# HOMER - A High Speed Robot for Indoor Exploration

**Matt Rozyn and Nathan Wilson**
Undergraduate Mechatronics students at the University of Technology, Sydney (UTS)
njwilson@uts.edu.au and mkrozyn@uts.edu.au

**Michael Andrews**
Master of Engineering (Software Engineering) student at UTS
mickdo@pacific.net.au

**David Eager and Ray Clout**
Thesis supervisors, UTS

Mechatronics & Intelligent Systems Group, UTS
No. 1 Broadway, Ultimo 2007
Sydney, Australia

## Abstract

This paper describes the design and development of a high-speed, lightweight robot for indoor exploration; HOMER (High-speed Obstacle Mapping and Exploration Robot). HOMER is designed to rapidly explore an indoor environment and generate 2D maps, using a stereoscopic camera as the primary range sensor. The focus of the HOMER project is to provide a platform for semi-autonomous exploration and visualisation of unknown indoor environments. HOMER has been designed with an implementation of the Player software architecture, making it possible to use it as a test bed for future research into vision-based navigation and mapping.

## 1 INTRODUCTION

The primary motivation of the HOMER project is to develop a lightweight robot that can quickly be deployed to gather information about unknown indoor environments. HOMER can operate autonomously or under remote manual control via a wireless network connection. An on-board computer coupled to a stereoscopic camera, along with a suite of sensors (including two sonars) is used to gather information about the environment. An intuitive user interface has been developed so that the user is presented with a visualisation of the environment being explored and provides a means of controlling HOMER. A team comprising of two Mechatronics undergraduates and a part-time Master of Engineering student at UTS are developing HOMER.

Although small commercial mobile robots are available with capabilities similar to HOMER (most notably the Pioneer DX-2), these tend to be relatively heavy or very slow. These are general purpose platforms and are designed to be able to be able to support large attachments or payloads at reduced speeds. HOMER's primary design goal is to quickly explore and map an indoor space, travelling at up 2m/s. HOMER uses a light and compact stereoscopic camera as the primary range sensor.

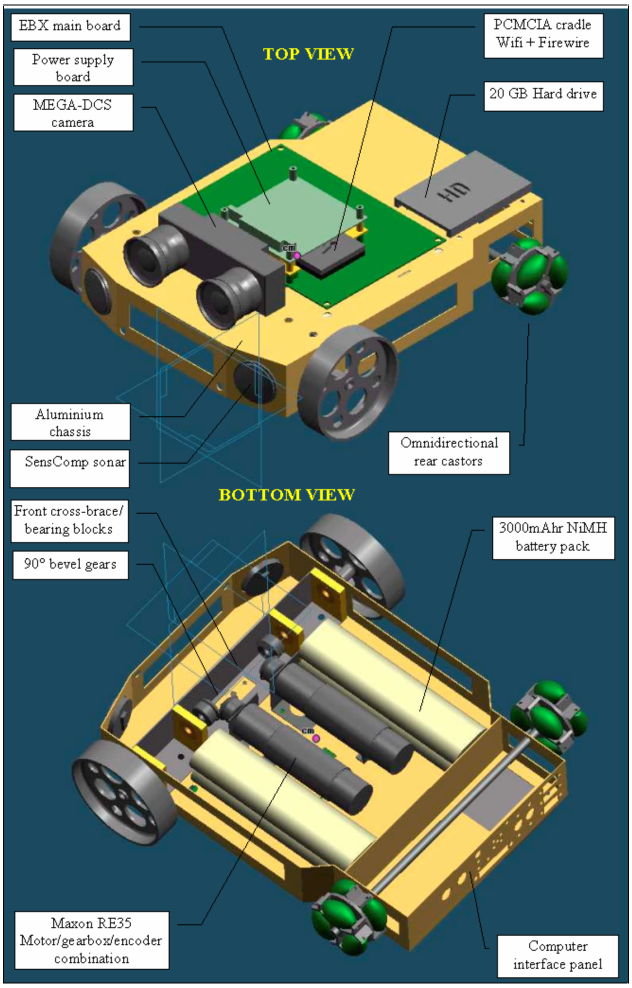


**Figure 1:** Isometric views of HOMER (Solid Edge 9)

HOMER uses the popular Player software architecture so that the algorithms currently being developed and implemented by many of our colleagues can be used on HOMER with few modifications. HOMER will be issued with full documentation, and is flexible enough to allow for added sensors and actuators. An important part of the project is to provide open standards and good documentation to make HOMER a viable and well-understood development platform for future students to work on.

The HOMER project is nearing completion and should be completed by late November.

## 2 MECHANICAL DESIGN

The mechanical design is often the most critical factor in determining the limits of performance of a robotic system. The main considerations that influenced the mechanical design of HOMER are the need for lightweight construction and high-speed operation.

### 2.1 Chassis

HOMER uses a conventional differential drive system for movement. Initially two chassis models were proposed. These were a conventional differential drive robot, and a car-type design with an Ackerman steering mechanism. Each design has its benefits and limitations. A car-type Ackerman design is more stable at speed, and has better traction due to an inherently better weight balance between the wheels. A differential drive robot can turn on the spot, is easier to control and fabricate, and has less moving parts.

After careful evaluation of the benefits of the two models, the differential drive option was chosen. This drive configuration gives the robot greater general-purpose usability with the reasonable cost of some reduction in maximum cornering speed relative to the car-type chassis. In addition, the kinematic model of such a robot is considerably simpler than a steered drive model, resulting in better dead-reckoning performance.

The chassis was manufactured from cut and welded 1.6mm aluminium sheeting, and uses a box section main body. The final chassis weighs only about 350g and is very stiff, with a maximised amount of flat payload area on top of the robot for future expansion.

### 2.2 Wheels and castors

Differential drive robots need a left and right drive wheel combination and a castor wheel system to provide balance and stability. To minimise weight it was decided that the front wheels would be machined from aluminium. Traction is provided by reversed rubber timing belts to give the wheel a serrated outer circumference suitable for surfaces such as carpet and wooden floors.

Swivel castor wheels were considered to support the rear section of HOMER's chassis, but it was decided that an omnidirectional castor wheel from Acroname (www.acroname.com) was a better option. The advantage of using such wheels is that they are able to roll forward and sideways at the same time. However, the wheels are somewhat heavier than a normal swivel castor.

**Figure 2:** Omnidirectional castor wheel

### 2.3 Weight

Keeping the weight of HOMER low is critical to maximising the speed and manoeuvrability of the platform. Excess weight in the chassis and peripherals has a flow-on effect to other components on HOMER; for instance the motor size selection and battery package capacity. The total design weight of HOMER is estimated to be around 6kg including the chassis, motors, batteries, sensors and EBX computer equipment.

### 2.4 Motor and gearbox selection

HOMER will be a test bed for future sensor systems and navigation algorithms that will allow faster and more reliable navigation of the indoor environment. HOMER uses motors and gearboxes that will enable the mechanical platform to keep up with these improvements. HOMER is designed to run at a top speed of 2m/s, with a maximum acceleration of $1m/s^2$. It is worth noting that the 2m/s target top speed is much faster than other comparable indoor robots such as the Pioneer DX-2, which struggles to reach 1m/s under most circumstances, with much slower acceleration.

The selection of motors and gearboxes for HOMER was a significant challenge. The three most important design factors were weight, power, and current draw.

Worst-case calculations for motor power requirements were done by modelling the load on the motors using a method suggested by Jones & Flynn (1993). This involved estimating the power required for HOMER to maintain a constant speed up an inclined plane of 30° at 1.5m/s on ordinary carpet. Motors and gearboxes were selected on, the diameter of the wheels and an approximation of power transmission losses it was estimated that the maximum required speed of the motor output shaft was 290rpm, and the maximum torque per wheel was 918mN.m.

HOMER's motors had to be light to achieve weight constraints, and needed to match or exceed the estimated torque and velocity requirements. Motors from many manufacturers were considered, including MicroMo (Faulhaber) and Portescap, but it was found that Maxon motors were best suited to our constraints of efficiency and size. Maxon's rare earth magnet motors (RE series) in particular have high efficiency (approximately 85%), low inductance, low inertia and low weight. We selected an 18:1 ceramic planetary reduction gearbox to match the speed of the motor to the speed of the drive wheels.

Using software supplied by Maxon we were able to quickly evaluate several models of the RE series motors for suitability, using power and torque requirement calculations. The Maxon RE-35 90 Watt rare earth magnet motors were chosen as the most suitable choice. The total length of the motor/gearbox combination is 125mm (including an integrated incremental encoder) with a diameter of 35mm. The RE-35 motors weigh approximately 500 grams each.

The current draw of the motor was minimised by using the highest possible voltage (29V max. possible from the battery pack) across the motor terminals.


**Figure 3:** Maxon motor and planetary gearbox (cutaway)

## 2.5 Battery and power consumption

HOMER is intended to operate under high accelerations and at high speeds, and is therefore expected to consume power rapidly. HOMER's battery supply is designed to last approximately one hour before the batteries need to be recharged, which while minimal should be sufficient for most research tasks. Trying to achieve this performance without an overly large and heavy battery package has been a significant challenge in the design.

It was calculated that under a worst-case scenario (see Section 2.4) the current drawn by the motors should be about 2.5A to 3A each. This figure was obtained by considering all the electrical systems on HOMER as a "black boxes" drawing current from a 29V supply. The expected current draw of the computer systems and electronics (more than 5.5A for an estimated 30W load) and the motors means that the battery supply could need to source approximately 7A for short periods of time when HOMER is operating under maximum load. Average power usage should draw significantly less than this.

Several rechargeable battery types were considered. Sealed lead-acid (SLA) batteries were too heavy and have relatively poor energy density compared to other types. Nickel-Cadmium (NiCd) batteries have excellent current-handling capabilities but unfortunately quite a low energy density. Lithium laptop batteries were also considered due to their excellent energy density, but were eventually rejected on the basis of cost and relatively poor high-current handling.

Remote-control car Nickel Metal-Hydride (NiMH) batteries were selected as the best compromise between energy density and current handling. The batteries that were selected provide 3000mAhr capacity in a 350g package, and can handle extremely rapid discharge or large currents without damage. Four of these batteries with a 7.2V terminal voltage were used in series to give a total terminal voltage of 29V. This high voltage is switched by a DMOS H-bridge to drive the motors directly, and also powers the computer and electronics power supply board.

# 3 ELECTRONICS AND SENSORS

## 3.1 Microcontrollers

Microcontrollers are extremely useful in robotics and most embedded applications. Two Atmel AT908535 8-bit microcontrollers are used to interface the main PC to HOMER's actuators and sensors. Both microcontrollers were programmed in 'C' using Codevision AVR.

The important features of this MCU are as follows:

- 8MHz clock speed
- 8kB in-system programmable FLASH memory
- 512 bytes of SRAM and EEPROM
- UART and SPI serial buses for communications
- 8-channel 10-bit ADC
- 3 counter/timers (TIMER1 can be used as a dual-output PWM signal generator)

One MCU is dedicated to the low-level control of HOMER's two drive motors. The other MCU provides an interface to the low-level sensors, such as motor current sensing and battery voltage measurement. Both MCUs use RS232 serial links to communicate with software running on the onboard computer (EBX).

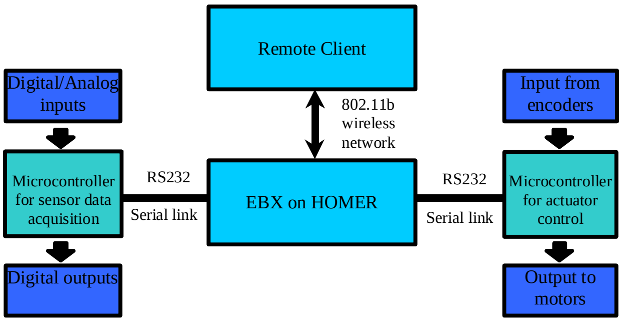


**Figure 4:** Microcontroller connections

# 4 SENSORS AND INTERFACING

## 4.1 Stereoscopic camera

HOMER's main sensor is a Videre Design MEGA-DCS stereoscopic camera (www.videredesign.com). The camera has an IEEE 1394 (Firewire) interface, and is capable of providing up to 30 frames per second at a 640 by 480 resolution. It is packaged with SRI International's Small Vision System (SVS) stereo analysis software. The SVS software provides an interface to the camera, as well as stereo processing libraries for post-capture operations. These libraries can be used to obtain a 'point cloud' from a stereo image. This point cloud is an array of 3D coordinates to represent an obstacles in the image.

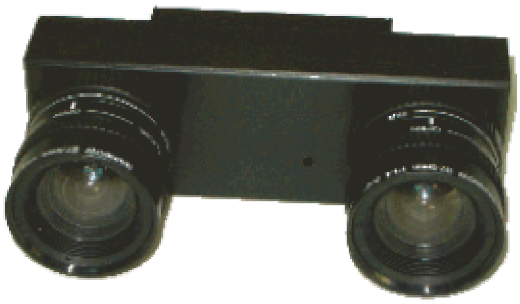

**Figure 5:** Videre Design MEGA-DCS stereo camera

### 4.2 Sonar modules

The MEGA-DCS camera and SVS software can fail to detect surfaces with poor contrast, such as the plain painted plasterboard walls found in most residential housing for example. At close range in a space with reasonable lighting conditions these surfaces can be detected, but this may not allow a fast-moving robot time to alter course or stop if a collision is imminent.

Two SensComp sonar modules are fixed to the front of the chassis, angled 15° away from each other to reduce beam interference. These sonar modules are based on the popular Polaroid 6500 units, but with several significant improvements. The interface board is integrated onto the back of the transducer, and a 5Hz auto-fire option can be set via a jumper on the back of the unit.

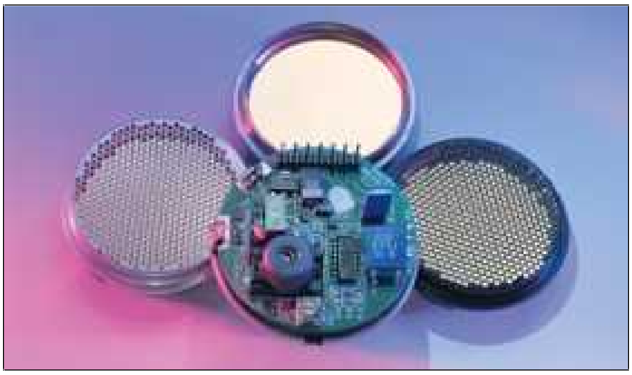

**Figure 6:** The SensComp series 600 SmartSensor (rear view showing the integrated interface board at centre)

### 4.3 Motor control and drive electronics

HOMER uses a dedicated Atmel microcontroller to control the velocities of the two drive motors. The Atmel samples motor velocity using the output of a quadrature encoder, calculates a correction using a PID algorithm, and applies the correction to the output going to the motor drive stage. The elements of the control loop are shown in Figure 7.

The low-level motor control software uses a simple PID (Proportional, Integral, and Derivative) controller design. HOMER uses a velocity control system, where the motor velocity is measured using encoders, and this information is used to calculate a change in the MCU's PWM (Pulse Width Modulation) output. The parameters for this model will be selected using a Matlab model and tuned for best performance.

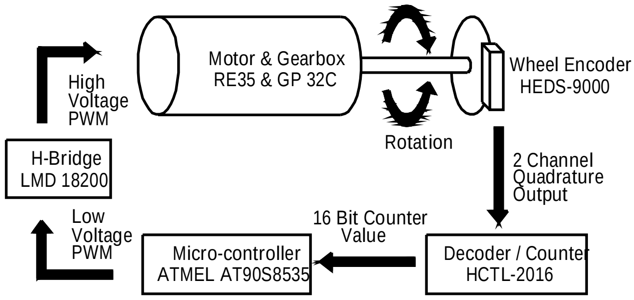


**Figure 7:** Flow diagram of HOMER's motor circuitry

Two incremental encoders attached to the rear of the two differential drive motors are used to measure speed and displacement (odometry). Absolute encoders were also considered, but rejected since they are considerably more expensive and have less resolution.

The encoders used on HOMER are HEDS-9000 units made by Hewlett-Packard. HEDS encoders are commonly used in robotics, plotters and photocopiers because they are small, high precision and low cost. The HEDS-900 is a quadrature encoder – it outputs two pulse trains with pulses exactly 90 degrees out of phase with each other. Hence a 500-line encoder wheel can have its resolution multiplied by 4 to give an effective 2000 clicks per revolution.

The dual output of the HEDS-9000 is read by an HCTL-2016 decoder/counter. The HCTL-2016 shifts time-intensive quadrature decoding and pulse counting functions away from the microcontroller to a dedicated IC. The HCTL-2016 chip was chosen due to its reasonably low cost and ease of interfacing the HEDS series of encoders.

The HCTL-2016 has an internal 16-bit up/down counter, which counts the number of (quadrature decoded) pulses that have arrived since the last reset. The 16-bit counter value can be transferred to a microcontroller via an 8-bit bus. The microcontroller must read the 16-bit number in two sequential bytes and recombine them in software. The counter value can then used to calculate the velocity of the motors.

HOMER uses a PWM (Pulse Width Modulated) system to control the motors. PWM uses controlled high-speed switching of the motor supply voltage to adjust its speed. This technique is well suited to digital control, and many microcontrollers have special hardware to create PWM signals. The PWM signal used on HOMER runs at approximately 16kHz to avoid generating audible controller noise.

The velocity of the motors is used as an input to PID control algorithm running in the Atmel microcontroller. The microcontroller calculates the appropriate correction, adds it to the current output and applies the changes.

HOMER's motors require an H-bridge to operate in both the forward and reverse directions. H-bridges operate by using pairs of switches to control the direction of current flow through the motor, allowing the motor to be run forwards and backwards.

The LMD18200D H-bridge by National Semiconductors was selected for HOMER's motor drive stage. The important technical specifications of the LMD18200D are as follows:

- 3A continuous, up to 55V drive voltage.
- DMOS power stage for high efficiency switching.
- Back-emf protection for driving inductive loads such as motors.
- Brake input for stopping motors quickly.
- Logic-level inputs (5V) can be used to switch drive-level outputs (30V). This enables the Atmel microcontroller to drive the motor without needing voltage amplification.
- Thermal protection circuitry, including automatic shutdown if overheating.
- Over-voltage and over-current protection.
- Current sense output (outputs a small voltage proportional to current in motor).

These chips are driven by a PWM input taken from the Atmel. The LMD18200D has an extremely low 'on' resistance and is virtually open-circuit when switched off, making it extremely power efficient. A small amount of power is consumed by switching.

### 4.4 Gyroscope

In addition to HOMER's incremental wheel encoders, rate gyroscopes will be used to increase the odometric dead-reckoning accuracy by providing a reference direction. Gyroscopes measure angular velocity using a variety of physical principles. One gyroscope is required for each of the axes along which angular velocity information is required.

There are several different styles of gyroscopes, the main ones being mechanical, piezoelectric and optical. Tokin CG-16DO piezoelectric sensors were chosen to measure the angular rate of rotation of HOMER in the x and y and possibly z planes. The Tokin gyroscope was selected due to its cheap cost and relatively high sensitivity (1.1mV/deg/sec).


**Figure 8:** Tokin GC-16DO piezoelectric gyroscope

### 4.5 System status sensors

All robots need some auxiliary sensors to monitor the state of the robot and to check for fault conditions. HOMER includes:

- Motor current sensors (via special output on LMD18200 H-bridge)
- Battery voltage measurement sensor
- Tip sensors
- Other future sensors (uncommitted IO)

These sensors are interfaced to a dedicated sensor MCU with the sonar modules (see Figure 4).

## 5 COMPUTER HARDWARE

### 5.1 PC on-board HOMER

#### 5.1.1 Main board

A VSBC-8 EBX embedded PC supplied by VersaLogic (www.versalogic.com) was chosen to be the on-board processor for HOMER. This board uses the standard desktop PC x86 architecture and so can be used with a conventional operating system such as Windows or Linux. The board is built to a military grade quality specification and is thus relatively well equipped to deal with the rough life on board a mobile robot.

The important technical specifications are as follows:

- Physical dimensions: 140 x 204 x 45mm, approximately 350g
- 850MHz CPU clock speed
- 256 MB SDRAM
- 8 x 12-bit ADC inputs, 16 bits of digital I/O
- PC/104 (ISA based) and PC/104-plus (PCI based) compact expansion buses, internal IDE and floppy
- 1 Parallel, 4 serial, 1 USB
- 27 watts power consumption from single 5V supply

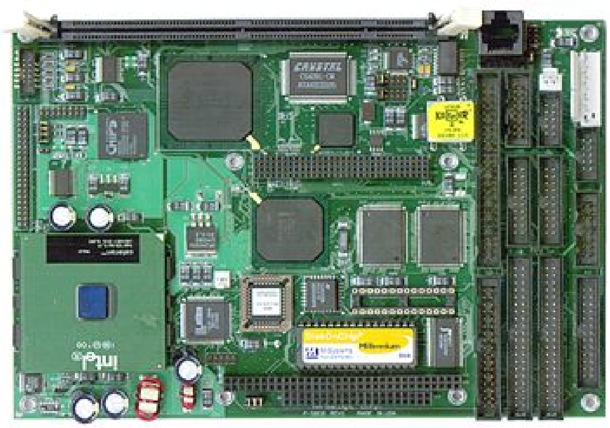

**Figure 9:** VersaLogic VSBC-8 single board computer

#### 5.1.2 Operating system

Red Hat Linux 9 was chosen as the operating system, since it is stable, easily scalable from desktop to an embedded system and is available as a free download. The Player 1.3 software (see Section 6.1) used to facilitate client-server communications will only run on Linux or other Unix-based operating systems, further limiting our OS options.

Red Hat Linux 9 is not a real-time operating system, so microcontrollers have been used to interface to low-level time-critical items such as the motors, where necessary. Our sensor/actuator software drivers on the microcontrollers communicate with the main PC via two serial ports.

All of HOMER's hardware works with Linux using open-source drivers and some drivers that ship with Red Hat 9 (for instance, Firewire is supported as a kernel module). A surprising range of hardware will work in Linux without any additional drivers, particularly PCMCIA cards, and often community technical support is quite expert.

#### 5.1.3 Non-volatile storage (hard drive)

HOMER uses a standard Fujitsu laptop hard-drive with 20GB of storage.

Solid-state hard drives were rejected early in the design process because they are generally very expensive and have little storage compared to a conventional hard drive. Since this robot is intended to function as a general research tool, it is also desirable to leave space for future software expansion, alternative operating systems and so on.

Some trial runs using HOMER's main computer on an autonomous wheelchair have revealed that the hard drive is vulnerable to crashing if it receives a severe shock (for instance during a high-speed collision). Steps have therefore been taken to provide basic shock and vibration mounting for our hard drive by isolating the drive using four felt pads. The high internal damping of the felt is expected to reduce the probability of a shock crashing the system.

#### 5.1.4 PCMCIA cradle

A 2-slot Prestico PCM-225 PC/104-plus board is used to provide an interface to our Firewire and wireless Ethernet PCMCIA cards (see Sections 5.1.5 and 5.1.6). The PCM-225 is a full 32-bit CardBus compatible board, which is necessary for the Sunix Firewire card. This board installed without trouble in Red Hat 9.

Using a PCMCIA cradle gives a designer a lot of flexibility to the system designer. A reasonable range of PC/104 and PC/104-plus cards are available, but the PCMCIA card is extremely popular and virtually any device required can be obtained. This includes hard drives, modems, wireless cards and high-speed peripheral buses such as USB-2 and Firewire.

#### 5.1.5 Firewire interface card

A Sunix 3-port Firewire card is used to interface our stereo camera. This unit is inexpensive and has proven more than adequate for our purposes. The card cost only a quarter as much as the cheapest dedicated PC/104-plus board.

#### 5.1.6 Wireless network communications

An Avaya WorldCard 802.11b wireless Ethernet PCMCIA card is used to achieve wireless communication with a remote client computer. HOMER is used on a conventional wireless local Ethernet, which uses a Proxim wireless access point to connect to a client running on the university LAN.

The 802.11b standard for wireless communications provides for a maximum of 11 Mb/s. This is of course under ideal or nearly ideal conditions. If significant noise or interference is present then the devices communicating will automatically lower their data rates to improve reception. The lowest data rate 802.11b will support (for instance if the signal must penetrate several walls) is 1Mb/s. If the signal degrades beyond this point then the signal will drop out completely rather than scaling to 0Mb/s.

#### 5.1.7 Power supply board

The computer systems are driven using a Diamond Systems Jupiter-MM board. This can deliver 8A on a 5V rail and 2A on a 12V rail. The EBX board (see Section 5.1.1) can require up to 5.5A without any expansion boards.

### 5.2 Remote PC

The remote client machine is a dual-monitor 2GHz Pentium 4 running Windows 2000 Professional. The first monitor contains the remote client GUI (see Section 6.3) and the second monitor is used to hold a VNC network[1] window for remote access to HOMER's desktop.

## 6 SOFTWARE

### 6.1 Player

The software architecture for HOMER is based on Player [Gerkey et al 2001]. Player is a device server designed to facilitate remote access to sensors and actuators mounted on a mobile robot. It is published as open-source software by the University of Southern California (USC) robotics lab[2]. Player is popular among many robotics researchers and has an extensive world wide user community. It has also proved very useful during the HOMER project, since Player elegantly handles the difficult task of network communication and allows more time to be spent on core activities such as writing device drivers.

Player 1.3 utilises client/server software architecture [Gerkey *et al* 2001]. The server side of Player, which runs on the robot, manages devices (actuators etc) and gathers sensor data information (sonar measurements etc) to transmit to a Player client.

The client side of Player runs on a remote machine with an ordinary network connection to the server robot. Most clients perform navigational and mapping functions and may provide an interactive user interface. HOMER uses two clients. The remote client provides a navigational and

[1] VNC: Virtual Network Connection – a way of getting remote desktop access to another PC over a LAN (Local Area Network)
[2] Player can be obtained from www.playerstage.sourceforge.net

mapping user-interface, while the local client acts as a basic failsafe mechanism to take control of the robot when the remote client loses the network connection.

The Player 1.3 system has been used to control HOMER remotely via a wireless Ethernet network. Player provides a mechanism for accessing a robot's devices via any TCP/IP network, and while the server software is only available on the Linux platform, clients can be implemented on any platform that supports TCP/IP.

## 6.2 Player server side software

Player's server side has been extended through the creation of new device drivers, which run on the robot and interface to the low-level hardware.

Three new Player devics have been created:

- *HOMERSensor:* Reads analogue and digital data sent from HOMER's sensor microcontroller.
- *HOMERVision:* Interfaces to MEGA-DCS via SVS driver.
- *HOMERPosition:* Interfaces to HOMER's mechanical subsystem to control motors and read encoders etc. Provides a high level API for the Player client to read and command the mechanical subsystems.

HOMERSensor and HOMERVision incorporate a serial port driver to allow them to communicate with the MCUs, whereas HOMERVision uses the SVS software to control and read from the camera.

The problem of unpredictable wireless network dropout is significant in the unstructured environments HOMER is intended to operate in. A small local client that works cooperatively with the remote client is the proposed solution to this problem.

If the remote client loses contact with the robot, a local autonomous client running on HOMER will take control to use a simple navigation algorithm to try to return the robot to a location where the remote client can reconnect. A lost connection will be indicated by the loss of a 'heartbeat' signal from the client.

## 6.3 Remote Software

The remote software Player client is a Microsoft Visual C++ application running on a Windows 2000 platform. The client has four main responsibilities:

- Retrieving the robot's sensor data and issuing control commands via Player.
- Mapping the robot's environment based on sensor information from odometry and the stereoscopic camera.
- Navigating the robot through its environment using one of several path-planning algorithms.
- Providing a Graphical User Interface (GUI) for the operator to observe the robot's surroundings and provide high level commands to the robot.

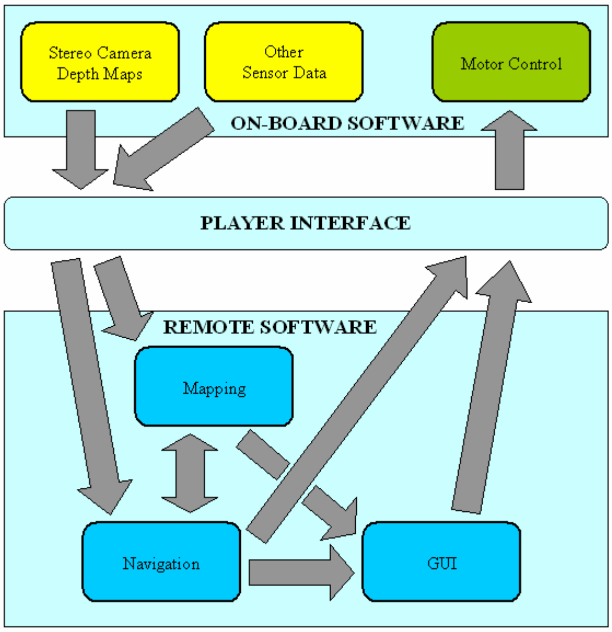


**Figure 10:** On-board and remote software architectures showing Player interface and information flow

### 6.3.1 Data Retrieval and Robot Control

The client software system implements a Player client interface to retrieve data from the robot and issue it commands. Upon initiation the client application automatically connects to the robot server via the Player interface, and begins polling the server for new data. As new data is obtained, the client forwards this information to its Mapping and Navigation subsystems for processing.

The remote client can control the robot via the Player interface. Commands are sent to HOMER when the Navigation subsystem plans a new route for the robot to take, or when an operator enters a command manually via the GUI (see Section 6.3.4).

### 6.3.2 Mapping

The Mapping subsystem receives sensory data from the robot's stereo camera in the form of stereo depth maps. These depth maps provide a set of 3D coordinates that indicate the presence of detected objects. HOMER creates a 2D representation of its environment (length and breadth, but not height), so the 3D data is filtered to eliminate any points that are above or below the robot's vertical extremities. The remaining 3D points are converted to 2D information, and the data is stored in an occupancy grid. A two dimensional array of occupancy probabilities within 'cells' is used to represent these locations in the environment. The occupancy probability of a cell determines whether the cell is classed as having an obstacle within it, is free of obstacles, or is of an undetermined state.

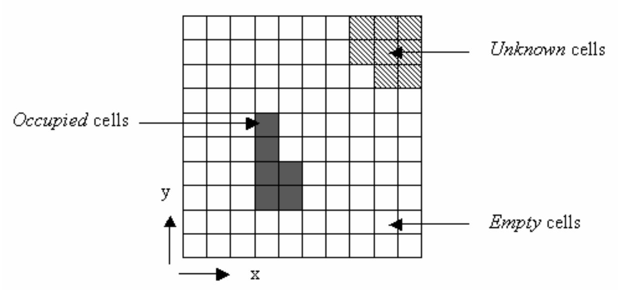


**Figure 11:** Graphical representation of an occupancy grid

### 6.3.3 Navigation

The Navigation subsystem maintains the current position of the robot, and is responsible for determining its future movements. Using the occupancy grid information from the Mapping subsystem as a reference, the Navigation subsystem utilises path-finding algorithms to calculate the best route that HOMER should take through its environment.

Several path planning methodologies have been investigated for HOMER, focussing on the A* [Nilsson, 1980] and D* [Stenz, 1994] algorithms. Both methods work by breaking down an environment into a series of interconnected nodes, to try to determine the best path from a designated start location to a designated finish location. HOMER will use the D* algorithm for planning paths in unknown environments, while the A* algorithm will be used in known environments.

#### 6.3.3.1 A* Algorithm

The A* algorithm is widely used today as a path finding solution in many strategy computer games. It was developed to incorporate formal methods with heuristic methods (those that involve knowledge about the problem itself) into the one algorithm.

The algorithm begins at a start node, and then processes each of its neighbouring nodes, maintaining their traversal costs back to the start node. It also maintains a heuristic estimate of the traversal cost from the node to the finish node. This process is continually repeated, expanding the node that has the lowest combined start traversal cost and heuristic goal cost, until the finish location is eventually reached.

A* also keeps track of those nodes that have been examined, but not yet expanded. This allows the algorithm to 'backtrack' if a particular search path results in a dead-end. The heuristic can be adjusted to modify the algorithm between computational speed and path accuracy.

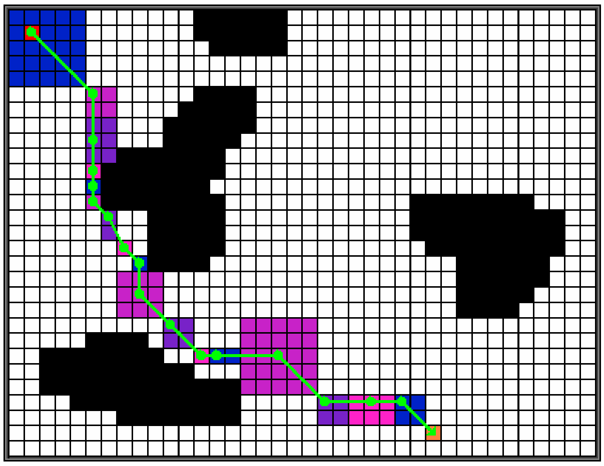

**Figure 12:** Sample Planned Path (A*)

#### 6.3.3.2 D* Algorithm

While the A* algorithm is very efficient in finding a path solution for a small search area, it can become computationally expensive in larger environments. A* can also fail when the state of the environment is not entirely known, as it requires complete recalculation whenever new information is discovered. In order to combat these deficiencies, the D* algorithm (Dynamic A*) was developed from A*.

The D* algorithm is well suited to robotics, as it stores the state of the environment, and when a robot's sensors discovers new obstacles, usually only the area affected by the changes will need to be updated.

### 6.3.4 GUI

HOMER's client user interface provides detailed information about the current state of the HOMER system, as well as allowing user interaction. The interface is structured in the form of multiple dockable windows, each with their own functional purpose. These windows include the following:

- A Map view, which displays a detailed map of the detected terrain surrounding the robot, and allows the user to choose the robot's destination graphically.
- A Control dialog, for issuing commands to the robot, and displaying basic telemetry data.
- A Logging dialog, which displays important data that the application has recorded.
- A Vision dialog, which displays stereo disparity information from the robot.

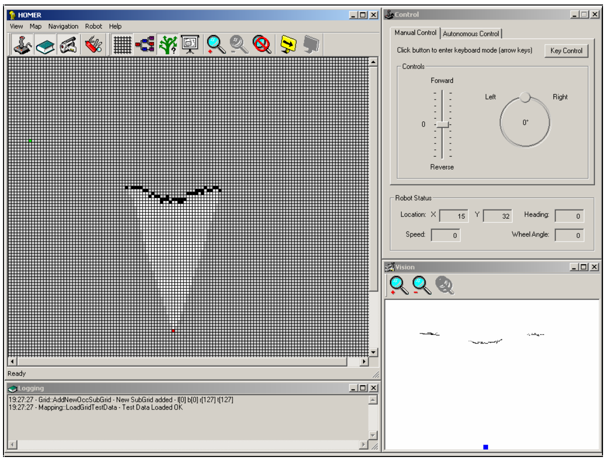

**Figure 13:** Graphical User Interface

## 7 CURRENT PROJECT STATUS

The design presented in herein is currently being built and tested. So far the project is on schedule for completion at the end of November. It is anticipated that HOMER will be demonstrated at the ACRA conference.

Figure 14 below illustrates the current status of HOMER. The mechanical design and assembly is two weeks from completion. The motor controller and sensor boards have been built and are being tested as this report is written.

The software design and development is well advanced. Prototype versions of the GUI, occupancy grid map building, implementation of A* and D* are all complete. Software for the microcontrollers for motor control and data acquisitions is currently being tested. However, much remains to be done, in particular the software integration and microcontroller communication protocols.

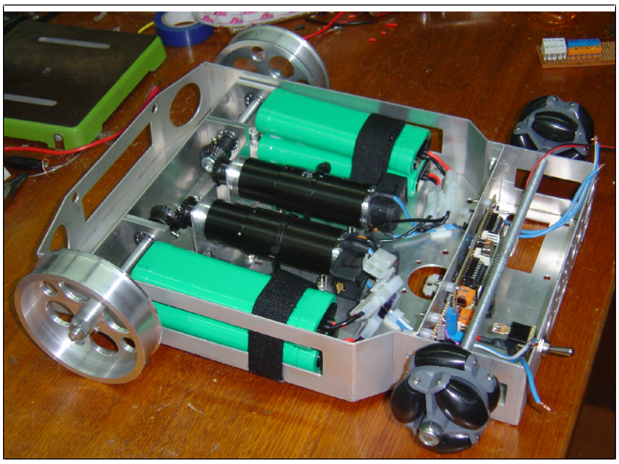

**Figure 14:** HOMER chassis showing mechanical underpinning

## 8 CONCLUSIONS

A design for a high-speed small indoor mobile robot, HOMER, has been presented. This robot has been manufactured and assembled and is currently in the final phases of testing before completion. Remote sensing and control of HOMER has been achieved using Player, and the client GUI is successfully receiving data from the stereoscopic camera and mapping the results in 2D. Implementation and testing of high level mapping algorithms has commenced and will be an area of ongoing research. Future applications for HOMER include being used for research into robot search and rescue, vision system map building and large-scale collaboration research.

## 9 ACKNOWLEDGEMENTS

The UTS node of the ARC Centre of Excellence for Autonomous systems has funded this work. The support and guidance from our supervisors Gamini Dissanayake, David Eager and Ray Clout is gratefully acknowledged.

## 10 REFERENCES

[Gerkey *et al* 2001] B.P. Gerkey, R. T. Vaughan, K Støy, A. Howard, G. S. Sukhatme, and M. J. Mataric. *"Most Valuable Player: A Robot Device Server for Distributed Control"*. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2001), pages 1226-1231, Wailea, Hawaii, October 29 - November 3, 2001.

[Jones and Flynn, 1993] Joseph L. Jones and Anita M. Flynn. *Mobile Robots – Inspiration to implementation*. AK Peters, Wellesley, Maryland, 1993.

[Borenstein and Koren, 1990] J. Borenstein and Y. Koren. *Real-time obstacle avoidance for fast mobile robots in cluttered environments*. IEEE international conference on Robotics and Automation, Cincinnati, Ohio, May 13-18, 1990.

[Borenstein *et al* 1999] J. Borenstein, H. R. Everett, and L. Feng April *"Where am I? Sensors and Methods for Mobile Robot Positioning"*. University of Michigan.

[Nilsson, 1980] N. J. Nilsson. *Principles of artificial intelligence*. Tioga, Palo Alto, California, 1980.

[Stenz, 1994] Anthony Stenz. *Optimal and Efficient Path Planning for Partially Known Environments*. Proceedings of the IEEE International Conference on Robotics and Automation, May 1994.