'90 KACC 1990. 10. 26~27

The Teleautonomous Control of an Integrated FRHC-PUMA Telerobot Control System

Jin S. Lee
Department of Electrical Engineering,
Pohang Institute of Science and Technology

Edwin P. Kan
Jet Propulsion Laboratory
California Institute of Technology, Pasadena, Ca., USA

Abstract

The system discussed in this paper is an integrated stand-alone system with the full functional capabilities required of a telerobot system. It is complete with a force-reflecting 6-DOF hand controller, driving a PUMA 560 or 762 robot, with an integrated force-torque sensing wrist sensor and servo-driven parallel jaw gripper. A mix of custom and standard electronics, distributed computers and microprocessors, with embedded and downloadable software, have been integrated into the system, giving rise to a powerful and flexible teleautonomous control system.

1 Introduction

The telerobot system discussed in this paper was developed at the GE Advanced Technology Laboratory under contract from the Jet Propulsion Laboratory. It was developed for the Goddard Space Flight Center (GSFC) to respond to the needs for an integrated laboratory system that could be used as a baseline to compare with other telerobot systems available from the market or research community.

This system is a teleautonomous system, capable of force reflecting pure teleoperation at one end of the spectrum, and capable of fully autonomous force compliant control on the end of the spectrum. A linear blend of the two ends of the spectrum can produce a powerful teleautonomous control system.

Being a research system in an ever evolving technological environment, and also bound by certain project constraints, this system comprises electronics and software technology elements that were developed independently. In some ways, this system is a hybrid between the JPL Advanced Teleoperation System [1] and the JPL Telerobot Testbed Manipulation System [2]. Its architecture can also be unified into a more homogeneous configuration such as in the Kali system [3]. In fact, the initial project intent was that this system development be followed by an upward compatible conversion to the JPL Telerobot System and/or the Kali system as they both mature in CY90.

The major components of this integrated system include: (1) FRHC - a six axes force reflecting hand controller; (2) UC - a 'Universal Controller' for the low-level servo control of the FRHC and the hand trigger in the FRHC; (3) VME high level control electronics - for driving the UC and interfacing with the robot electronics; (4) RCCL - control software on a microVAX for the robot high level controls; (5) Unimate Controller - the low level servo controller for the PUMA 560 or 762 robot; (6) PUMA 560 or 762 robot; (7) a 'GSEE' - a sensor-based end effector, which has wrist force-torque sensing and force-controlled finger gripping; (8) graphics generator - displaying real-time force-controlled information; and (9) a user interface - a multi-window 3-tier menu system on a SUN 3/60 workstation.

Figure 1 shows the system configuration. In the foreground, from left to right are the SUN terminal (with 5 windows displayed), a color graphics display of force-torque and GSEE data, 2 video monitors showing camera views of the worksite, the UC driving the FRHC, the FRHC (right-handed), an IBM PC interfacing with the UC, and the VME electronics chassis. In the background from right to left (some items not visible) are the Unimate controllers for the PUMA 560 and 762, the microVAX driving the robots, a PUMA 762 robot with the GSEE, and a PUMA 560 robot with a pneumatic gripper.

The overall capabilities of this integrated system can be summarized as follows:

- It is a telerobot system capable of force reflecting teleoperation and shared/teleautonomous control (i.e. teleop mixed auto force compliant control).
- The teleautonomous capability allows gain mixing per axisl gain (k) for teleop and gain (1-k) for auto control.
- The man-machine interface is a 3-tier menu system, for models and parameters selection.
- It has a multi-window display on a SUN 3/60 workstation.
- Teleoperation modes include: position (Cartesian) mode in tool or mixed world-tool coordinates; rate (Cartesian) mode; force reflection on/off mode; index on/off mode.
- Control scaling includes: position, rate and force scaling; force-torque biasing.
- Color graphics provide a display of force-torque data,

the end-effector (GSEE) grip force, GSEE grip opening, GSEE internal temperature; graphics software also permits axes transformation and scaling.

- GSEE gripper control is integrated with the FRHC hand trigger; a hold button on the hand trigger commands the GSEE to hold position.
- GSEE has position control mode, rate mode, and force mode; a combined position/rate/force mode allows the closing of the gripper until object contact, when force control is effected.

For further details on the specifications of the GSEE and the FRHC, refer to [1, 5].

2 System Description

Figure 2 and Figure 3 shows the overview hardware block diagram of the system and the corresponding algorithm flow respectively. For detailed board-level hardware configuration, date flow diagram and software flow diagrams, refer to [6].

The UC (Universal Controller, also known as UMC, Universal Motion Controller in [1]) provides the control and direct interface with the FRHC. It contains two NS 32016 processors, two joint interface cards, power amplifiers, and one XYCOM parallel interface card. One NS 32016 processor (also named UC processor) is in charge of executing servo algorithms and internal user interface, whereas the other NS 32016 processor (also known as the UC COMM processor) is in charge of communication with the VME chassis. In position mode, the UC receives six joint encoder positions and other analog readings from the FRHC. The UC processes and transmits this data to the VME processors via the XYCOM parallel interface. In the return loop, the UC receives the joint setpoint (encoder counts) from the VME processors, and drives the FRHC.

An IBM PC provides the development environment and downloading facilities to the UC processors. Software for the two UC processors are written in assembly language [1].

The VME chassis consists of three Motorola 68020 processor cards, one Ethernet card, one XYCOM parallel interface card, one BIT3 shared memory card, and a set of two Parallex graphics cards. The main 68020 processor card, known as the HACS processor card, is mainly responsible for computing the forward kinematics of the FRHC, hence the Cartesian incremental positions/orientations for the PUMA robot. This same card also converts the robot force/torque sensor data to pseudo robot incremental setpoints, and computes the inverse kinematics of the FRHC to produce the desired joint positions for creating force reflection in the FRHC. The secong 68020 processor card, known as the Interpolator I/O card (also called the UC/COMM card or FRHC/Moper Card), directs I/O with the UC via the XYCOM card, and computes the interpolated joint encoder setpoints for the FRHC. The third 68020 processor card known as Graphics processor card, is in charge of the graphics display, driving the two Parallex graphics cards. The BIT3 shared memory card provides the communication medium among the VME cards and the microVAX on the robot side.

The data sharing and computation in the VME chassis are performed in the following manner. The Interpolator I/O processor receives the FRHC encoder values and analog readings (digitized in the UC) from the UC via the XYCOM interface and deposits them on the BIT3 shared memory. Then the HACS processor reads them from the shared memory, and computes the corresponding Cartesian incremental transformations to derive the PUMA robot setpoints. In the return loop, the HACS processor reads the force-torque sensor readings on the shared memory, as deposited by the RCCL/microVAX. A force gain matrix transformation provides the incremental pseudo PUMA setpoints, corresponding to the force/torque "position accomodation" errors. The HACS then computes the FRHC inverse kinematics to obtain the corresponding desired FRHC joint positions. The Interpolator I/O processor then processes and interpolates these joint encoder setpoints, and returns them to the UC, backdriving the FRHC.

A SUN 3/60 computer running on the VxWorks commertial operating system provides the development environment for the VME processors. All programs were written in "C" and downloaded to the processors either via the Ethernet or via serial ports.

For the robot control software execution and development, a microVAX running on the UNIX operating system and installed with RCCL [7] is used. This microVAX consists of one main CPU and one slave CPU. The main CPU provides the UNIX program development environment and the user interface to the operator/programmer. The slave CPU executes the real time portion of the RCCL code on a modified UNIX kernel. The interface with the VME chassis (and hence FRHC) is through the BIT3 shared memory. The interface with the PUMA Unimate controller, specifically the LSI-11 processor in the controller, is through a DRV-11 parallel interface. Every cycle (normally 20 milliseconds), the slave CPU interrupts the VME system and gets the Cartesian setpoint for the PUMA robot. It also computes the inverse kinematics, converting the Cartesian setpoint to the joint setpoints, then sending them over to the LSI-11 PUMA processor via the DRV-11 interface. In the return loop, the slave CPU receives the force /torque data from the LSI-11 processor and deposits them on the shared memory to be received by the VME processors. Notice that no forward kinematics of the robot is performed on the microVAX, because the design does not specify force reflection due to robot position lag from the FRHC commanded position. That is, the present design derives the FRHC force reflection purely from the robot wrist sensor.

The LSI-11 processor in the PUMA controller performs I/O functions for the PUMA robot. It reads joint setpoints from RCCL/ microVAX and sends them over to the lower level 6503 processors. Conversely, it reads the joint encoder positions of the PUMA's six joints and send them over to the microVAX. The LSI-11 also collects the force/torque data from either the GSEE (which has its own integrated force/torque sensor) on the PUMA 762, or from the Lord force/torque sensor on the PUMA 560. The data is shipped to the microVAX. Finally, the low level 6503 microprocessors provide the direct interface with and control over the individual robot joints. They perform joint interpolation, run the servo algorithms, and drive the robot joints. They also process the data coming from the PUMA encoders, compute the encoder values and make them available for the LSI-11 processor to read.

3 System Operation and Performance

This teleautonolous system is designed to be a turn-key system, to be used by application users who are not system programmers nor development programmers. Facilities are, however, very flexible and available to programmers for software modifications. All software for the VME processors can be edited, compiled and downloaded from the SUN workstation operating under VxWorks development environment.

The user interface is a multi-window menu display on the SUN workstation (see Figure 1). These windows include the (a) FRHC_MAIN (also known as the HACS, HAnd Controller Software) user window, (b) RCCL_MAIN user window, (c) VME_UC_MAIN user window, and (d) Graphics window. Two other windows are the editing window and the console window (for system messages).

The user windows allow the user to change operation modes and specify parameters and configuration. Examples include the choice of position versus rate mode, manual non-force reflecting control versus shared control versus teleautonomous control. Force/posi- tion loop gains, tool transformation parameters, end-effector grip force, rate, etc. are all selectable and changeable from the menus. Typical robot align, visit, and move commands are available at the RCCL window.

Figure 4, 5 and 6 show the 3-tier menu input system for the FRHC_MAIN, RCCL_MAIN, and VME_UC_MAIN processes. For further details, refer to [6].

Very satisfactory, nontheless limited, operational experience has been gained in the use of this teleautonomous system in the performance of generic telerobotic tasks, such as expected for the NASA's Flight Telerobotic Servicer. These tasks include peg-in-hole insertion, contour following, and door opening. Certain combination of gains, teleautonomous scaling, and choice of shared control axes, have been determined to be efficient in the performance of these tasks (see reference [6]).

The performance of this system is mainly constrained by the speed of the microVAX RCCL hardware/software. The bottleneck there requires the whole system to be basically synchronized to 50 Hz. When the entire control goes around the loop, from the hand controller to the robot controller/end-effector and back to the hand controller (for force reflection), a time latency of some 100-125 milliseconds is estimated. For pure teleoperation control, such a time latency of some 100-125 milliseconds is estimated. For pure teleoperation control, such a time latency is not desirable. In fact, experiments on this system have shown potential control stability problems - not in the sense of control stability when the system is on its own, or even affected by outside disturbances. Potential problems arise when the human operator is in the loop, depending on the operator's experience and "impedance" in his grip of the force-reflecting hand controllers.

It is experienced, as also predicted by the theory, that the teleautonomous capability of the system can overcome such potential instability problems. A linear combination of the teleop control input with the autonomous force control has provided very satisfactory system response in the performance of the above tasks. Note that the autonomous mode of control is not performance limited by the same latency rate. Control there is local to the microVAX/RCCL, which has 50 Hz cycle rate. Low level servo rates are even higher, close to 1 KHz. The teleautonomous performance has consistently ranked better than teleoperation or shared (hybrid force-position) control performance.

4 Summary

A powerful and flexible telerobot control system has been integrated and tested, which has a teleautonomous capability that outperforms teleoperation systems and shared (hybrid force-position control) systems. It is complete with hand controllers, electronics, and integrated with the PUMA robots equipped with a custom servo driven end-effector. This system is also designed to be upward compatible with a higher-performance system, where the computation limitation of the microVAX/RCCL is replaced by a faster computer or cluster of distributed processors. As is, this system can serve as a flexible telerobot testbed whereby the human factors aspects and application experiments can be extensively investigated.

5 Acknowledgement

This work was performed at the GE Advanced Technology Laboratory, under contract from the Jet Propulsion Laboratory, California Institute of Technology.

References

[1] A. Bejczy and Z. Szakaly, "Universal Computer Control System (UCCS) for Space Telerobots," Proc. 1987 IEEE Int. Conf. on Robotics and Automation, Raleigh, N. Carolina, Mar. 1987, pp 318-324

[2] S. Hayati, T. Lee, E. Kan and J. Lloyd, "The Telerobot Manipulator Control and Mechanization Subsystem," Proc. 2nd NASA Symposium on Space Telerobotics, Pasadena, CA. Jan. 31-Feb. 2, 1989

[3] P. Backes, S. Hayati, V. Hayward, and K. Tso, "The KALI Multi-Arm Robot Programming and Control Environmen,", Proc. 2nd NASA Symposium on Space Telerobotics, Pasadena, CA. Jan. 31-Feb. 2, 1989

[4] P. Blaire, F. Hawes, R. Killion, and L. Robinson, "Goddard Smart End Effector (GSEE): System Description and Operating Instructions," Jet Propulsion Laboratory, California Institute of Technology, Internal Document, Feb. 1988.

[5] D. McAffee, "Teleoperator Subsystem/Telerobot Demonstrator: Force Reflecting Hand Controller Equipment Manual," Jet Propulsion Laboratory, California Institute of Technology, Document #JPL D05712, Jan. 1988.

[6] M. Junod, "Operation Manual for the GSFC Telerobot Control System 1," GE Aerospace/Advanced Technology Laboratory, developed under Contract #958149 with Jet Propulsion Laboratory, Pasadena, CA., Sept. 1989.

[7] J. Lloyd, M. Parker, and R. McClain, "Extending the RCCL Programming Environment to Multiple Robots and Processors," Proc. 1988 IEEE Int. Conf. on Robotics and Automation, Philadelphia, Pa., Apr. 24-29, 1988.

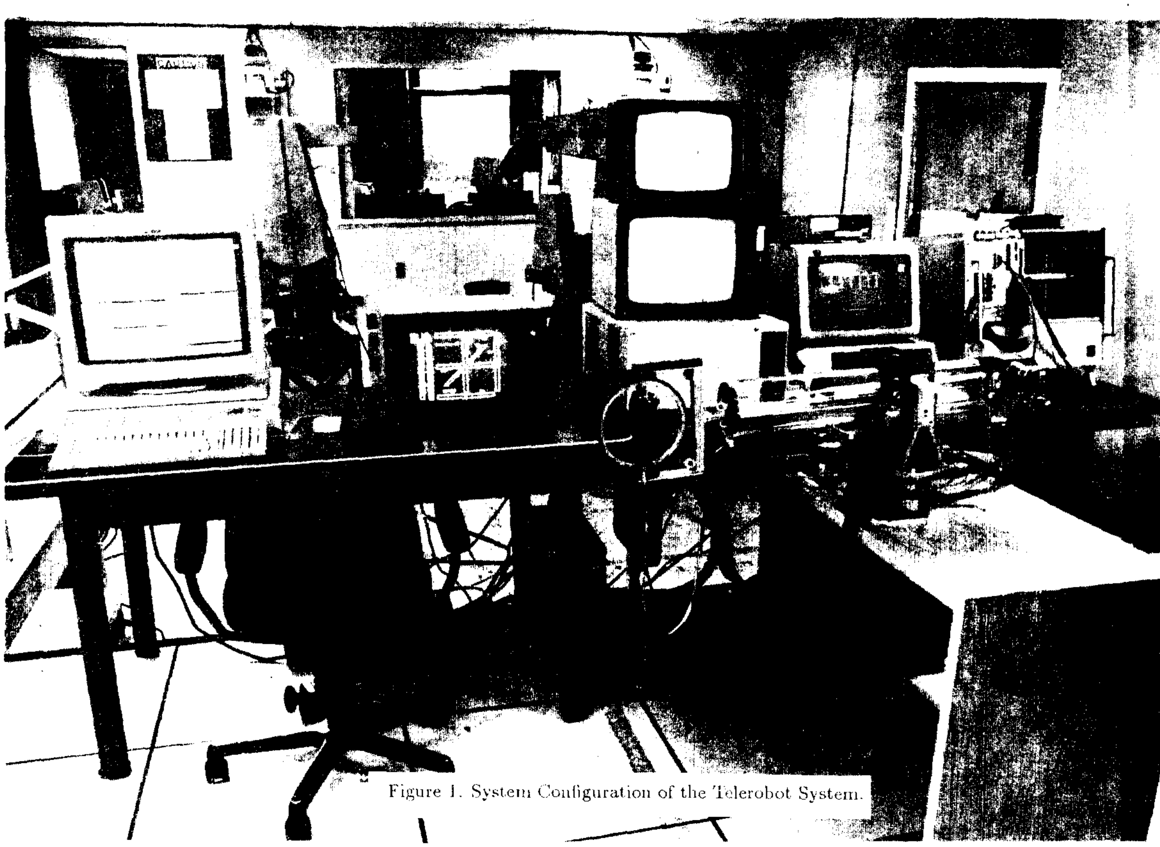

Figure 1. System Configuration of the Telerobot System.

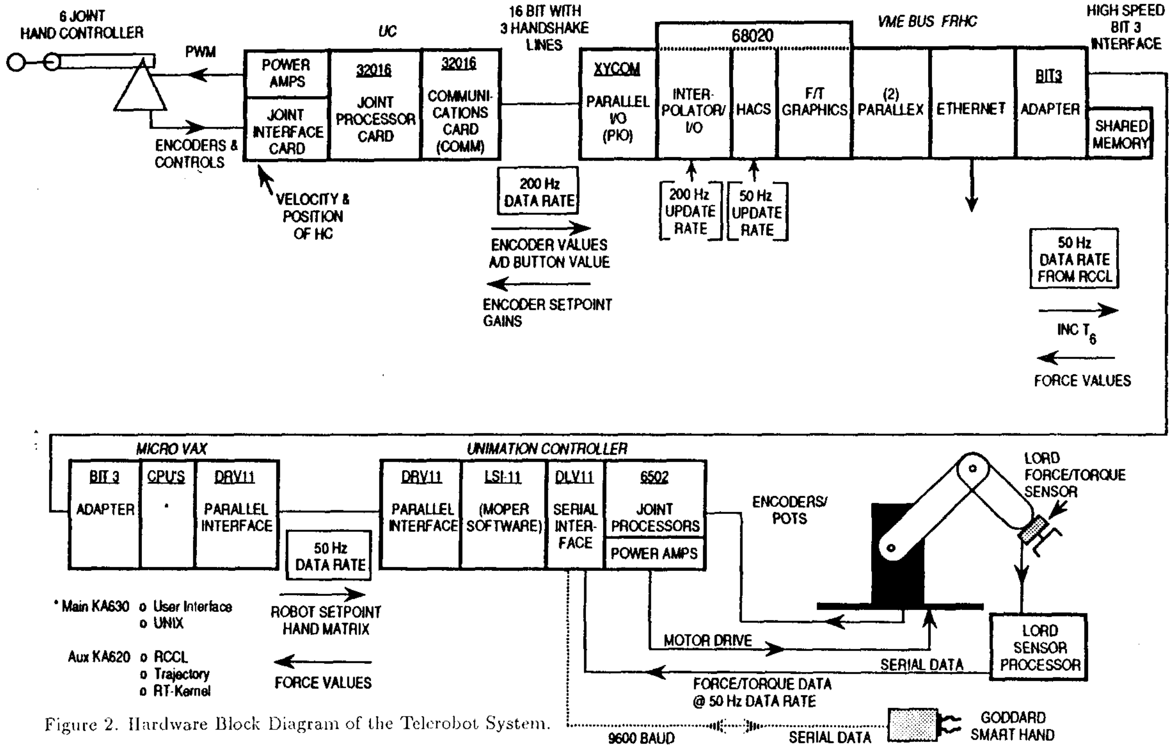

Figure 2. Hardware Block Diagram of the Telerobot System.

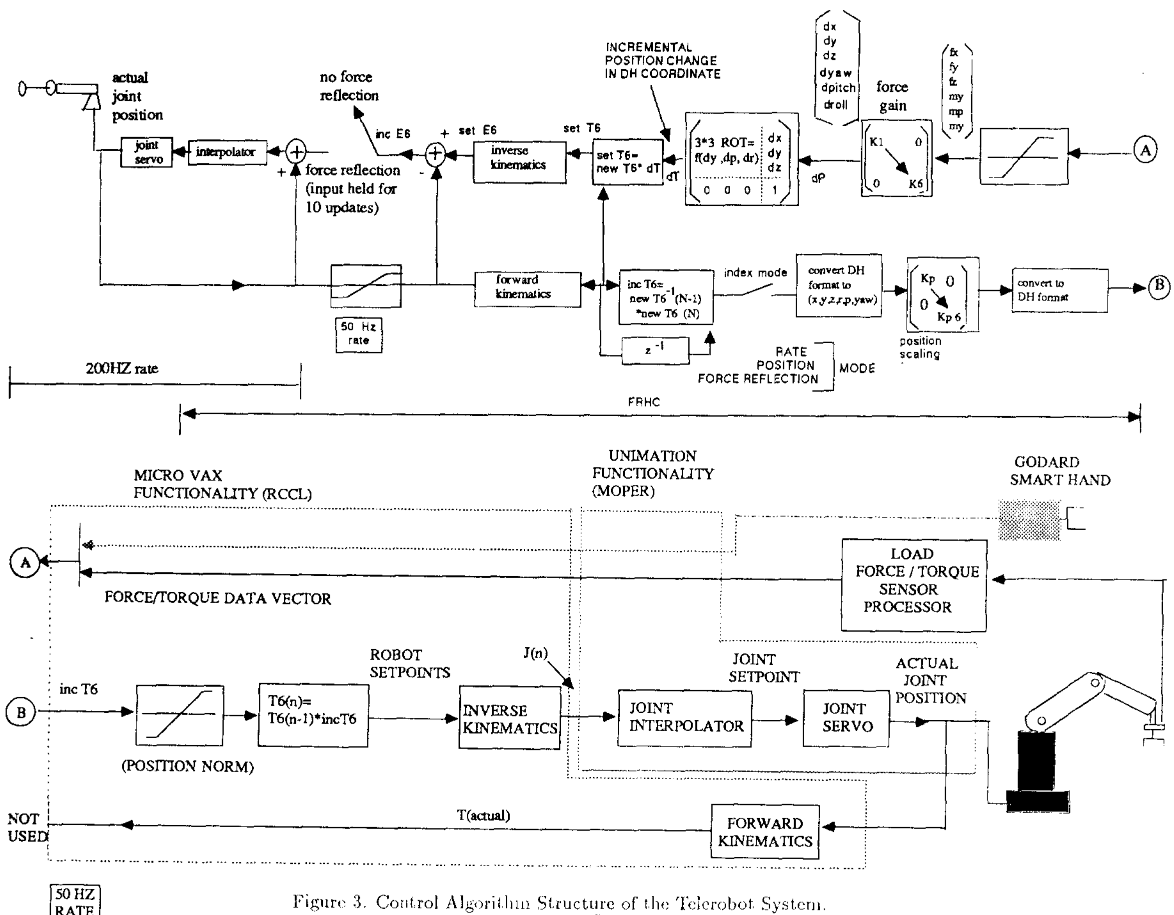

Figure 3. Control Algorithm Structure of the Telerobot System.

Figure 4. RCCL_MAIN Menu

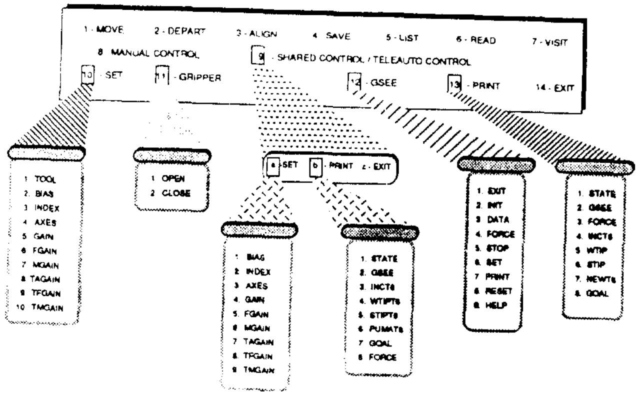

Figure 5. VME_UC_MAIN Menu

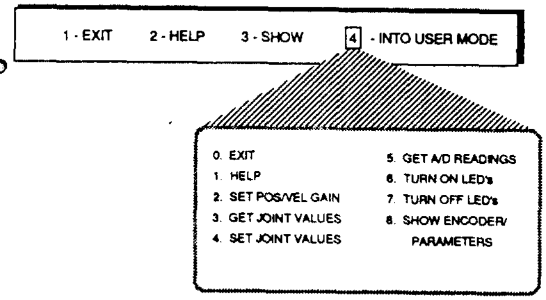

Figure 6. FRHC_MAIN Menu