Proceedings of the 1991 American Control Conference
June , 1991, Boston, MA

Real Time Control of the MIT Vehicle Emulator System

William K. Durfee, Husni R. Idris and Steven Dubowsky
Department of Mechanical Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139

INTRODUCTION

The MIT Vehicle Emulation System (VES) is an experimental facility designed to facilitate the study of controlling robots fixed to nonstationary bases. This includes assembly and repair manipulators attached to space vehicles, or to the space shuttle arm, as well as manipulators fixed to the bed of moving ground vehicles. Controlling manipulators under these conditions for endpoint force and positioning tasks requires both the development of new theory, and the development of a test-bed for verifying theoretical results through experimentation. The concept, operating principle, basic design, and basic control principles were realized in the first generation VES system, completed in 1988 [1, 3, 4, 6]. This paper describes the technical details of the real-time controller for the second generation VES system which is under development at MIT.

The VES is a six DOF position-controlled platform which mimics the motions of the vehicle being emulated. A manipulator is bolted to the top of the platform through an in-line, six DOF force sensor. The force sensor detects the inertial and gravitational forces reflected to the base of the manipulator caused by motions of the manipulator links, and by the interaction of the manipulator endpoint with the external environment. The platform is then servocontrolled to match the motions of the emulated vehicle if that vehicle were to be subjected to the same forces exerted through the base of the manipulator.

The design of the mechanical system was dictated by the platform payload and motion requirements, which include the capability of moving up to 1000 pound loads through a workspace of @Math[@PM]12 inches in each of three linear directions and @Math[@PM]30 degrees in three rotations at frequencies of up to 10 Hz. The VES mechanism is based on a Stewart platform configuration, where the moving platform is connected to ground through six prismatic actuators acting in parallel [2, 5]. Stewart platform mechanisms are also seen in high-performance, motion-based flight simulators. The VES servoactuators are hydraulic cylinders (Model A86, Moog, Inc.) with a working range of 30 inches. The actuators are driven by a 1500 psi hydraulic supply which can deliver a steady-state flow of 40 gallons-per-minute. The position of each leg is sensed by a Temposonics linear position transducer (MTS Systems Corp.) which is integrated inside the hydraulic servoactuator. This sensor transduces actuator position into a pulse-width-modulated logic signal with the width of the pulse being proportional to piston position.

The computer-based, real-time control system for the VES was designed to enable accurate emulations of vehicles modeled by linear or nonlinear dynamics with a 0 to 10 Hz bandwidth. The design specifications also mandated that, (1) the system be safe for both the operators and the platform, (2) the computer-control architecture be expandable, and include the capability to control manipulators mounted to the platform, (3) the operator interface be simple yet have the capability to run a wide range of experiment capabilities, and (4) the system meet budget constraints.

CONTROL SYSTEM COMPONENTS

Figure 2 shows the layout of the computer-control hardware. A host computer uses a network connection to communicate with two single-board slave computers configured on a VMEbus system. I/O cards on the VMEbus handle data acquisition and control commands to and from the platform. The VxWorks (Wind River Systems) real-time operating system manages task generation and synchronization, and inter-processor communication.

The host processor is a SunStation 3/80, a Unix-based workstation with a 68030 CPU and a 19" display. A "point-and-click" user interface runs on the host. The interface was built with software tools provided by the X Windows System and OSF Motif. The network connection from the host to the slave computers is through a standard 10 MHz Ethernet. The network communication is dedicated to supervisory control commands to configure, start, and stop a VES experiment, and to pass information on the status of an experiment to the operator. All real-time control is implemented on the slave processors.

The system contains two slave processors, a 20 MHz, 68030 CPU board (HK68/V30XE, Heurikon Ssytems, Inc.), and a 20 MHz 68020 board (HK68E/V20, Heurikon). Both boards contain 68881 floating point coprocessors. Communication from the VMEbus to the sensors and actuators on the platform is through a custom circuit built on a VMEbus prototyping card (XVME-085, Xycom, Inc.). The platform communication card (the "LegHost") in turn connects to six custom circuit boards (the "LegSlaves") which handle closed-loop position control of the servoactuator legs. Each LegSlave board contains a 12 MHz, 8031 8-bit microprocessor which runs a 16-bit digital PD control algorithm with a sampling period of 1.02 milliseconds. The boards also read the pulse-width modulated output of the Temposonics position transducers. The output of the PD controller is converted to an analog command signal through a 16-bit DAC, and then amplified to a controlled-current signal which is sent to the coils of the actuator servovalve. Figure 2 contains a block diagram of the LegSlave control board.

Forces at the base of the manipulator are transduced by a six degree-of-freedom force sensor (OR6-5-4000, AMTI, Inc.), which has a 4000 pound vertical load capacity. The load capacity of the force sensor was chosen based on the requirement that it must handle the large moments generated by fast, rotational motions of the manipulator.

The expected sensor force and moment loads were computed through manipulator simulations, and the transducer selected accordingly. The six channels of force sensor strain gage signals are amplified (1B31AN, Analog Devices), and then digitized by an 16-bit, differential input, VMEbus compatible ADC board (DT1401, Data Translation).

CONTROL SOFTWARE

The real-time software to control the platform, and to perform continuous safety checks is structured as a number of separate tasks which run on the two slave processors under supervision of the VxWorks operating system. The tasks are best explained through an example of a specific experiment. Assume that the vehicle to be emulated is a large mass which can move in six DOF, but is coupled to the ground through six independent spring-damper suspension systems, one for each degree of freedom (a "linear MKB" vehicle). The tasks to control this experiment are listed below.

Wait for sample.

The sampling period for the vehicle simulation is set by the operator. This task waits for the next tick of the sample clock.

Read forces.

The six channels of force information are sampled, and the results multiplied by a 6x6 matrix to correct for sensor cross-talk.

Compute model.

The vehicle to be emulated is structured as an admittance model which, at each sampling instant, takes a six element applied force vector as input and computes a six element desired platform position vector as output. The model computations require integration of a set of coupled, first-order equations which are linear for simple systems, but nonlinear for complex systems.

Inverse kinematics.

The desired platform position vector is transformed to a set of six desired leg lengths based on the Stewart platform geometry.

Safety checks.

The desired platform positions are checked against the platform workspace constraints, and the desired leg velocities are checked against the flow-rate limits of the hydraulic servoactuators to determine if the platform is capable of moving to the new position. The desired leg positions are also compared to the last set of measured leg positions, and the experiment aborted if there is an unexplained discrepancy. This check confirms the proper operation of the distributed, PD controllers for each leg.

Leg communication.

The set of desired leg positions are sent to the six LegSlave control boards, and a set of six measured leg positions are received from the LegSlave boards. The LegSlave boards use the desired leg position as the reference input to the PD controller.

The tasks are synchronized through VxWorks semaphores to run in sequence. The first three run on one single board slave computer, and the last three run on the second. This effectively doubles the peak sampling rate since the two sets of tasks run in parallel. For emulation of a linear MKB vehicle, the minimum sampling period is 4 milliseconds, which meets the 10 Hz vehicle bandwidth specification easily. Emulation of nonlinear vehicles, however, increases the minimum sampling period.

ACKNOWLEDGMENT

This work is supported by NASA (Langley Research Center,Automation Branch) under Grant NAG-1-801.

REFERENCES

[1] Dubowsky S, I Paul, and H West. An analytical and experimental program to develop control algorithms for mobile manipulators, *Proceedings of the 7th CISMIFToMM Symposium on Theory and Practice of Robots and Manipulators. 1988.*

[2] Fichter EF. A Stewart platform-based manipulator: general theory and practical construction, *Int. J. Robotics Res. 5(2):157-182, 1986.*

][3] Ismail AN. *The Design and Construction of a Six Degree-of-Freedom Parallel Link Platform Type Manipulator,* Master's thesis, Massachusetts Institute of Technology, May, 1988.

[4] Stelman NM. *Design and Control of a Six Degree-of-Freedom Platform with Variable Admittance,* Master's thesis, Massachusetts Institute of Technology, May, 1988.

[5] Stewart D. “A platform with six degrees of freedom”. *Proc. Inst. Mech. Engr. 180(1):371-386, 1965.*

[6] West H, N Hootsmans, S Dubowsky, and N Stelman. “Experimental simulation of manipulator base compliance”. *Proceedings of the 1st International Symposium on Experimental Robotics. 1989.*

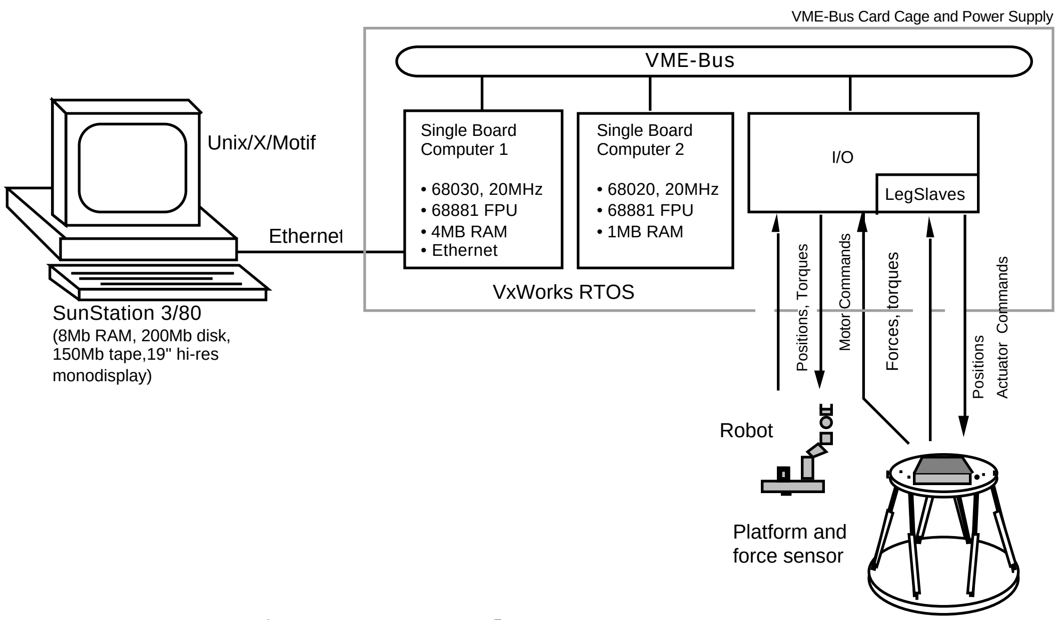

Figure 1: Control System Components

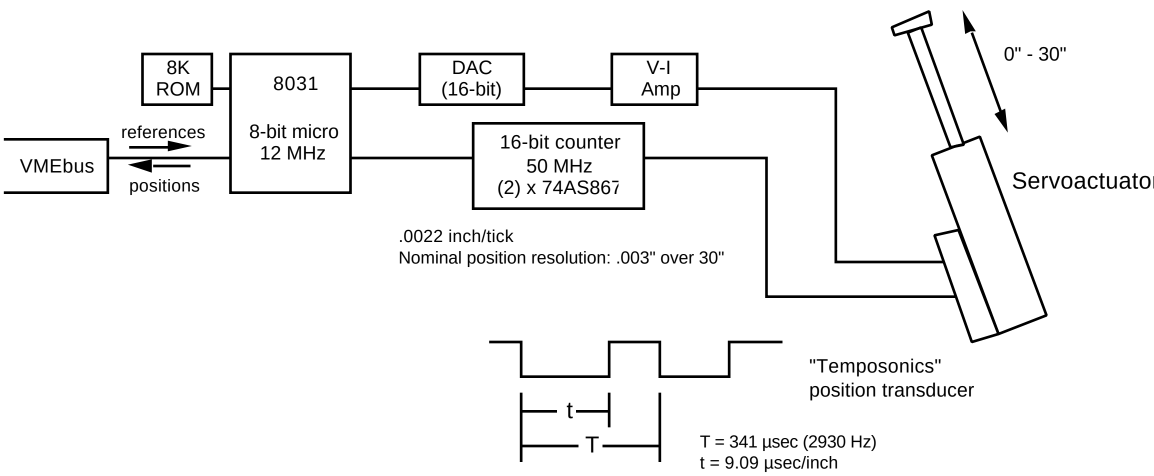

Figure 2: LegSlave PD control board