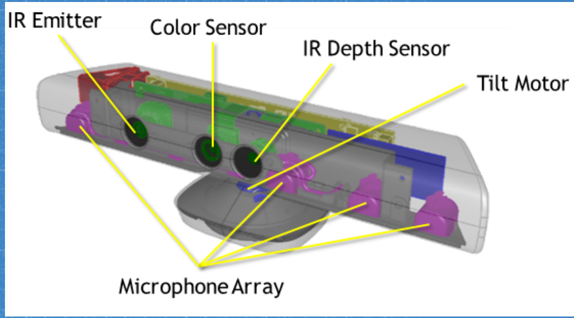
Tactile Sensor

- Used as last layer of obstacle avoidance.
- Four Vex Bumper Switches
  - Two on the front corners
  - Two on the back corners
  - $12.99 for two



Microsoft Kinect

- Developed for the Xbox 360
- RGB camera with a resolution of 1280x960
- Infrared (IR) emitter and an IR depth sensor
- Microphone array
- 3-axis accelerometer
- Tilt Motor
- Requires a power adapter to work with a computer
- Costs $20.00 (used)

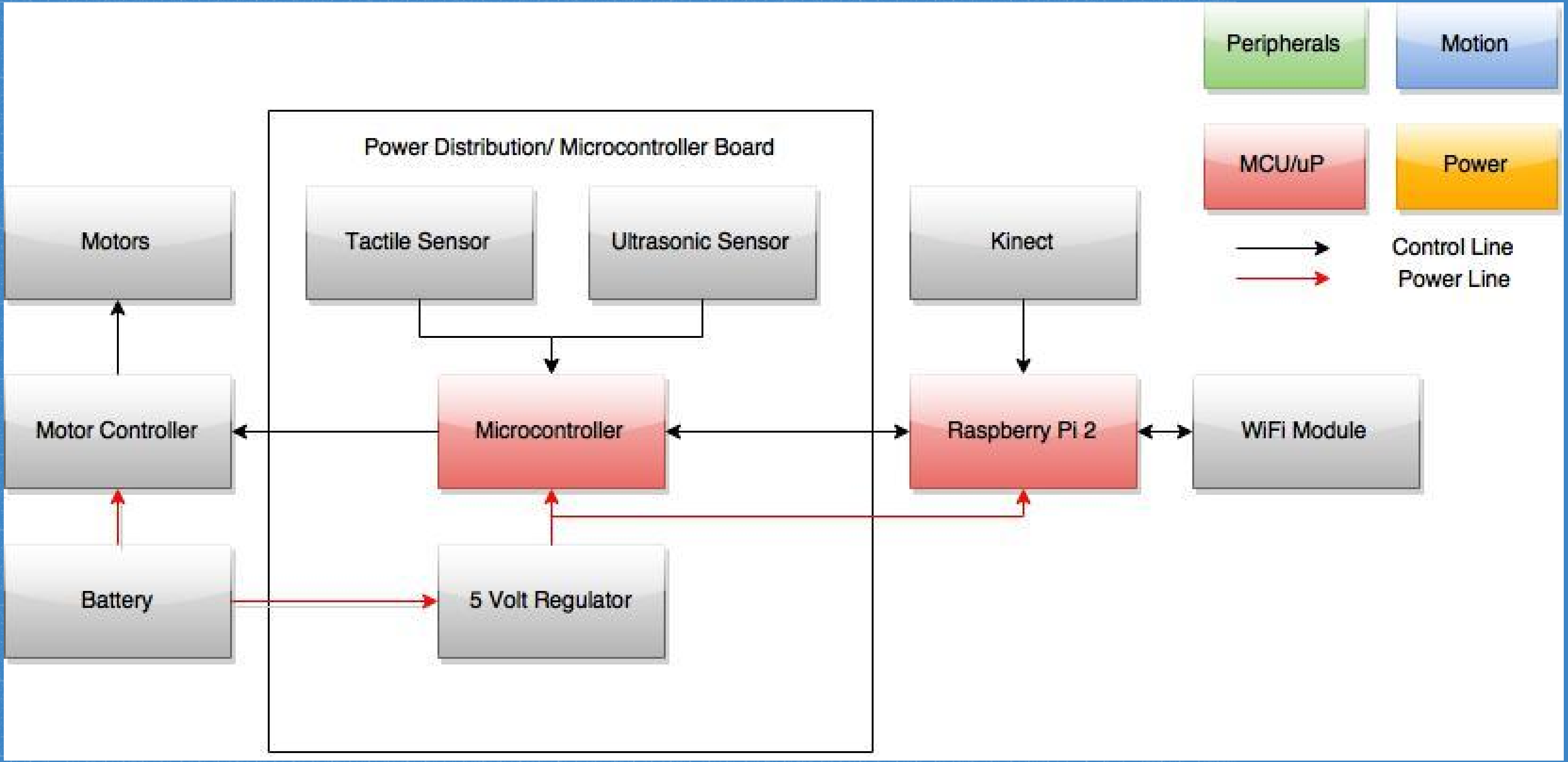


Wireless Connectivity

- The Raspberry Pi 2 will be a server
- Plug and play USB WiFi NIC
- Connect via SSH
- Use a remote desktop client for headless access
- Chose the EW-7811, the cheapest option

NIC	Max Data Rate	Frequency	Security	Cost
PAUO5	300Mbps 802.11n	2.4GHz	64b/128bit WEP, WPA and WPA2 (TKIP+AES)	$16
PAUO6	300Mbps 802.11n + 5dBi antenna	2.4Ghz	64b/128bit WEP, WPA and WPA2 (TKIP+AES)	$20
EW-7811Un	150Mbps 802.11n	2.4Ghz	64/128bit WEP Encryption and WPA-PSK, WPA2-PSK security; WPS	$10

Microcontroller / Microprocessor

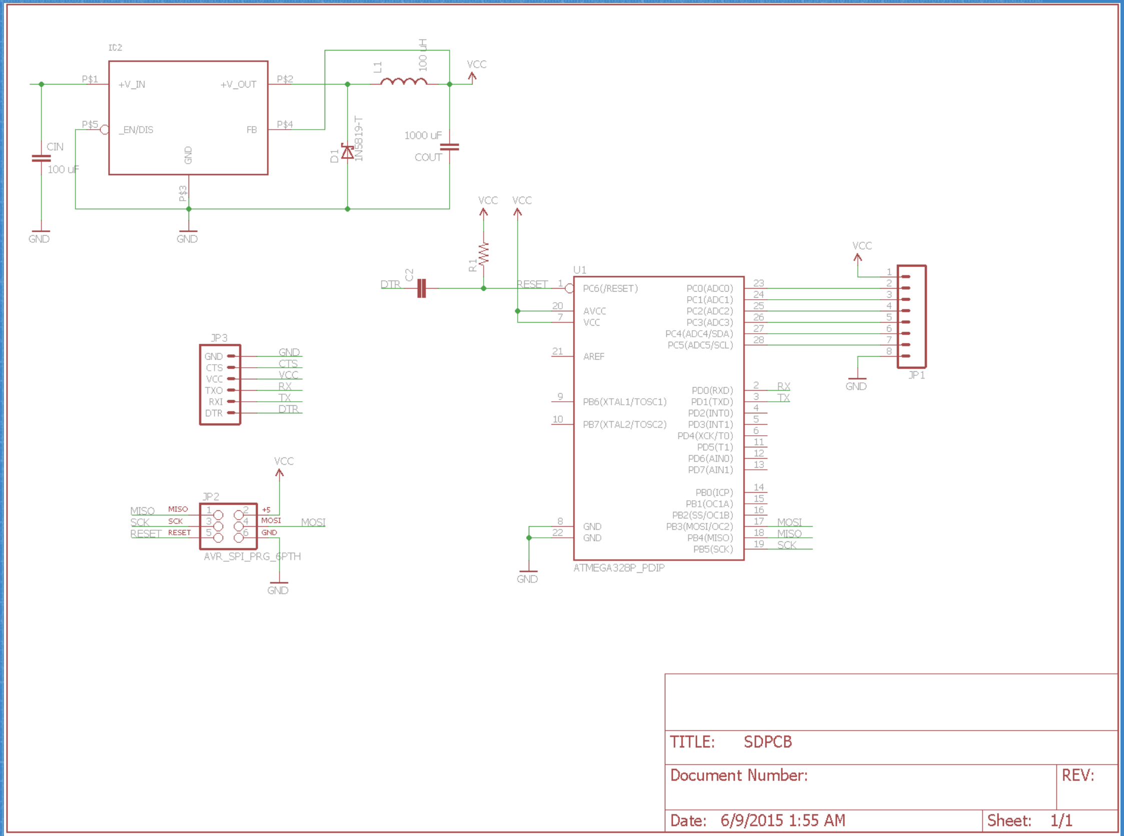


Microcontroller

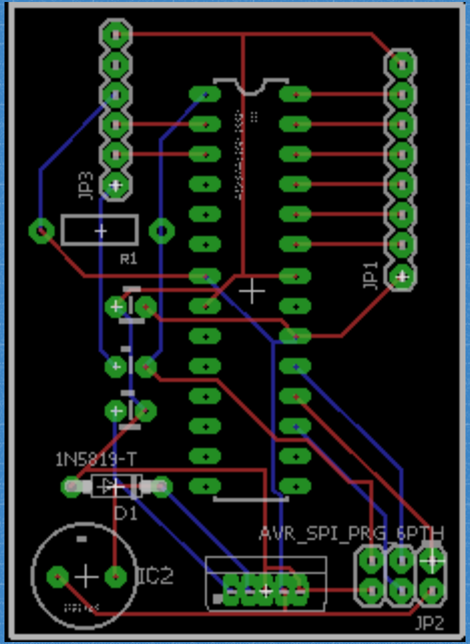
- ATmega328P
- Arduino Bootloader
- Will use a 28 pin DIP socket on the PCB.
- Program with a Mintduino kit which comes with an FTDI programmer.

Architecture (bits)	8
Frequency (MHz)	20
Max Operating Voltage (V)	5.5
Program Memory (KB)	32
RAM (KB)	2
USART/SPI	1/1
I2C	1
I/O Pins	26 max
Price per Unit	$3.38 (Digi-Key)

PCB Schematic



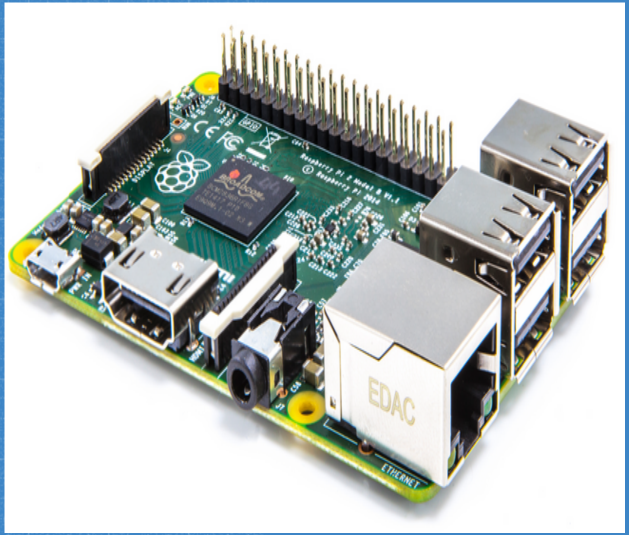
PCB Layout



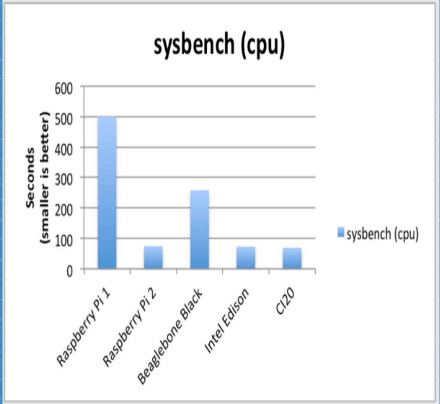
Microprocessor

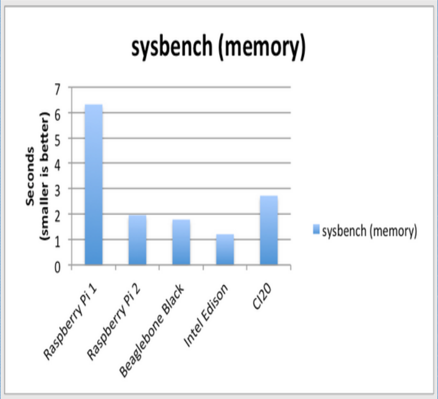
- Raspberry Pi 2
- Cheap, powerful, trusted platform
- Compatible with ROS

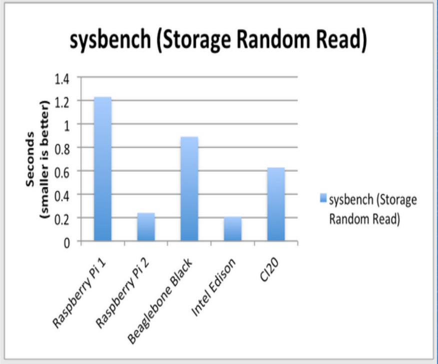
Released	February 2015
Price	$35 USD
OS	Linux, Windows 10, RISC OS, FreeBSD, NetBSD, Plan9, Inferno
SoC	Broadcom BCM2836 (CPU, GPU, DSP, SDRAM, 1 USB)
CPU	900 MHz quad-core ARM Cortex-A7
GPU	Broadcom Videocore IV 250MHz, OpenGL ES 2.0
RAM	1GB SDRAM
Storage	MicroSDHC
Network	10/100 Mbit/s Ethernet (8P8C)
Video Output	HDMI 640x350 - 1920x1200
USB	4x USB 2.0
Power	5V, 800mA, 4.0 W
Size	85.6mm x 56.5mm
Weight	45g



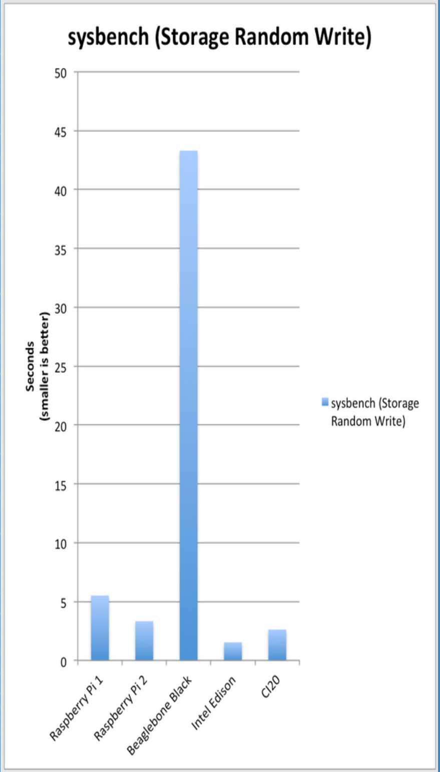
Microprocessor Benchmarks



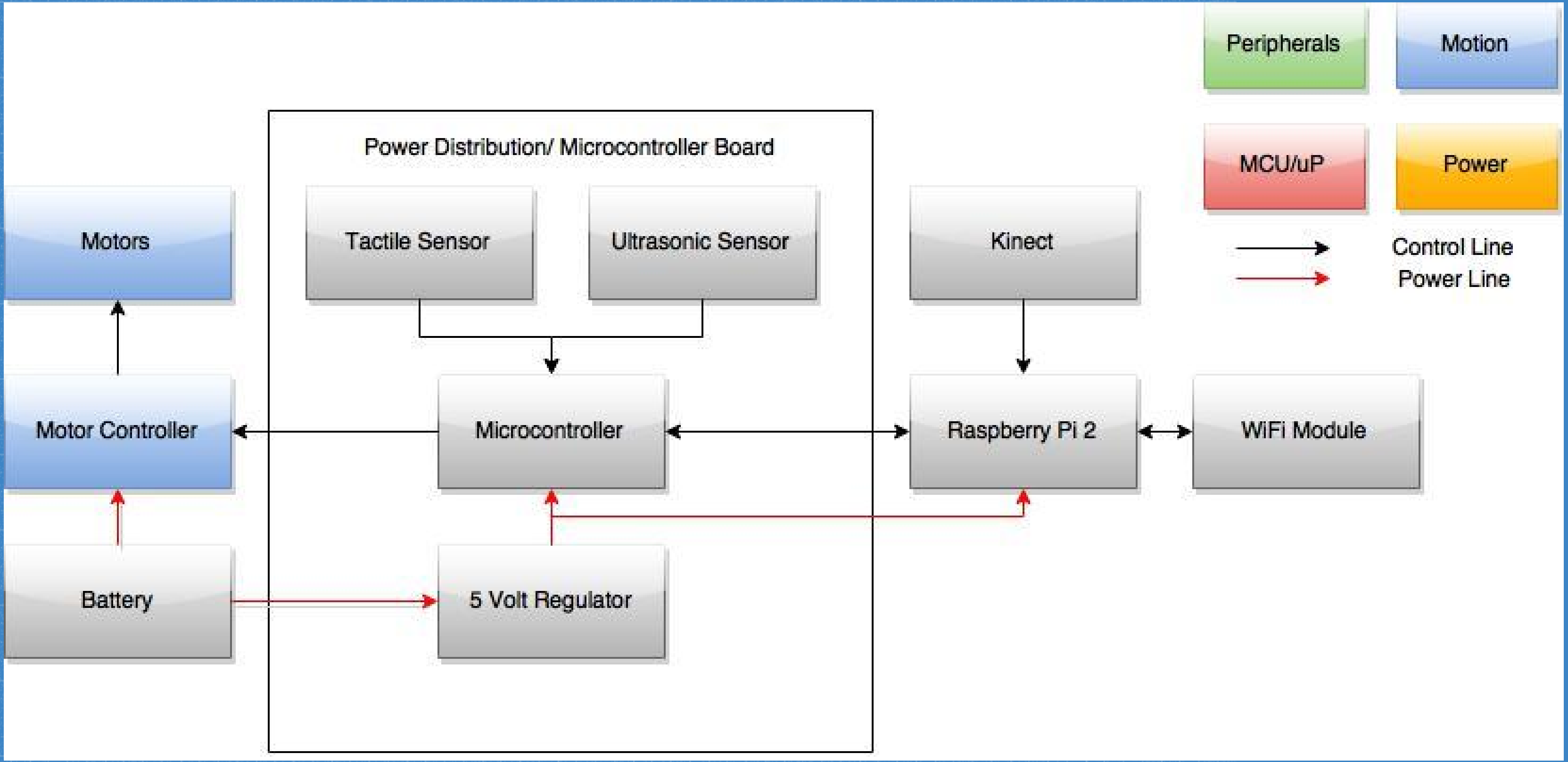




- Smallest number is best
- Pi 2 outperforms or performs nearly as well as more expensive boards in all benchmarks



Motor Controller and Motors

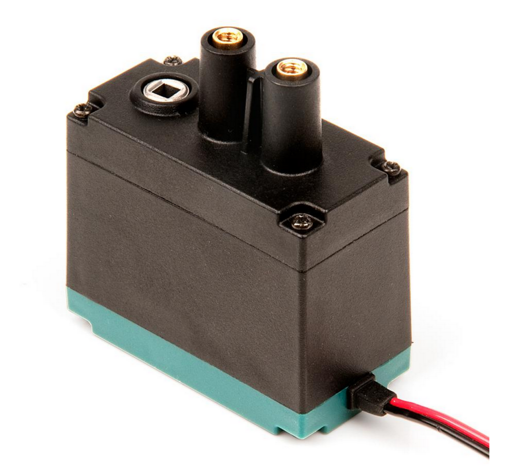


VEX Motor Controller 29

- Small form factor
  - <2" in length
  - <1" width and height
- Inexpensive
  - $10.00 x 4
- Uses Pulse Width Modulation
  - Speed Control
  - Direction Control

Vex 2-Wire Motor 393

- Runs on 7.2V
- Multiple Gear Ratios
- $14.99 x 4
- Compatible with the VEX Chassis



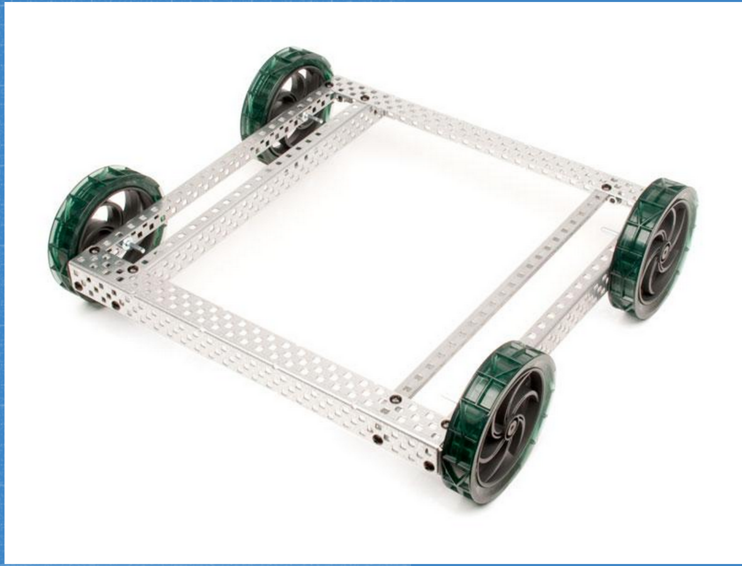
	Output Stage Driving Gear	Output Stage Driven Gear	Output Speed (RPM)	Output Stall Torque (N*m)	IME Ticks per Revolution
Standard Motor 393 Gearing	10t	32t	100	1.67	627.2
High Speed Option (included with Motor 393)	14t	28t	160	1.04	392
Turbo Gear Set (sold separately)	18t	24t	240	0.7	261.333

Wheels

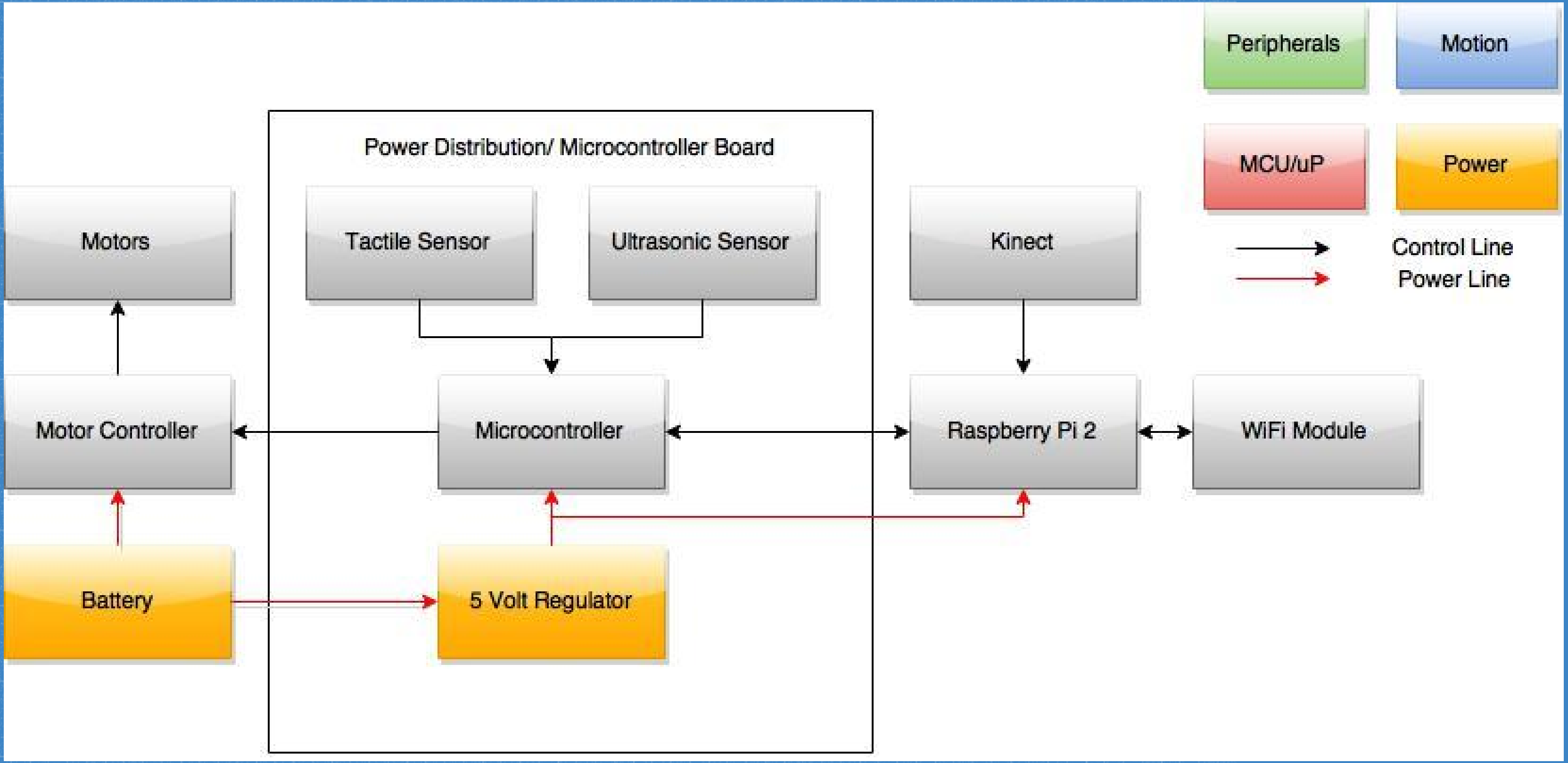
Wheel Type	Mecanum	Omni	Traction
Image			
Price	$60.00 for 4	$25.00 for 2	$10.00 for 4
Advantage	• Best Mobility • Can move in any direction	• Reduces Friction • Greater Mobility	• Simple • Inexpensive • Great traction
Disadvantage	• Slippage • Complexity • Price	• Slippage • Price	• Lacks mobility

Chassis

- Medium Chassis Kit - VEX Robotics
- Rectangular Form Factor
- 12.6″ X 12.6″
  - Acceptable Size
  - Adjustable width
- Polycarbonate top
- Chassis only - $21.35
- Suitable for tank drive



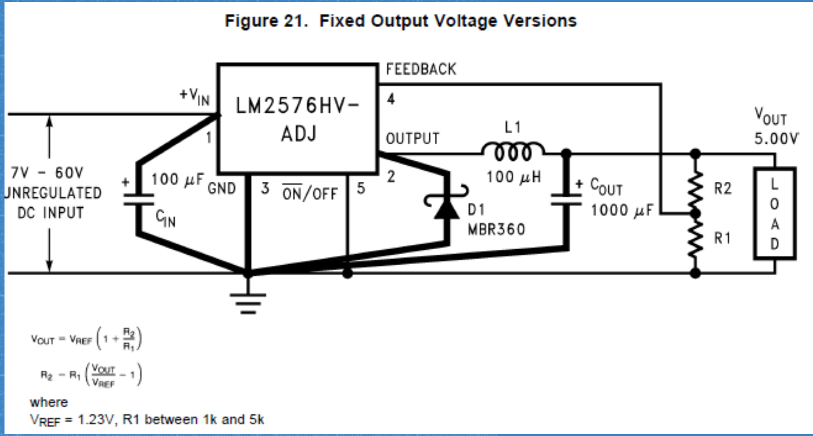
Power



Peripherals
Motion
MCU/uP
Power
Control Line
Power Line
Power Distribution/ Microcontroller Board
Motors
Tactile Sensor
Ultrasonic Sensor
Kinect
Motor Controller
Microcontroller
Raspberry Pi 2
WiFi Module
Battery
5 Volt Regulator

5 Volt Regulator

Voltage Regulator	**LM2576-ADJ**
Max Input Voltage (V)	40
Output Voltage (V)	1.23 to 37
Peak Current (A)	3



Reference Schematic from TI's LM2576 Datasheet

Nickel-Metal-Hydride Battery

Advantages	30–40 percent higher capacity than a standard NiCd Less prone to memory than NiCd Simple storage and transportation; not subject to regulatory control Environmentally friendly; contains only mild toxins Nickel content makes recycling profitable
Limitations	Limited service life; deep discharge reduces service life Requires complex charge algorithm Does not absorb overcharge well; trickle charge must be kept low Generates heat during fast-charge and high-load discharge High self-discharge; chemical additives reduce self-discharge at the expense of capacity Performance degrades if stored at elevated temperatures; should be stored in a cool place at about 40 percent state-of-charge

Nickel-Metal-Hydride Battery Cont.

- Safety is the number one priority
- NiMH was chosen over Lithium Ion due to the charging station
- When the system is perfected a switch to Lithium Ion could occur for greater battery life

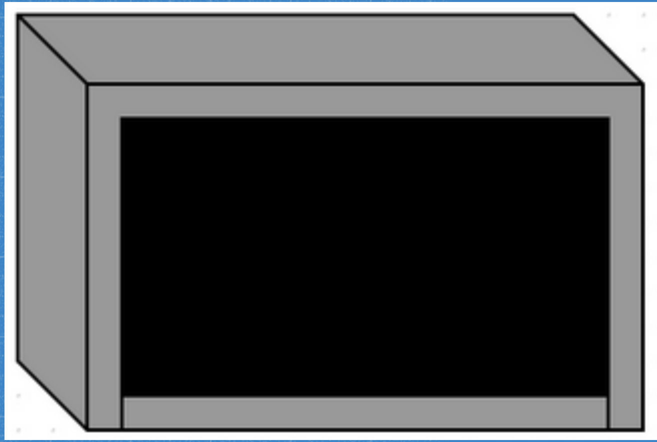
Nickel-Metal-Hydride Battery Cont.

- Price was also important
- Designed to be as cost effective as possible
- 5000mAh will allow our battery to run for an acceptable amount of time

Battery (Chemistry)	Capacity (mAh)	Price ($)	Voltage (V)
Tenergy (NiMh)	5000	32.99	7.2
Tenergy (Li-Ion)	5000	55.00	7.4

Charging Station

- Designed to be an end table
- Utilizes a Tenergy Universal Smart Charger
  - Used for 6-12V Batteries
  - Used with NiMH and NiCd
- Wood construction
- Vented for safety

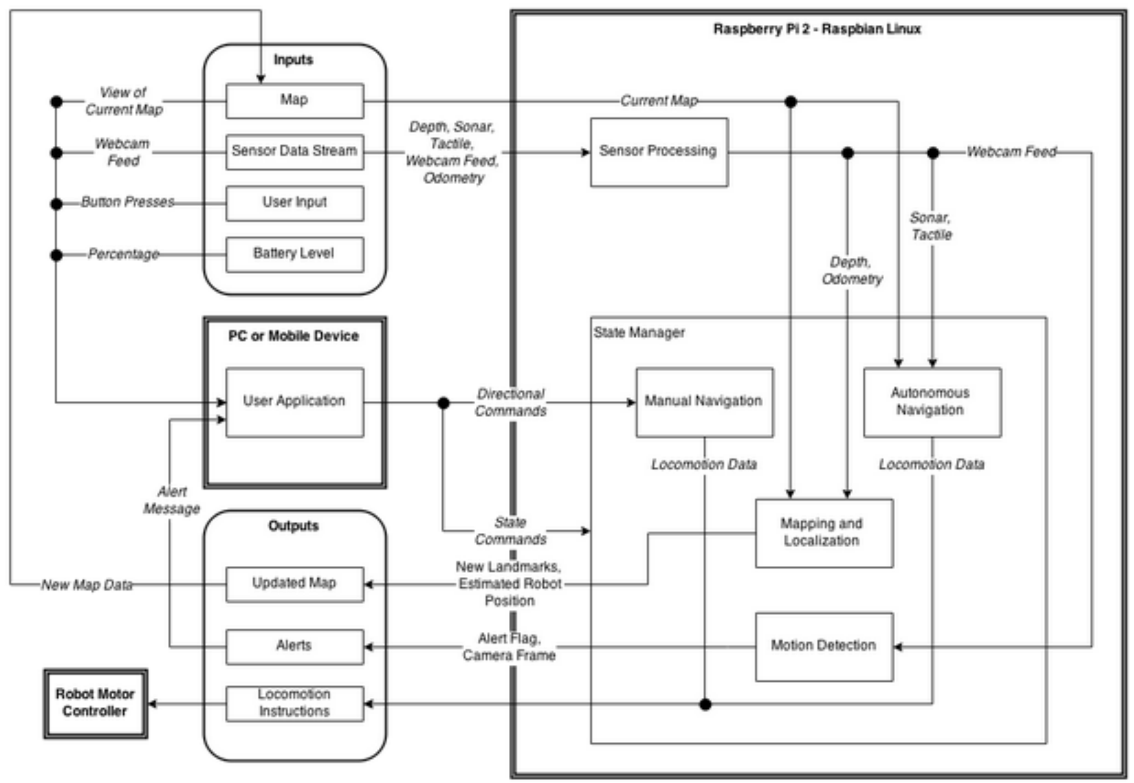


Charging Station

- Battery charger leads will be enclosed and insulated to prevent shock
- The lead enclosure will open when the ASR enters the station
- The station entrance will be just larger than the robot

Software Design

High Level Software Architecture



Development Environment

- Laptop running Ubuntu Linux
- Raspberry Pi 2 running Raspbian Linux
- ROS
  - SLAM
  - Hardware Drivers
  - General Framework
- OpenCV
  - Computer Vision Systems

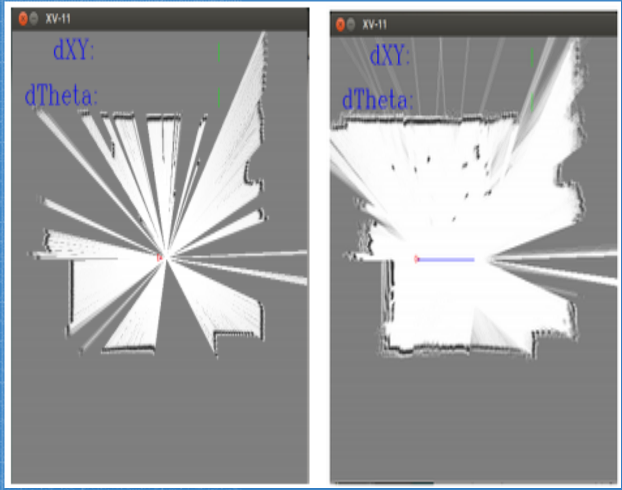
SLAM

- Simultaneous Localization and Mapping
- Original approach
  - Modify BreezySLAM
  - Didn’t originally plan on using ROS
- New Approach - Experimental
  - Use available ROS packages for SLAM
  - HectorSLAM - Uses No Odometry
  - GMapping - Uses Odometry

SLAM Examples

HectorSLAM

GMapping



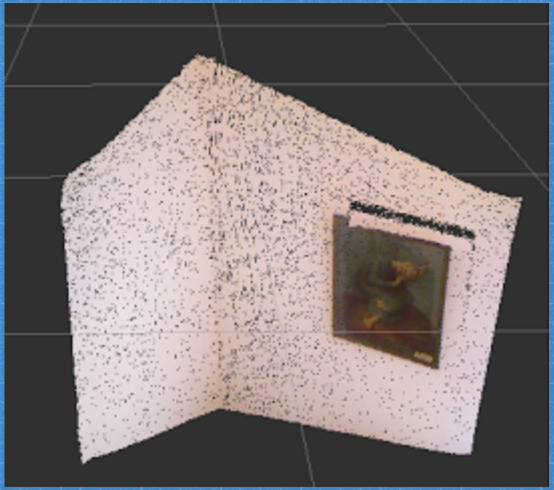
BreezySLAM

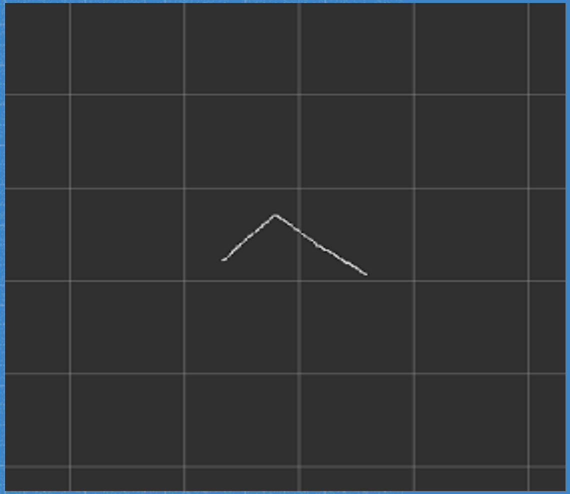
SLAM With Kinect

- Problem
  - Kinect supplies 3D depth clouds
  - Our SLAM choices output 2D grids
- Solution
  - Slice the depth cloud
  - Trick SLAM into thinking the kinect is a laser scanner

R.O.S. is Awesome

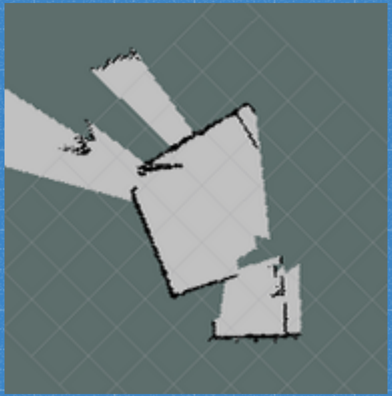
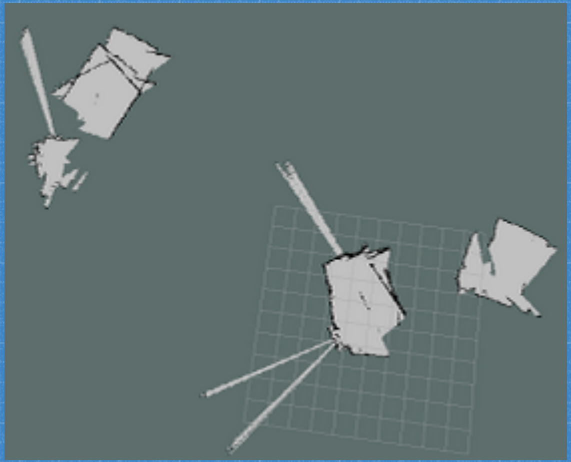
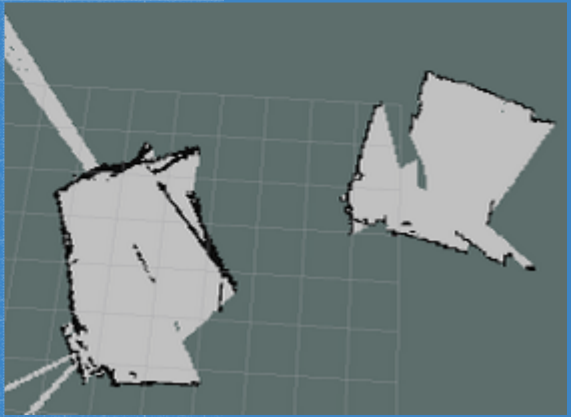
- Our idea was available as a ROS node
- Here it is, already working:





R.O.S. is Awesome

- While we're at it, here's our prototype of the SLAM system "working" as well:

R.O.S.

“The Robot Operating System (ROS) is a flexible framework for writing robot software. It is a collection of tools, libraries, and conventions that aim to simplify the task of creating complex and robust robot behavior across a wide variety of robotic platforms.”

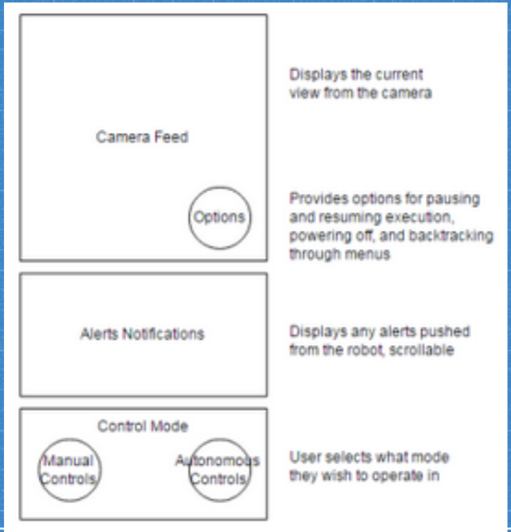
R.O.S.

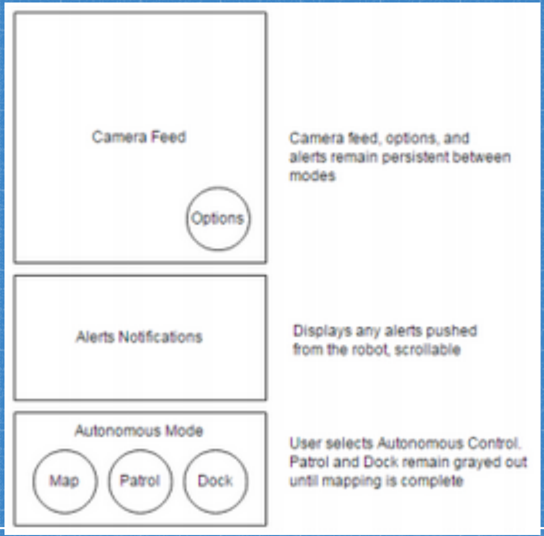
- “Meta” Operating System
- Open-source under BSD License
- C++, Python, Java, Lisp
- Network of nodes (processes)
- ROS Core
  - ROS Master
  - Parameter Server
  - rosout

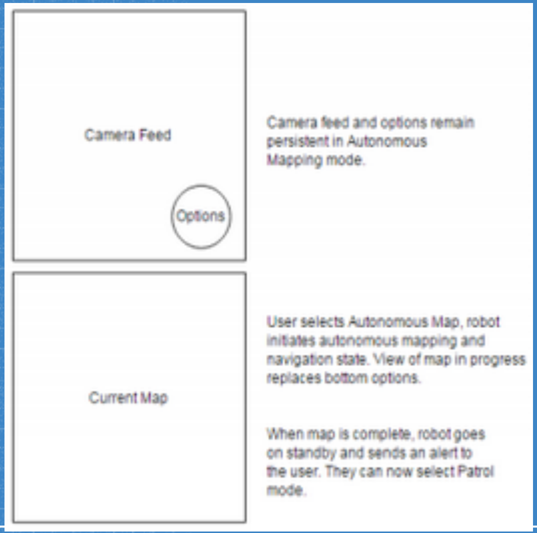
- Topics
  - Stream-like Communication
  - TCP/IP or UDP
  - Publishers
  - Subscribers
- Services
  - TCP/IP or UDP
  - Function-like Communication
  - Server
  - Client

Control - Option A

- Design a simple user-facing application
- Only necessary buttons, options, and features
- Persistent camera frame
- Web or mobile version
- Constrained by time and knowledge, but better for users







Control - Option B

- Treat the ASR more like a remote access computer
- SSH in, give instructions through the command line
- Integrate ROS visualization tools
- Easier for us, more time for other systems
- Worse for user
- Ex) Start autonomous mapping procedure

```
pi@raspberrypi ~ $ asr mapping -a
```

Web vs. Mobile

Best Option:

- Chrome / Firefox Compatible Web App
  - Capture 81.53% of market
- Android 4.1 - 5.1 Compatible App
  - Capture 47.5% of mobile market
  - Capture 87.5% of Android market

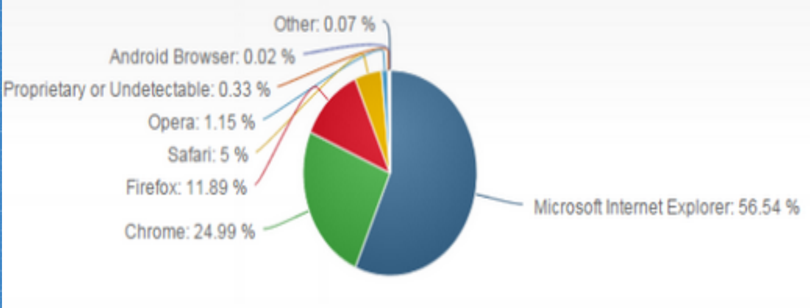


Figure 3.13.2.2a: Market Share for Browsers - 3/15/2015

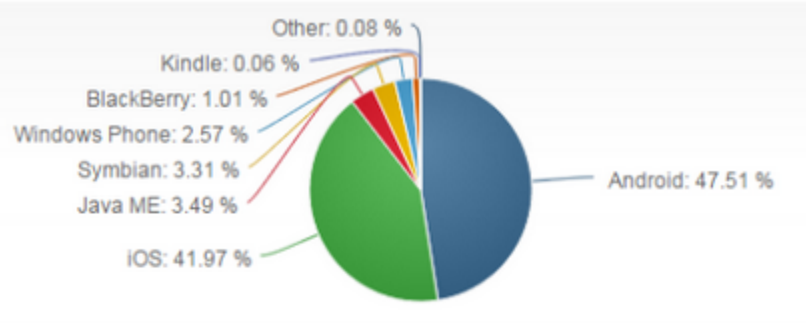


Figure 3.13.2.2b: Market Share for Mobile Operating Systems

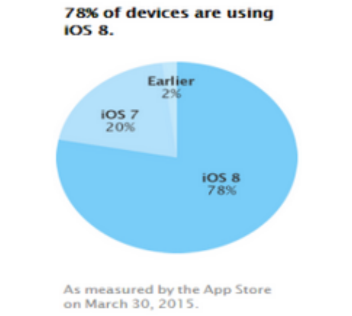


Figure 3.13.2.2d: Market Share of iOS Versions

Version	Codename	API	Distribution
2.2	Froyo	8	0.4%
2.3.3 - 2.3.7	Gingerbread	10	6.4%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	5.7%
4.1.x	Jelly Bean	16	16.5%
4.2.x		17	18.6%
4.3		18	5.6%
4.4	KitKat	19	41.4%
5.0	Lollipop	21	5.0%
5.1		22	0.4%

Figure 3.13.2.2c: Market Share of Android Versions

State Manager

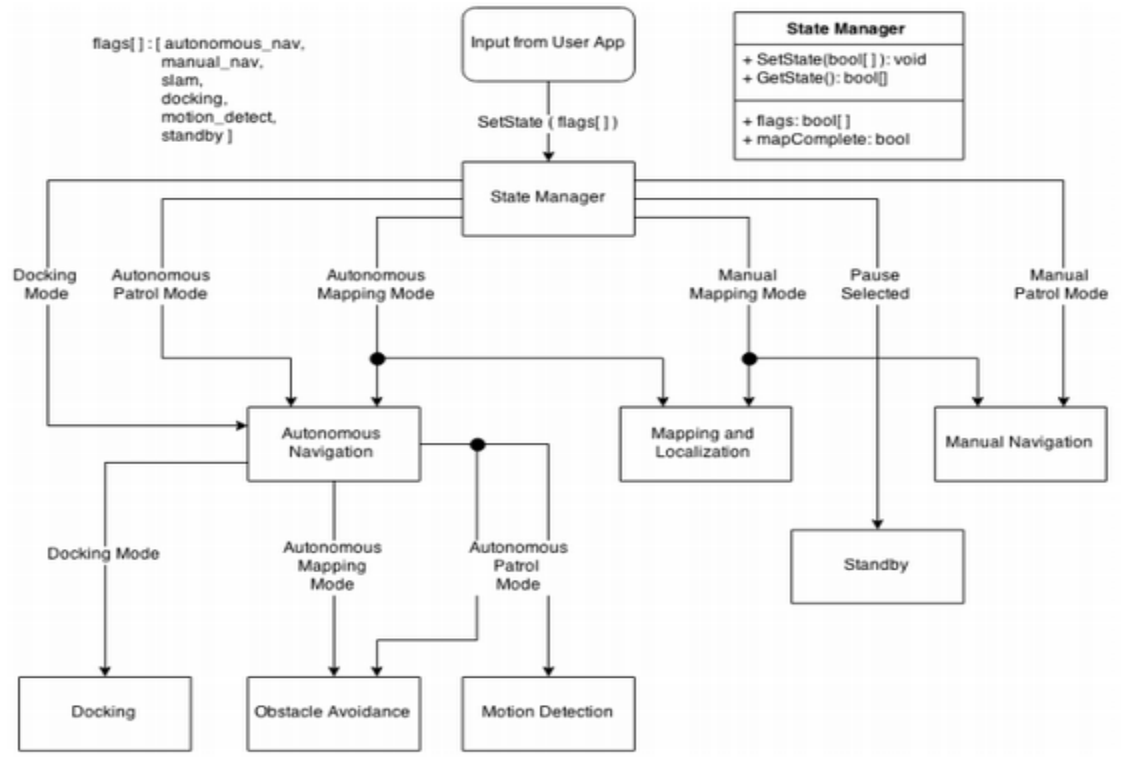


Figure 7.4: State Manager Architecture

Autonomous Exploration

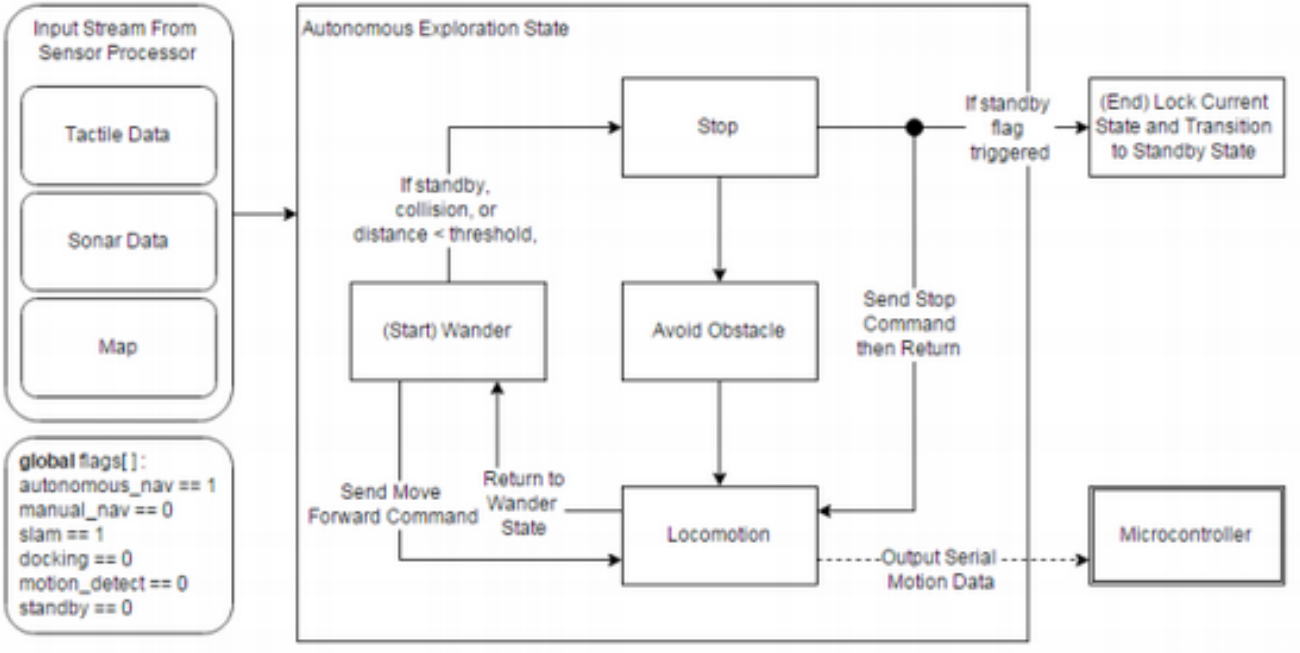


Figure 7.5.1: Autonomous Exploration State Architecture

Autonomous Patrol

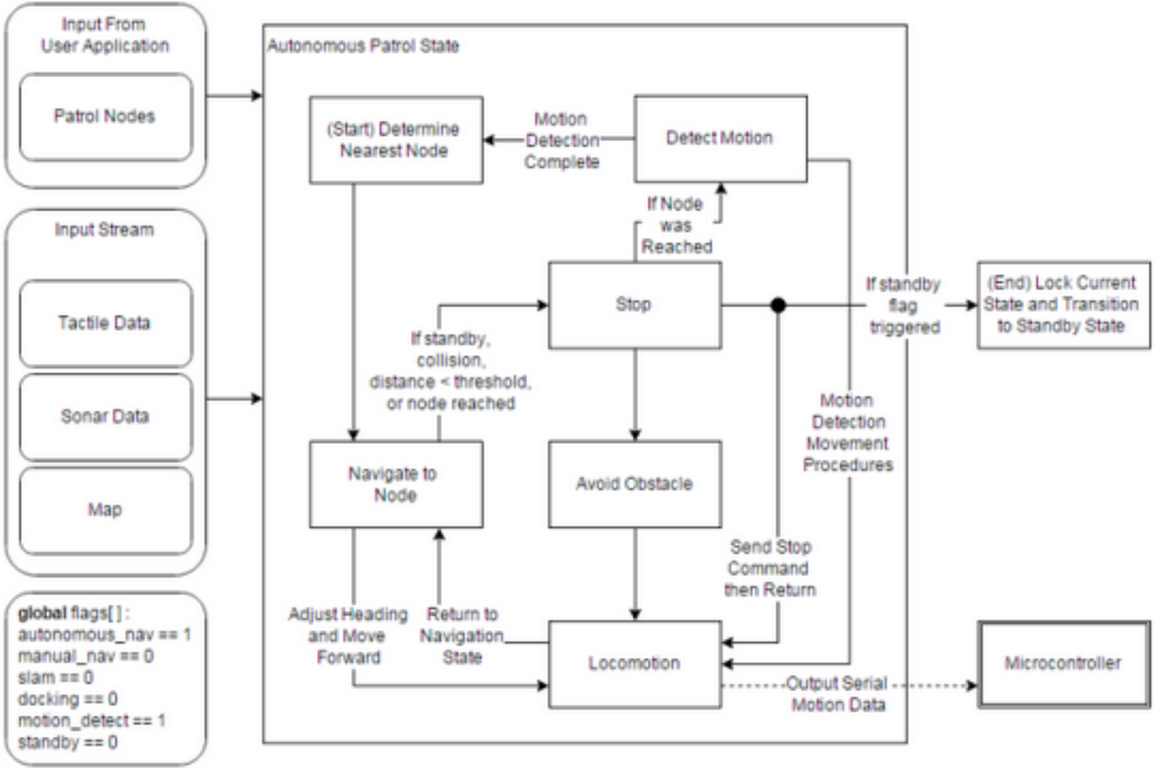


Figure 7.5.2: Autonomous Patrol State Architecture

Autonomous Docking

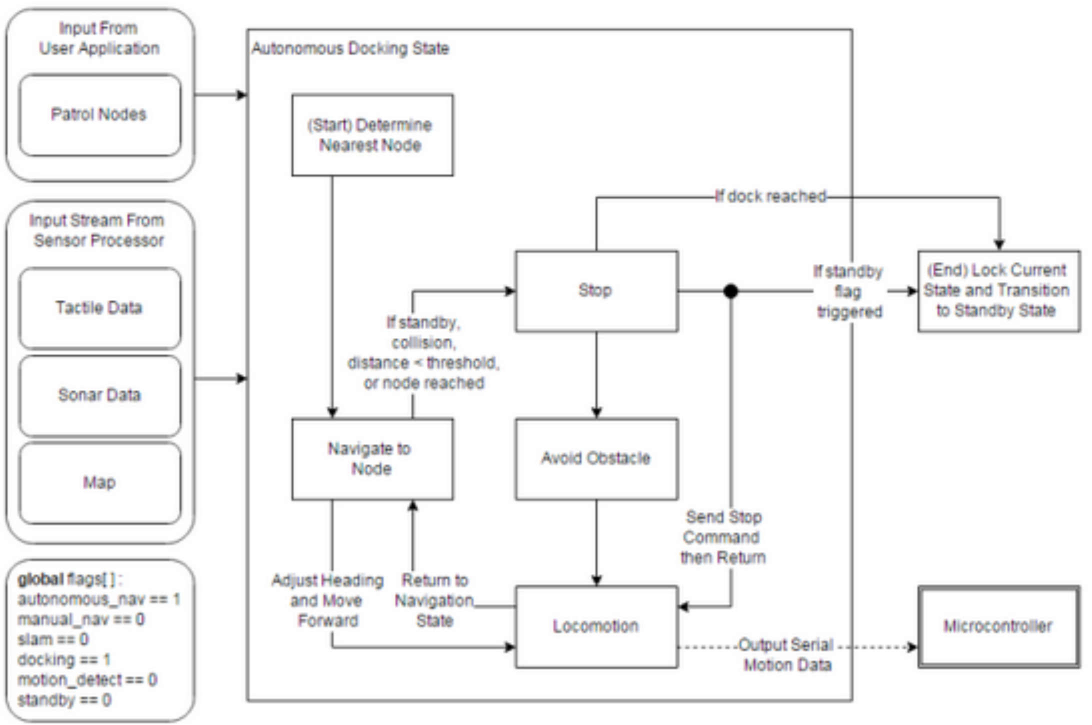


Figure 7.5.1: Autonomous Docking State Architecture

Manual Exploration / Patrol

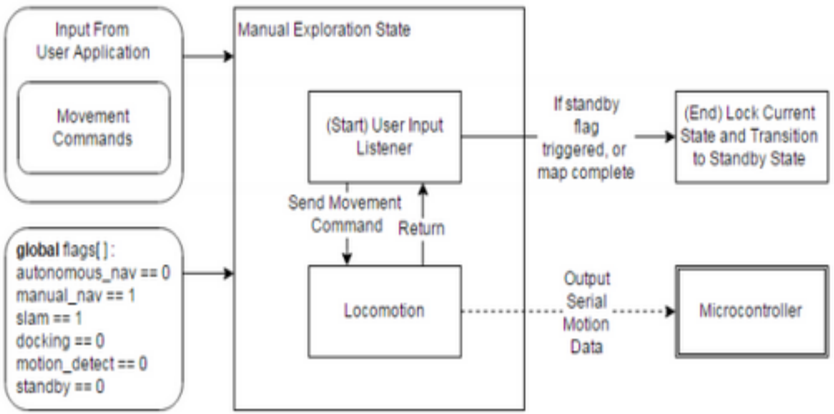


Figure 7.6.1: Manual Exploration State Architecture

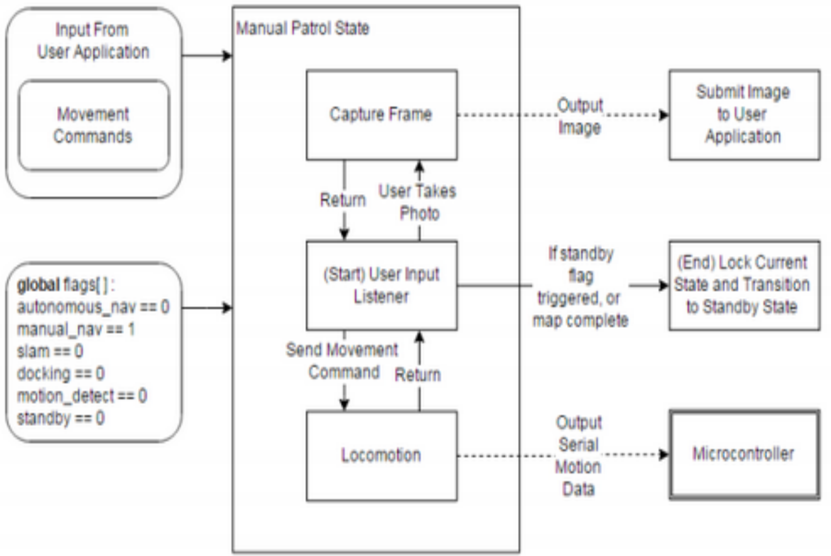


Figure 7.6.2: Manual Patrol State Architecture

Motion Detection

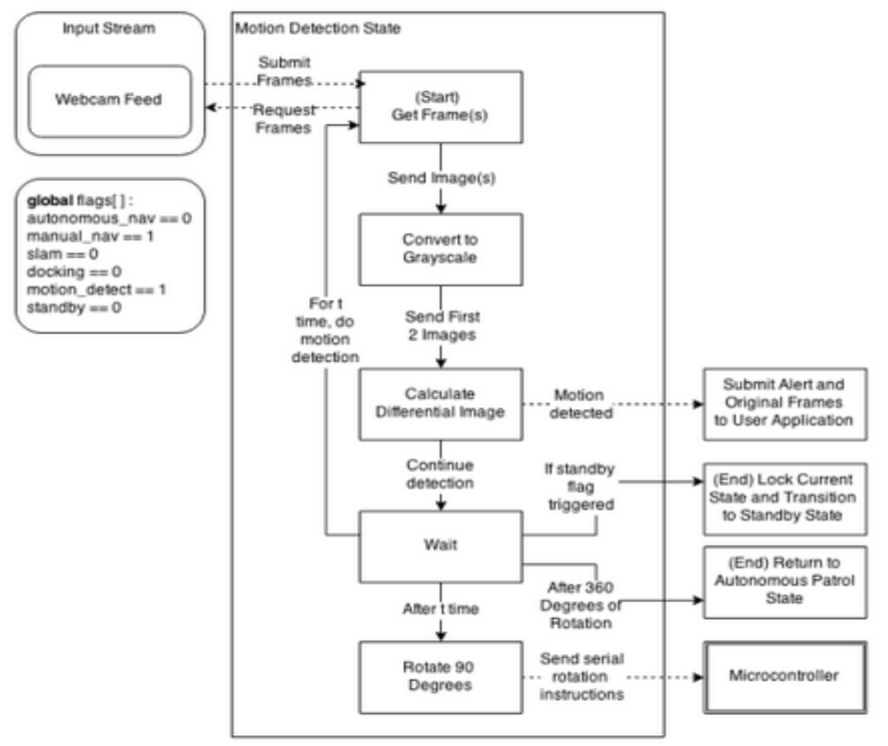


Figure 7.8: Motion Detection State Architecture

Administrative Content

Work Distribution

Name	Electrical System	Hardware Assembly	Hardware Programming	Software Systems
Brian	P	S	P	S
Nick	S	P	T	T
Trevor	T	T	S	P

P - primary
S - Secondary
T - Tertiary

Development Budget

Part	Quantity	Unit Price	Expected Cost	Actual Cost
Robot Chassis - Vex medium chassis	1	$21.35	$21.35	$21.35
3.25 inch Vex Wheels	4	$19.99 (for four)	$19.99	$0.00- Already Owned
Microsoft Kinect	1	$20.00 (Used)	$20.00(Used)	$0.00 - Already Owned
Ultrasonic Module HC-SR04 Distance Sensor	4	$8.99 (for two)	$17.98	$8.99 - Already Own 2
VEX Bumper Switch	4	$12.99 (for two)	$25.98	$12.99 - Already Own 2
ATmega328P	1	$3.70	$3.70	$0.00 - Already Owned
Mintduino	1	$24.99	$24.99	$0.00 - Already Owned
Raspberry Pi 2 Model B	1	$35.00	$35.00	$0.00 - Already Owned
Vex 393 Motors and Motor Controller 29	5	$24.98	$124.90	$124.90
PCB	1	$66.00	$66.00	$66.00
Edimax EW-7811UN WiFi transmitter	1	$14.99	$14.99	$0.00 - Already Owned
Tenergy 7.2V 5000mAh NiMH battery	1	$89.00 (set of two)	$89.00	$89.00
Tenergy Battery Charger	1	$22.99	$22.99	$22.99
Power Regulators components	N/A	N/A	$50.00	$50.00
Grand Total			$536.87	$396.22

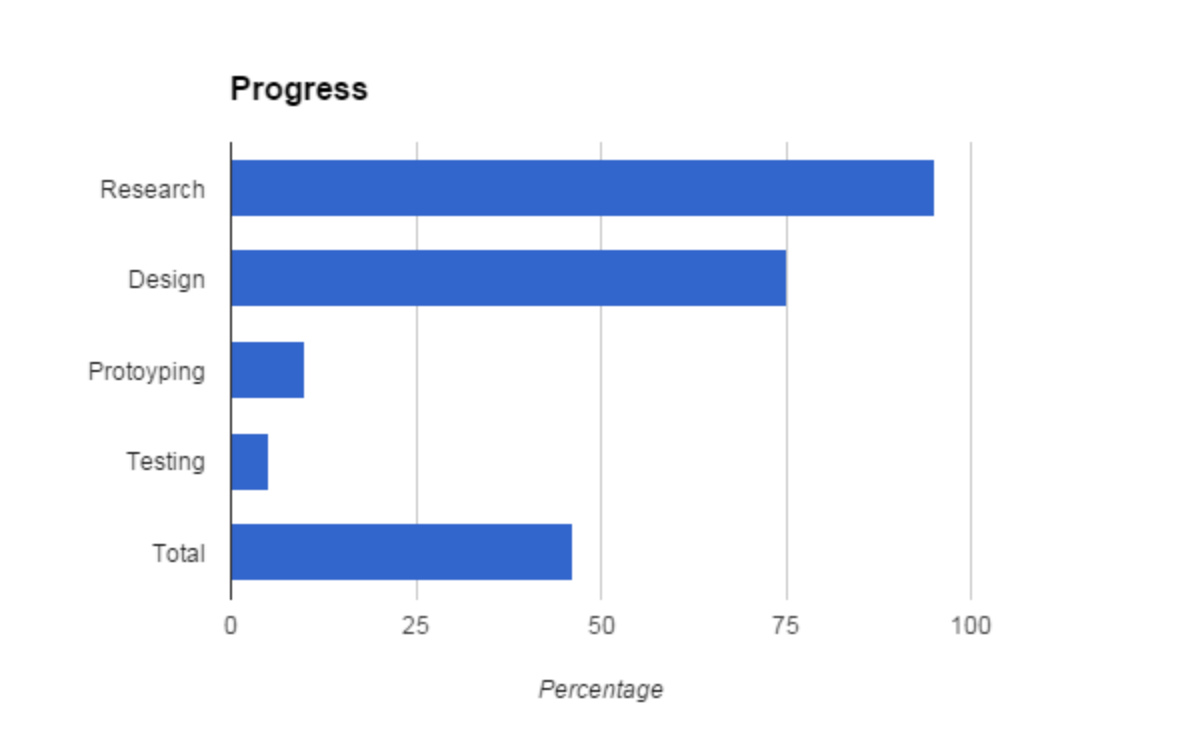
Build Budget

Part	Quantity	Unit Price	Total
Robot Chassis - Vex medium chassis	1	$21.35	$21.35
3.25 inch Vex Wheels	4	$19.99 (for four)	$19.99
Microsoft Kinect	1	$20.00 (Used)	$20.00(Used)
Ultrasonic Module HC-SR04 Distance Sensor	4	$8.99 (for two)	$17.98
VEX Bumper Switch	4	$12.99 (for two)	$25.98
ATmega328P	1	$3.70	$3.70
Raspberry Pi 2 Model B	1	$35.00	$35.00
Vex 393 Motors and Motor Controller 29	5	$24.98	$124.90
PCB	1	$66.00	$66.00
Edimax EW-7811UN WiFi transmitter	1	$14.99	$14.99
Tenergy 7.2V 5000mAh NiMH battery	1	$89.00 (set of two)	$89.00
Tenergy Battery Charger	1	$22.99	$22.99
Power Regulators components	N/A	N/A	$50.00
Grand Total			$511.88

Financing

- Project sponsored by Boeing
  - $580.11
- Partially financed ourselves

Progress



Progress
Research
Design
Protoyping
Testing
Total
0
25
50
75
100
Percentage

Plans to Complete the Project

Hardware:

- Finish ordering parts
- Order PCB
- Prototype power distribution
- Test components
- Build robot
- Build Charging Station

Software:

- Prototype each subsystem
  a. Mapping (Done)
  b. Navigation
  c. Motion Detection
  d. State Manager
- Integrate into full software system
- Integrate into full hardware system

Issues

- ROS
  - Extremely powerful and reliable
  - Inexperienced, have to learn from scratch
  - Have to rethink some systems to work with ROS
- Remote Access
  - Can the Pi 2 handle computing SLAM and other systems?
  - If not, what’s the most efficient way to transmit data and compute on a more powerful machine?

Issues

- Charging station interface with battery charger
- Heat dissipation
  - Will venting be enough?
- Battery level sensor for when to return to dock and to leave for patrol
- PCB Design - Design software doesn’t have all of the components, need to find libraries or alternative software

Questions?