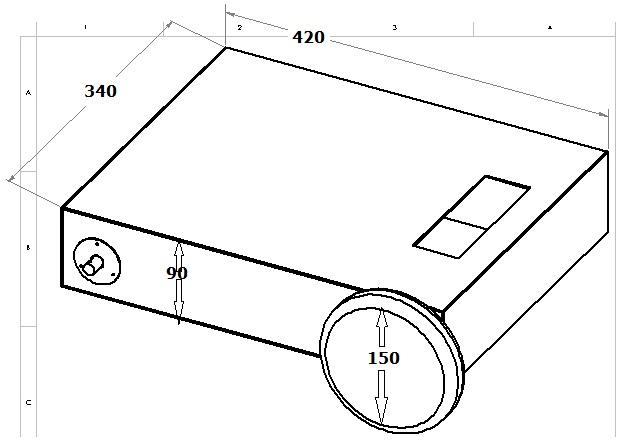
MYRO (myRAig Robotic Platform) is a Linux based mobile robot platform. Implemented with the latest robotics technologies, It provides easy integration for researchers, educators and developers. Myro robot platform is the ideal starting point for applications including research and development, industrial automation, security, and defense.

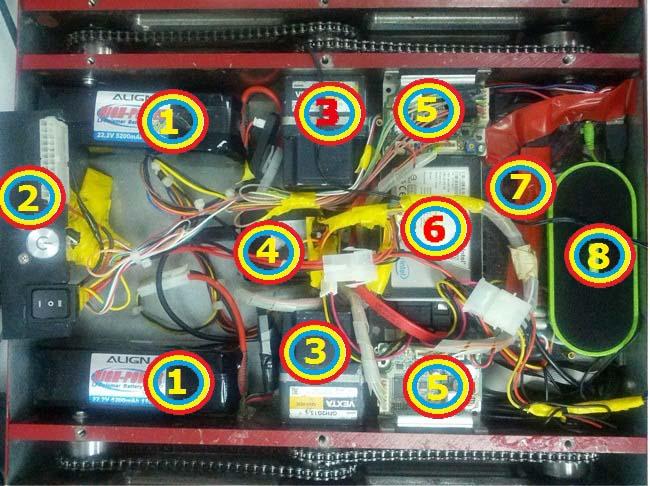
Myro classic is a four-wheel two-motor differential drive robot ideal for indoor laboratory or classroom use. The classic version is more focused on educational purposes but expandability of Myro enables us to modify it for a variety of applications.The Mayro's versatility, reliability and durability have made it the reference platform in malaysia for robotics research. Unlike hobby and kit robots, Myro is pre-assembled, customizable, upgradeable, and will last through years of tough classroom and laboratory use.

# Product Features and Benefits

* **Easy to Use** - Comes assembled and integrated with its accessory packages.
* **Myro software** - Myro's main software is complete open source code written in c++ for linux (Current OS is Ubuntu 12.04), a complete set of open source robotics applications and libraries that accelerate the development of robotics projects are already implemented inside software and are ready to use.
* **Customizable** - Easily accessorize by choosing from dozens of supported and tested accessories that integrate with the robotic platform. Additional help is available for future upgrades or added accessories. Myro classic have dozens of unused pots and IO’s for further expansion and customization.
* **Technical Support** - Myro’s software and hardware is completely designed and manufactured inside malaysia, comes fully documented with additional help available through our local product support team.
* **Reliable** - Construction is durable and rugged. Easily handles the small gaps, minor bumping, jarring,or other obstacles that hinder other robotic platforms.

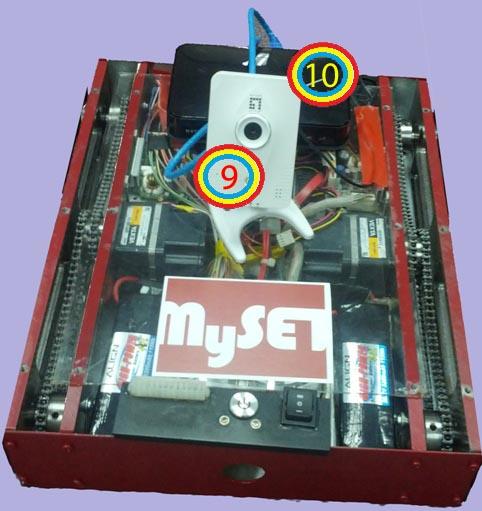
# Dimensions (mm) Top view





# Myro classic components: (refer numbers to top view picture)

1. **2 X Lipo rechargeable batteries:** provides 22.2V DC, 10400 mAh, runs the robot up to 3 hours, with 1 hour charge time.
2. **Main panel:** includes main switch for changing between operating, charging and off modes, Controller power push button switch, Charger 20 pin connector.
3. **2X brushless DC motors:** motors are 30W combined with a 1:15 gear provides enough power for up to additional 15 KG payload.
4. **Smart USB I/O**: 18 General purpose I/O, with analog input, analog output, pwm, UART and SPI connection.
5. **2X Brushless motor controller:** 24V brushless motor controller with protective functions to handle overload, overvoltage, undervoltage, overspeed and out-of-phase power.
6. **SSD Hard Drive**: current version of Myro is running on Ubuntu 12.04 Linux which is installed on this SSD hard drive, the system will boot into os and run Myro software in 40 seconds.
7. **MainBoard**: Main controller of Myro is a intel D510MO with dual core 1.6Ghz Atom processor and 2Gb of ram.
8. **Speaker**: enables Myro to play its speech synthesizer output,this speaker powers up through usb port.

**Front view** 

9. **Ip Camera:** Ip Camera transfer pictures to Myro through local network connection for image processing. Myro is using camera with 640X480 resolution ,30Fps in H 264 format. Using Ip Camera reduce encoding process from main controller and make myro camera easily accessible for remote controlling through its Wifi connection.

10. **Wifi Router:** supports 802.11N wireless lan standard,connects Myro its Ip Camera and enables user to connect to myro socket connection through Wifi and remote control it. Myro's socket protocol enable researchers to write their codes in any platform that supports TCP/IP connection and in any language that can access socket connections.

# Classic Myro Hardware Expandability

Myro with current controller based on an intel D510MO have 12 free GPIO through smart usb, 4 usb ports, 2 serial com ports, 1 parallel port and 2 free RJ45 Lan port which can support a large of additional sensors and other additional component.

# Optional Accessories supported by myro software are:

# • Laser-range finders • Mono- and stereo-vision cameras

# • Front and Rear SONAR • Robotic arms and grippers

# • Gyroscope and accelerometer • Segmented bumper arrays

# • Speakers and microphones • Joystick

# • Many more...

# Core Software

As described before Myro's core software is written in c++ and compiled in linux (ubuntu 12.04), this privilege enables developers to be able to compile Myro software on other linux operating systems (e.g Android) or other X86,AMD64 or ARM controllers, and embedded systems.

Core software is written multithreaded to be able to use power of new generation of multi core CPUs. Base software have 5 main threads which separates software task according to process power they need, core software threads are:

1. **Vision thread**, uses compiled and imported openCV open source image processing library, Current Myro classic software grabs frames from IP camera and easily enables developers to do their image processing routines and algorithms with openCV in this thread, examples of face recognition, ball tracking and pattern recognition are available for Myro classic .
2. **Communication Socket connection** **thread,** handles myro TCP/IP connection and communication protocol, gives full access to Myro IO, sensors, actuators and speech synthesizer. with this capability developers can remote control Myro or write their AI routines on any other operating system and language. samples of controlling myro with Microsoft Windows and Android are available for Myro classic. TCP/IP components like IP camera can be used directly through Myro router.
3. **speech recognition and speech synthesizer thread,** PocketSphinx, one of the best open source voice recognition libraries, and Festival one of the most common open source speech synthesizer libraries, are compiled for Myro and imported in its core software, these libraries are already initialized and ready to be used in this thread. samples of getting command with voice and reacting or answering with voice are available for Myro classic.
4. **IO and AI thread,** in this thread codes that are needed to read Myro Smart usb IO, com ports and other ports and controlling actuators like driving motors are already written and ready to use, This enables developers use easy commands to control Myro even without knowing hardware connections and structure. AI routines of Myro are prefered to be written in this thread.

Many other libraries are already compiled and installed on Myro like Sphinxbase which is needed for PocketSphinx, or FFmpeg and H.264 libraries which are used for OpenCV.