THE FLIGHT ROBOTICS LABORATORY

Patrick A. Tobbe, Marlin J. Williamson, and John R. Glaese
Control Dynamics Company

ABSTRACT

The Flight Robotics Laboratory of the Marshall Space Flight Center is described in detail. This facility, containing an eight degree of freedom manipulator, precision air bearing floor, teleoperated motion base, reconfigurable operator's console, and VAX 11/750 computer system, provides simulation capability to study human/system interactions of remote systems. This paper describes the facility hardware, software, and subsequent integration of these components into a real time man-in-the-loop simulation for the evaluation of spacecraft contact proximity and dynamics.

INTRODUCTION

The Flight Robotics Laboratory of the Marshall Space Flight Center provides sophisticated simulation capability in the study of human/system interactions of remote systems. The facility consists of a four thousand square foot precision air bearing floor, a teleoperated motion base, a dynamic overhead target simulator (DOTS), a remote operator's reconfigurable station, various simulation mock-ups, and a VAX 11/750 computer system for real time operation.

The motion base is an air bearing vehicle, with limited capability in six degrees of freedom, which may serve as a controllable chaser craft or as a target vehicle. This vehicle contains six pressurized air tanks for pneumatic power of its thirty-two thrusters. This provides a means of remote operation and control.

The dynamic overhead target simulator (DOTS) is an eight degree of freedom (DOF), heavy duty electric manipulator capable of traversing over the entire air bearing floor. The system is composed of a precision overhead X-Y crane to which a six degree of freedom robot arm is mounted. The VAX 11/750 computer is used in real time to convert tip position and orientation commands into crane position and arm joint velocity commands. These commands are generated through inverse kinematic relationships and digital control laws housed on the computer. An elaborate real time safety algorithm is also driven by the computer to perform collision avoidance, end effector force/torque limiting, joint position and rate limiting, and communications checks.

The remote operator's workstation consists of hand controllers, two monitors, and a large screen display.

The tip commands for the manipulator are generated by a position model on the computer. This position model is simply a set of equations which model the hardware

* Work performed by Control Dynamics Company for NASA Marshall Space Flight Center, Contract #NAS8-36570

that the facility is configured to simulate. For example, in order to perform studies of contact dynamics between two orbiting spacecraft, a position model consisting of the equations of motion for the two craft acted on by thruster, gravitational, and contact forces is used.

However, this facility is not limited to contact dynamics studies. With the correct position models, the facility may be used to simulate space station construction tasks in zero gravity or for the testing and training of personnel flying remotely piloted vehicles. These studies also investigate such human factor concerns as light and camera positions on the vehicle, control system sensitivity, and transmission time delays. Since the manipulator control law is implemented on the computer, it is a simple matter to use the facility as a testbed for new robotic control algorithms.

In the fall of 1988, the facility will be configured to perform real time, man-in-the-loop docking studies of the orbital maneuvering vehicle (OMV) with a moving target. This test series will make use of the manipulator, VAX 11/750, and the remote operator's work station. A FORTRAN 77 simulation consisting of the real time position model and arm controller, as well as dynamic models of the arm, actuators, and sensors, will be used to predict the results of the tests. These hardware models have been generated and incorporated into the simulation. The model parameters will be updated as the facility hardware becomes available for testing.

The purpose of this paper is to describe the hardware and software which will be used in these docking studies. It will explain the integration of these components and efforts taken to validate the facility.

FACILITY CONFIGURATION

Currently, the facility is being configured to perform real time, man-in-the-loop docking studies of the OMV with a moving target using the three point docking mechanism. The active half of this mechanism is simply a set of three electromechanical jaws mounted on the verticies of an equilateral triangle. The passive half of this mechanism is a set of three bars or "towel racks", similarly mounted, which the jaws will grapple. The active half will be mounted to the arm while the passive half will be fixed at some location in the facility.

Shown in Figure 1 is a block diagram of the facility configuration. A pilot at the remote operator's workstation views a screen generated by a camera located on the end of the arm. The position and orientation of the camera with respect to the docking mechanism would be such that it duplicates the flight hardware configuration. The pilot then issues of commands for the OMV control system through a set of joysticks at the workstation. The output of the joysticks is fed into the computer for use in the position model.

The position model for the OMV docking studies consists of a computer simulation of two rigid vehicles, the target and chaser, in a circular orbit. The vehicles are modeled in the simulation using Hill's and Newton-Euler equations (ref. 1). The target is acted on by gravity and vehicle contact forces and torques. The chaser is acted on by gravity, vehicle contact, and control system thruster forces and torques.

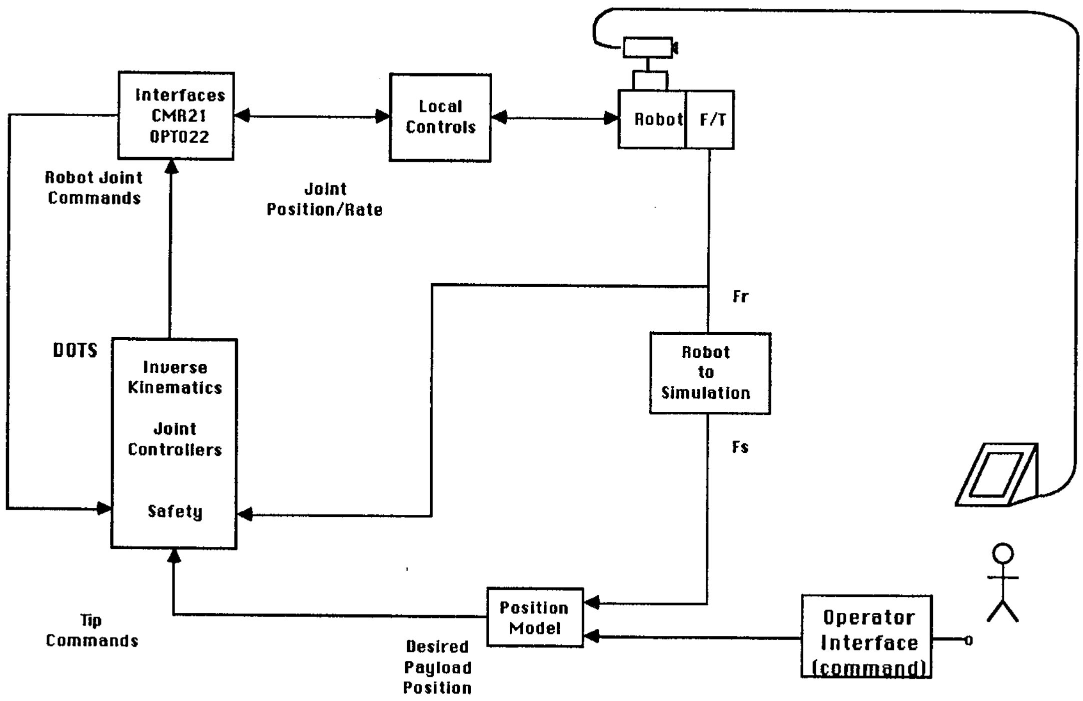

Figure 1 Facility Configuration

A force/torque sensor on the end of the arm senses contact, gravity, and small inertial forces and torques due to the motion of the arm, The output of the sensor is filtered to generate vehicle contact forces and moments for use in the equations of motion. The position model then produces the relative position and orientation of the chaser docking mechanism with respect to the target docking mechanism. Manipulator tip position and orientation commands are formulated such that the relative position and orientation of the docking mechanisms of the facility will match those produced by the position model.

The manipulator tip commands are now used by the inverse kinematics routine to produce commanded joint positions. The commanded joint positions, along with joint encoder feedback, are used by the digital control algorithms to calculate joint rate commands for the local rate servo systems mounted on the arm.

The safety algorithm also uses commanded joint and tip positions for real time collision avoidance. Safety envelopes are defined about the arm and various mock-ups on the floor. These zones are described by equations and never allowed to intersect. Safety also uses the output of the force/torque sensor to limit the reaction of the end effector. Encoder and tachometer feedback are used to limit joint position and rate.

The local controllers on the arm initiate the desired motion to mimic the relative motion of the orbiting spacecraft.

ORIGINAL PAGE IS
OF POOR QUALITY

FACILITY HARDWARE DESCRIPTION

The DOTS is presented in Figure 2. It is a 6-DOF arm suspended from an industrial X-Y overhead crane. The crane's bridge spans the width and travels the length of the air bearing floor. The crane's trolley travels along the span of the bridge (covering the width of the air bearing floor). These two joints are controlled with localized position control systems. These control systems are composed of PID controllers and electronic distance meters (EDM). The design of the PID controllers is automated via the set-up software associated with the position control systems. Each of the position control systems for the crane also have a rate loop closed about the joint with DC pulse-width-modulated servo drives and DC motors. These servo loops are also found on the remaining arm joints.

The control algorithm implementation in the real time software regulates the rate commands for the 6-DOF arm. This arm is composed of five rotational and one translational degrees of freedom. Starting at the base of the arm attached to the crane, the waist rotation joint allows continuous rotation about an axis perpendicular to the floor via a slip ring. Using the rated speed of the motor selected to drive this joint and its gear ratio, the continuous joint speed is 1.3 rpm.

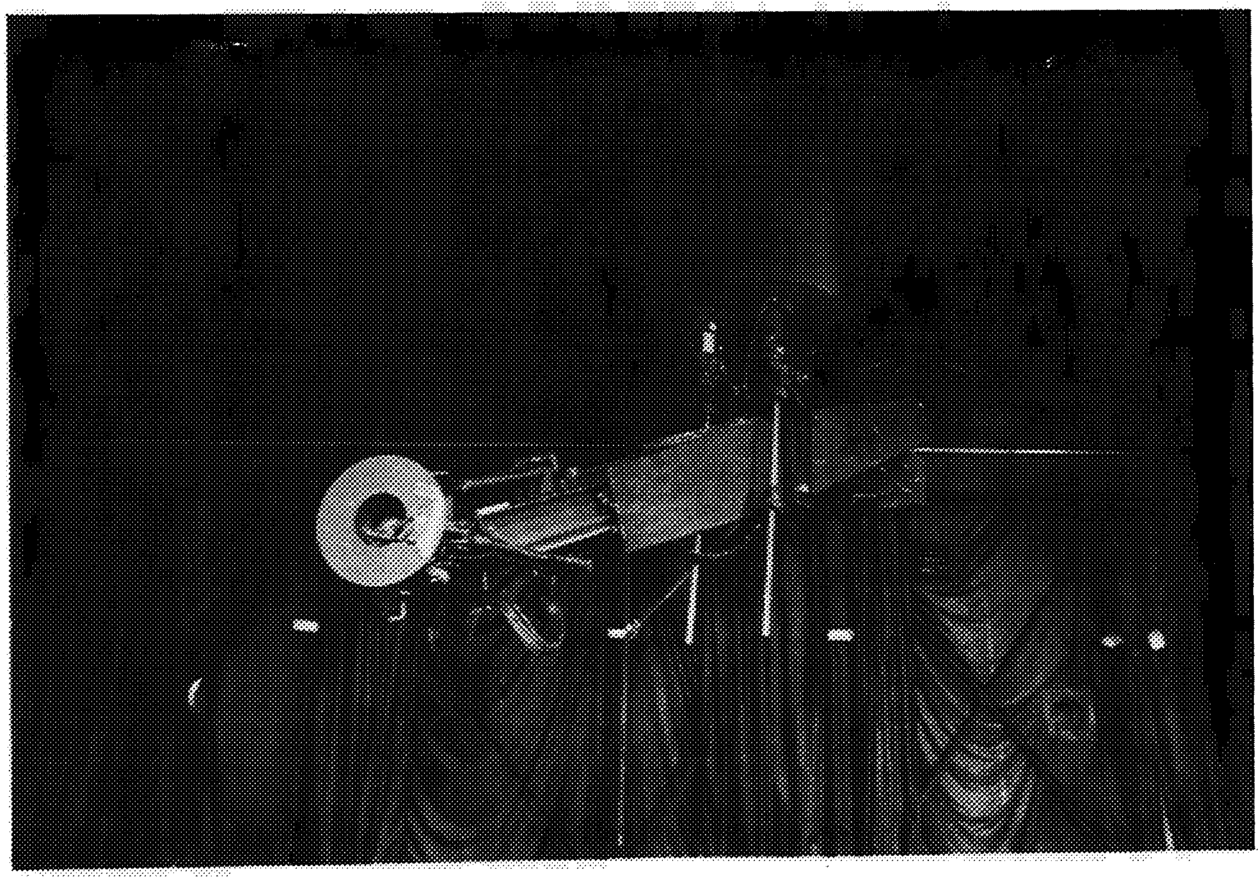

Figure 2 Dynamic Overhead Target Simulator (DOTS)

The next joint encountered is the shoulder pitch. This joint has a ± 50° range with respect to an axis parallel to the floor. A maximum continuous speed of 7.7 rpm is achieved. Two linear actuators (worm gear and motor) are used to drive this joint. A counter weight system offsets the loads seen by these actuators.

Farther up the arm, the lone translational joint is encountered. Its maximum continuous speed is 13.2 cm/sec (5.2 in/sec) with a range of 2.62 m (10 ft.).

The wrist assembly at the end of the arm is composed of three degrees of freedom; wrist yaw, pitch, and roll. The wrist yaw and pitch are linearly actuated with wrist pitch having two actuators (similar to the shoulder pitch). These joints are also limited to ± 50° travel and a maximum continuous rate of 0.2 rpm. The wrist roll has no limit in range due to a slip ring. Maximum continuous speed is 10 rpm for this joint.

Each joint is instrumented with 12-bit encoders and tachometers. The encoders record relative joint position. These values are read by the main computer system through the programmable serial communications device. Here, the values are transformed into a physical measurement of the joint positions.

The tachometers located at the motors (1:1 motor to tachometer shaft ratio) are integral parts of the servo loop systems. To get a reading of joint speed, the tachometer voltages are filtered (to reduce noise) and scaled (to match A/D input). Once filtered, digital measures are taken by a 12-bit A/D device. The main computer system reads these values via the programmable serial communication device. The integer values read are converted to real values and scaled to reform the tachometer voltages. The voltages are related to shaft speed by the voltage constants. Shaft speeds are transformed through kinematic relationships to a measured joint speed.

The programmable serial communication device used to read the tachometer and encoder settings is also used to send the rate commands to a D/A, which in turn generates voltage reference signals for each rate loop on the arm. It also writes the position commands to the position control systems for the overhead crane.

The payload capability of the arm is 454 kilograms (1000 pounds) with a center if gravity offset of 46 centimeters (18 inches) from the wrist roll axis.

The force/torque sensor mounted at the end of the arm has a capability of 44,482 newtons (10,000 pounds) and 6779 newton-meters (60,000 inch-pounds). The accuracy is one quarter percent of full scale.

FACILITY REAL TIME SOFTWARE DESCRIPTION

All of the real time software to be used in this test series will be housed on the VAX 11/750. An executive routine coordinates the input/output operations between the computer and the crane/arm sensors and actuators, force/torque sensor, and operator's console. This routine also calls the inverse kinematics, joint controller, safety, and position model modules. It generates output for a test conductor's terminal screen in the form of commanded and sensed tip/joint positions and error messages. It is being modified to generate additional output to a currently undefined device. The real time simulation cycle time goal is 33 milliseconds. The position model has been described in the Facility Configuration section.

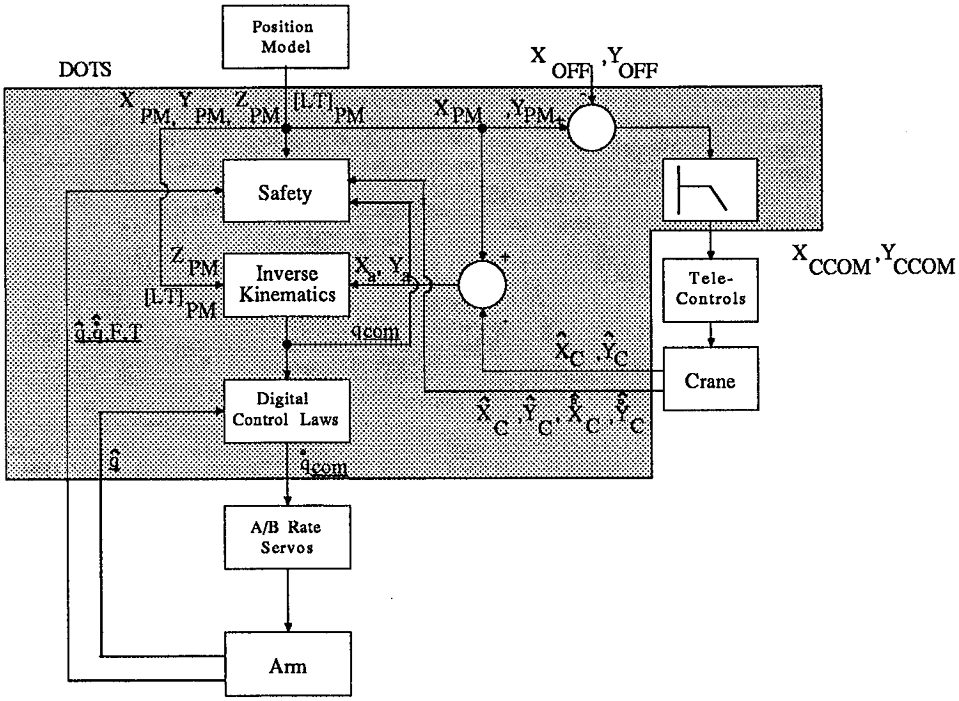

Figure 3 DOTS Controller

DOTS Controller

The DOTS controller, highlighted in gray in Figure 3, is simply the collection of the inverse kinematics, joint controller, and safety routines. An implicit assumption made in this algorithm concerns the bandwidths of the crane and the arm joints. It is assumed that the bandwidths of the arm joints are significantly higher than those of the overhead crane. The crane is commanded to move within an offset distance of the position model output. The arm tip is then commanded to move a distance equal to the difference between the commanded position model output and the current crane position. In this way, as the crane slowly approaches its commanded position, the arm joints will "backoff" to keep the tip at the desired location. This will effectively match the bandwidth of the system to that of the arm within a limited range of motion. This algorithm will also tend to drive the arm to a "home" configuration which avoids joint singularities.

As seen in Figure 3, the output of the position model are the tip coordinates Xpm, Ypm, and Zpm in a lab coordinate frame and the transformation matrix [LT]pm describing the orientation of the arm tip with respect to the lab frame. Xpm and Ypm coincide with the overhead crane degrees of freedom. Offsets in X and Y are subtracted from these commands to keep the arm extension in the "home" position at the middle of its travel. The resulting commands are compensated by a second order filter with a 1 Hertz break frequency. The filtered commands, Xccom and Yccom are then passed to the localized position control systems which drive the overhead crane. The sensed crane positions and rates, $\hat{X}_c, \hat{Y}_c, \dot{\hat{X}}_c$, and $\dot{\hat{Y}}_c$ are fed back to the computer.

The arm X and Y tip position commands, X_a and Y_a, are calculated by subtracting the sensed crane positions from the outputs of the position model, Xpm and Ypm. The inverse kinematics generates a set of six arm joint position commands, qcom. The digital control law produces a set of joint rate commands, $\dot{q}$com, based on the differences of the commanded joint positions, qcom, and sensed joint positions, $\hat{q}$.

The safety module uses the sensed arm joint position and rates, $\hat{q}$ and $\hat{\dot{q}}$ and the overhead crane sensed positions and rates, $\hat{X}_c, \hat{Y}_c, \hat{\dot{X}}_c, \hat{\dot{Y}}_c$, and the output of the force/torque sensor, F and T, to perform collision avoidance, joint position and rate limiting, and arm tip reaction force and moment limiting.

Inverse Kinematics

As previously stated, the inverse kinematics routine generates a set of six arm joint positions to produce a desired tip position and orientation. Since this is real time software, it was very desirable to find a closed form solution and avoid any numerical solution techniques. Care must also be taken to generate a set of equations with a unique solution in order to eliminate 180 degree joint rotations for small changes in the commanded tip position and orientation.

Using the Denavit-Hartenberg notation, seven coordinate frames were located from the base of the arm to the point on the tip to be controlled (ref. 2). This notation places the degree of freedom between the i'th and the i'th+1 coordinate frames along the z axis of the i'th frame. The position vector of the origin of the i'th+1 frame with respect to the i'th frame is along a coordinate axis of the i'th frame. The transformation matrix between the i'th and i'th+1 frames was easily derived for the seven frames. The product of these six matrices is then equal to the transformation matrix of the position model.

The vector from the base of the arm to the tip of the end effector is written as the sum of the vectors between the seven coordinate frames. This vector, when expressed in frame 1 coordinates, is equal to the commanded tip position, Xa, Ya, and Zpm of Figure 3.

The transformation matrix between the frame at the base of the arm and the frame at the tip of the arm, as well as the position vector locating the arm tip with respect to the base of the arm, have now been described as a set of equations in terms of joint degrees of freedom. These equations were then manipulated into a set of equations which produced joint positions as a function of the output of the position model and known arm dimensions.

Joint Controllers

The control algorithm for the DOTS is implemented on the main computer system via several series of digital filters. State variable techniques are used to implement each filter. Each joint (degree-of-freedom) is controlled separately. That is, each joint has its own series of digital filters. Currently, the filters are designed to an uncoupled plant. Each joint is isolated and linearized at nominal operating points prior to designing filters for that joint.

The digital control filters are composed of a first order filter series (first order polynomials over first order polynomials) and a series of second order filters (first order over second order polynomials). The series are implemented so that the output of the i'th filter is the input to the i'th+1 filter and so on. The filters are serially connected with the last first order filter output being the input to the first second order filter. The input to the filter series (first, first order filter) for the overhead crane are the position commands generated by the inverse kinematic module. The output of these two filter series will be the position command to be sent to the local PID controllers for each crane axes. The other filter series will have position error (commanded minus measured position) as inputs. These filters are designed so that the output will be the desired joint rate in terms of a voltage to be applied to the servo controllers.

The user specifies the coefficients of each filter in the associated series as well as the number of first and second order filters in that series. The gain specified for the series is the product of the individual filter gains in the series and the design gain. This implementation also allows for only first order filters in the design or only second order filters in the design or just a gain to represent the design. With the current designs, the overhead crane axes were limited to a 1 Hertz bandwidth and the arm joints have bandwidths on the order of 3 Hertz.

Safety

The safety software performs collision avoidance, force/torque limiting, joint position and rate limiting, and reasonableness checks on the integrity the sensor output. A failure in any of these tests will cause the simulation to stop and generate an error message on the test conductor's terminal.

To prevent undesirable collisions between the arm, simulation mock-ups, and the facility walls, floor, and ceiling, safety zones are defined in terms of planes, spheres, and shapes called cylindroids. A cylindroid is a cylinder with hemispherical ends. The facility walls, floor and ceiling are defined by planes. The arm is protected by a set of spheres and cylindroids which envelope it. A sphere is defined by locating its center and specifying a radius. A cylindroid is defined by locating the centers of the hemispheres and specifying a radius. The simulation mock-ups, which are held stationary, are also protected by user defined spheres and cylindroids. The collision avoidance problem then reduces to finding the minimum distance between the center of sphere and a plane, the centers of two spheres, the center of a sphere and the axis of a cylindroid, the axis of a cylindroid and a plane, and the axes of two cylindroids. These distances are then compared to the radii of the spheres or cylindroids to determine if the zones overlap. The locations of the spheres and cylindroids protecting the arm must be calculated in real time using encoder feedback and geometrical relationships which describe key points on the arm. Obviously, the zones must be large enough to allow for the system time delay and joint braking distances. The collision avoidance routine works on commanded positions to minimize the size of the zones. In order to allow contact necessary for docking studies, user defined zones may be turned off with respect to the sphere protecting the payload of the arm.

Force/torque and joint position/rate limiting is easily accomplished using appropriate sensor feedback. The integrity of the encoders and tachometers is tested in the following way. The encoder readings for consecutive cycles are numerically differentiated to produce an average joint rate. The tachometer readings of motor shaft rates, also for consecutive cycles, are averaged and transformed to joint rates.

If the difference of the calculations is larger than a specified band, the simulation is stopped.

There is also a limit on the magnitude of the error between commanded and sensed joint positions.

VALIDATION EFFORTS

A series of test are in progress to validate the facility. These tests begin with the check out of the communications software between the main computer system, the programmable serial communications device, sensors, and remote operator's console.

The motor and servo drives and controllers will be tuned to generate desirable responses. Rate step responses for each joint will be obtained to measure the motor and tachometer voltage constants, sensor parameters, joint friction, and mass properties.

The information obtained in the hardware characterization tests will be used in a numerical computer simulation of the facility. The simulation will be exercised to predict open loop responses and tune up the digital control filters.

Closed loop position step responses will be measured and compared to similar numerical simulation results to verify the joint controllers. At the same time the safety code will be exercised.

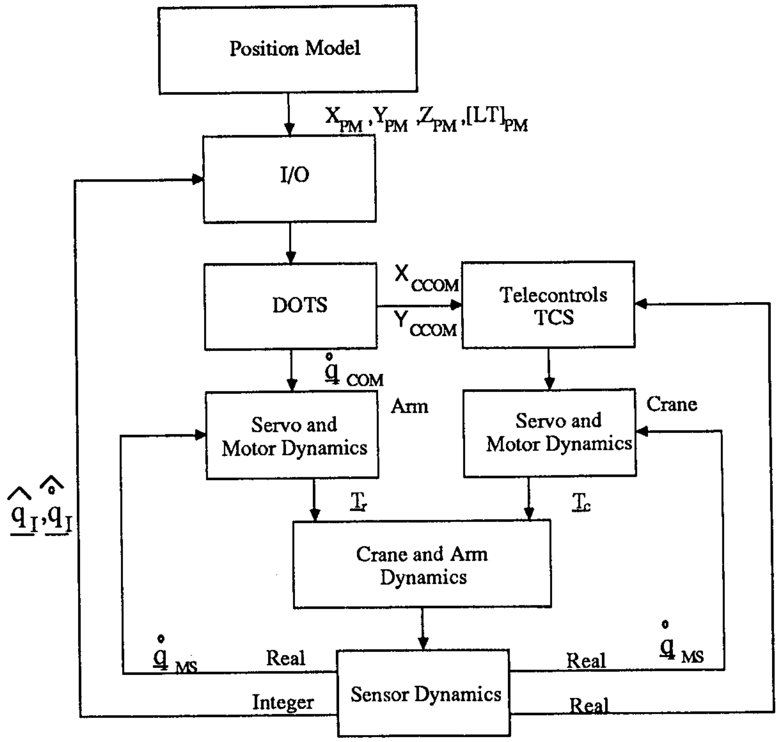

Figure 4 RF^3 Simulation

Upon successful completion of the joint controller and safety tests, simple springs will be used as docking mechanisms to validate the position model in end to end tests.

The numerical simulation of the facility is shown in Figure 4. It consists of the real time position model and DOTS controller software coupled with mathematical models of the manipulator, sensors, and actuators. The local PID controllers for the overhead crane are modeled as digital PID controllers (ref. 3) and generate voltage outputs for the servo drives. The commanded arm joint rates, $\dot{q}com$, are converted to voltages and processed through a model of a 12 bit D/A for the servo systems. The motors are modeled as typical DC armature controlled motors neglecting second order effects (hysteresis) (ref. 4). Using voltage outputs from the servos, the motor models generate torques, Tr and T_c, which are used by the dynamic model of the manipulator. The equations describing the manipulator were derived from a Lagrangian formulation assuming a connection of rigid bodies. These equations are not limited by small angle approximations. The manipulator dynamics module produces a set of joint positions and motor shafts rates for input to the sensor dynamics module. The sensor dynamics module generates output based on models of the encoders and tachometers with the 12 bit A/D quantization effects. The integer values of the joint positions and motor shaft rates are then converted to real numbers in the input/output routine preceding the DOTS controller.

Currently, this simulation is being updated to include a contact dynamics model of the docking mechanism and quantization effects of the force/torque sensor.

CONCLUSIONS

The Flight Robotics Laboratory is currently undergoing validation testing. It will provide real time, man-in-the-loop simulation capability for the study of human/system interactions. The facility can be used for contact dynamics studies, remotely piloted vehicle tests, lighting and camera placement evaluations, and simulation of space station construction tasks.

Future test programs include the verification of an automatic rendezvous and docking system and capture systems for tumbling satellites.

Late next year, the air bearing floor will be expanded. The motion base will be integrated for use with the manipulator and VAX 11/750 computer system.

REFERENCES

1. Kaplan, Marshall, Modern Spacecraft Dynamics and Control, John Wiley and Sons, New York, New York, 1976

2. Asada, Haruhiko; Slotine, Jean-Jaques, Robot Analysis and Control, Wiley-Interscience, New York, New York, 1986

3. Kuo, Benjamin, Digital Control Systems, Holt, Rhinehart, and Winston, New York, New York, 1980

4. Dorf, Richard, Modern Control Systems, Addison-Wesley, Reading, Massachusetts, 1981