Scott Kanowitz, A. Antonio Arroyo

Machine Intelligence Laboratory
Department of Electrical and Computer Engineering

332 Benton Hall
University of Florida
Gainesville, FL 32611

Email: skano@mil.ufl.edu

Development Of An Autonomous All-Terrain Vision Research Platform

Scott M. Kanowitz, Dr. A. Antonio Arroyo, Dr. Eric M. Schwartz
Machine Intelligence Laboratory
University Of Florida

ABSTRACT

The MIL Rover is a platform designed to test autonomous vision research and control. The Rover maintains an open operating environment for simple access to code changes and upgrades. The mechanical design enables performance on a variety of previously unreachable terrain for sampling “real world” images. The main code contains routines for simple control of actuation allowing easy development of image processing feedback. The entire Rover design is upgradeable for future project developments.

PHYSICAL SYSTEM

Electrical Subsystem

The Rover control suite consists of a 486SX uP, 68HC11 uP and motor control board. All high-level control decisions are processed using the 486SX uP, while low-level control is performed with the 68HC11. All motor control is performed using the 68HC11 via the motor driver board. The sensor suite of the Rover is controlled with the 68HC11, except for the color camera, which is directly interfaced to the 486SX uP. The design hierarchy is illustrated in Figure 1. The 486SX uP is part of a ZF Micro Systems 486SX 104 Card/EV embedded controller system (Figure 2). This controller contains all the necessary elements of a larger personal computer, enabling ease of use in the development of applications. Attached to the controller via the IDE bus is a Fujitsu 2.1 GB notebook hard drive. Together, the system operates on the Red Hat Linux 5.2 operating system.

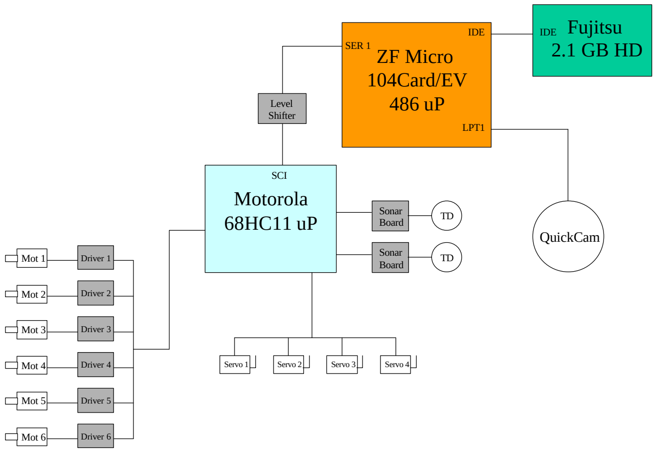

Figure 1. Rover Design Hierarchy

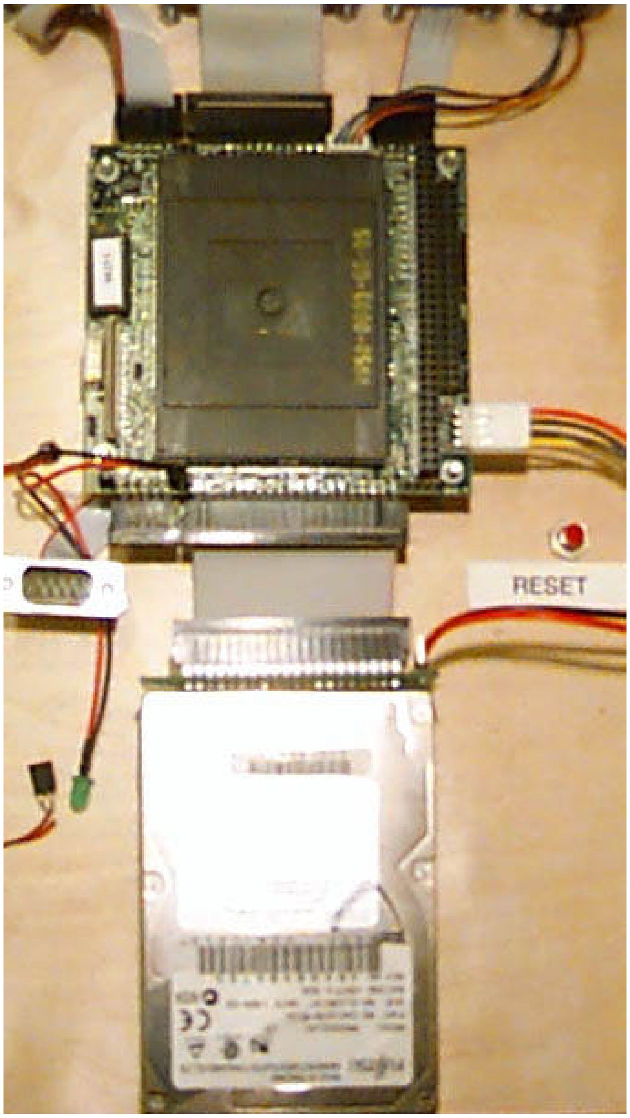

Figure 2. 486 Embedded Computer

Communication with the 68HC11 uP is performed using the serial port (COM 1). The controller board housing the 68HC11 is the MSCC11 (Figure 3). This board enables low-level control of all subsystems in the Rover as dictated by the 486 embedded computer. Sensor feedback, excluding the color camera, is gathered with the 68HC11, and directly delivered to the 486 embedded computer for processing. The motors' speed and direction are controlled via the motor driver board through pulse width modulation (PWM) and digital I/O provided by the 68HC11. The four steering servos are controlled in a similar fashion. The main sensor of the Rover is a Connectix color camera. This sensor provides high bandwidth data of the environment for decision processing. The remaining sensor suite, used for short-term navigation and distance measurement, includes two sonar transducers.

Figure 3. MSCC11 Board

Mobile Platform

The Rover's mechanical structure is modeled after the Jet Propulsion Laboratory's "*Mars Pathfinder.*" This design has been proven on Mars, and allows the Rover to negotiate the "real world's" rough terrain. The platform consists of two legs, each containing three wheels (Figure 4). Each leg is attached to a main controller box via an aluminum shaft and adjustable balance rods. Each wheel on the leg assembly is actuated using a Micro Mo 3557K018CR + 34PG 90:1 motor. These motors deliver high levels of torque while operating on less than one ampere. The motors are located inside the wheel assembly and are attached directly to the leg. The wheels are covered with rubber foam for added traction. The leg assembly is entirely aluminum. Two joints on the leg enable it to shift for climbing objects using all six wheels, however the Rover can operate with any four wheels contacting the ground. The front and middle wheels are attached to the same leg joint, while the rear wheel is isolated, enabling the wheels to climb up to five radii high. A shock is used to maintain stability and contact of the rear wheel.

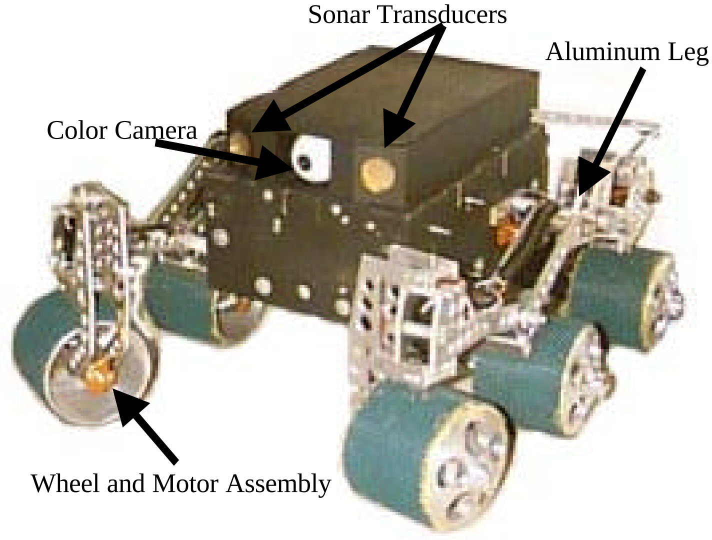

Figure 4. Leg and wheel assembly

The main controller box assembly rests on an aluminum shaft that travels through the center of the body. The shaft attaches to the center of the legs and provides front and back rotation to preserve heading while on unstable terrain. The stability of the main controller box is maintained using adjustable rods connected to the back of the box and middle of the legs.

Adjusting these rods changes the pitch of the main controller box.

Actuation

The Rover is driven with six direct drive motors controlled by the 68HC11 via the motor control board. A DPDT relay and transistor system is used to provide high current speed and direction control of the motors. Speed control is achieved using PWM signals from the 68HC11 to switch the gates of TIP120 transistors. Direction control is performed through the coil of the relays. Steering is achieved using four Cirrus servos each actuating a single front and rear wheel. The servos are able to turn 110^0 to allow a wide range of steering options. The rover is capable of forward motion, turning on a pivot point and moving side-to-side. Control of the servos is performed through PWM via the 68HC11.

Sensors

The sensor suite of the Rover consists of a color camera and two sonar transducers (Figure 4). The color camera provides color image data for processing. The camera is interfaced through the parallel port (LPT 1) of the 486 embedded computer. Most control decisions of the Rover are developed solely from the data gathered by this sensor. The sonar transducers, mounted on opposite sides of the camera, are used for object avoidance and distance measurement. The sonar control system is interfaced with the 68HC11 to allow for digital distance calculations. Decisions regarding base speed and direction rely on the sonar system. Together these two sensors provide a pseudo two-dimensional image-processing platform. While two cameras would provide absolute distance to an object, the sonar ranging system gives a relative distance to the closest object in view. This design is suitable for current project goals.

SOFTWARE SUBSYSTEM

The software subsystem is comprised of two main components, the code for the low-level control system and the code for the 486 embedded computer. The code for the low-level control is designed completely in assembly language for execution on the 68HC11, allowing control over the exact timing of all signals. The code enables the 68HC11 to generate a variety of PWM signals while only using three or four timers. The timer interrupts are multiplexed and used to control up to six pins, however, up to ten pins can be accommodated. Four PWM signals are used to control the servo positions and six are used to control each motor's speed. The input capture ports on the processor are interfaced with the sonar. A timer is started when the sonar is enabled, and an interrupt is generated when a return signal is received. The time duration is then used to calculate the varying distances. When the 68HC11 is powered up the software initializes all systems into a wait state, holding

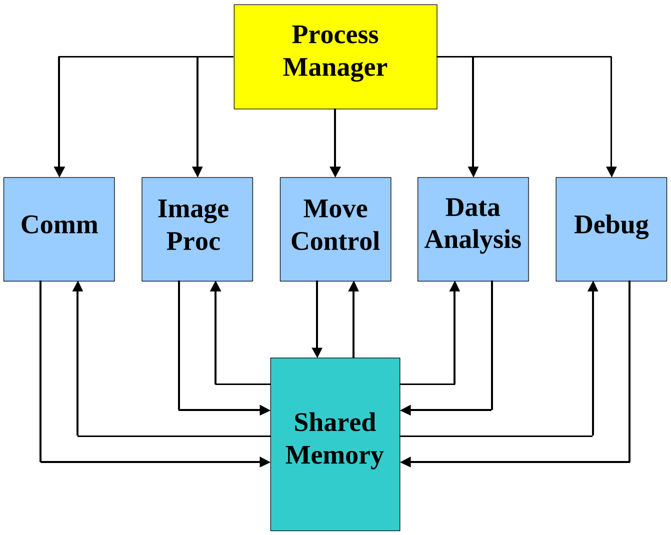

Figure 5. High-Level Software Diagram

until control commands are received from the high-level processing system. A string set, terminating with a carriage return, instructs the low-level code to perform a desired operation.

The code on the 486 embedded computer is being developed in the C programming language for the Linux 5.2 operating system. The code consists of six modules, all of which are individual processes (Figure 5). The main executable is a process manager that initializes memory space and executes the requested modules. The communication module is necessary for control of the serial port to send command strings to the 68HC11. It continually checks a send queue to determine if there are any commands waiting to be distributed. All returned strings are stored in a received queue for processing. The image processing module requests frames from the color camera and processes the data according to a predetermined algorithm. The results from the process are then sent to the other modules via a shared memory space. The movement control module keeps precise records of the current states of all the actuators. This data is used to determine what control is necessary to move to a desired position. The data analysis module will contain the primary functions for all of the data processing. This module enables autonomous control of the entire system. All the other modules contain the data input and output functions to provide the data analysis module with all the necessary parameters. A debug module is included in the design, enabling human intervention into the control of the platform. The debug module provides a screen image of all the data gathered by the system, and allows an override of any control parameters.

CONCLUSION

The Rover provides a reliable platform for autonomous vision research. The ability of the platform to negotiate rough terrain enables the Rover to operate in the real world, away from a laboratory setting. This ability allows for gathering of substantial image data for processing and navigation. The high bandwidth sensors on the Rover allow for high-level and high-speed computing provided by the Rover's internal processing and control structure. The final platform will provide a reliable testing ground for current developments in image-processing algorithms.