2. Hardware

2.1 Mechanical Hardware

The main work-horse of the NYU educational project was chosen to be a multi-functional ED I robot system. This system was designed with several goals in mind: Firstly, the system was required to be simple and inexpensive in order that it could be reproduced and disseminated easily—with the cost going down with each reproduction process.

The simplicity of the system was achieved by chosing the arm to be based on the *direct-drive technology*. Each joint (altogether four) itself is a motor without any complicated transmission or coupling mechanism. Each of the main links (altogether two) is simply an aluminum hollow rectangular tubing with the ends being motor mounts that provide adaptive connections. Also, the rectangular tubing can be interchanged with another of a different size, thus providing an easy way of modifying the robots kinematic (as well as dynamic) parameters, if a particular application requires it.

The kinematic structure is based on the standard SCARA architecture, which simplifies the kinematics and dynamics considerably as the main component of the arm is just a two-link planar kinematic chain. The actuators are also so chosen that the control laws are not too complicated. The only sensors that are currently available are the position sensors (encoders) already integrated with the motors and easily accessible via the motion control hardware.

ED I Mechanical Features	
θ_1 *Axis Range*	0–300 degrees
θ_2 *Axis Range*	±105 degrees
z *Axis Range*	0–200 mm
ρ *Axis Range*	∞
Work Coordinate System Selection	YES
Absolute/Incremental Programming	YES
Emergency Stop	YES (Hardware and Software)

Table 1: ED I Features

The main-link motors (Yokogawa DR 1400A and DM 1060B) are hollow. As the links themselves are made of hollow tubings the cabling of the system is rather simple. As a result, the entire robot system can be easily assembled and disassembled with little difficulty and thus, amenable to experimentations requiring reconfiguration and modifications.

Another important design goal was to make the robot system flexible (i.e., multi-functional) by allowing for several attachments and integration of other devices. For instance, the system can be easily modified to act as a manufacturing cell with addition of chucks, fixtures and x-y table (with a pan-tilt mechanism). Also, cameras can be easily attached to the arm itself for visual servoing, or in a stationary position just to provide some amount of visual information. Some of the experiments using the arm include: measurement of magnetic fields of various components of NYU miniature DD actuators, filming of models to be incorporated with computer generated animation graphics, controlling a reactive gripper, etc.

Last but not least, the robot system was designed to be safe for even novice users. This goal was to be achieved without sacrificing the system's ability to be used in heavy industrial applications.

Thus, the safety was achieved not by down-scaling or under-powering the entire system, but by providing layers of protective mechanisms. This issue is addressed in detail in Section 5.

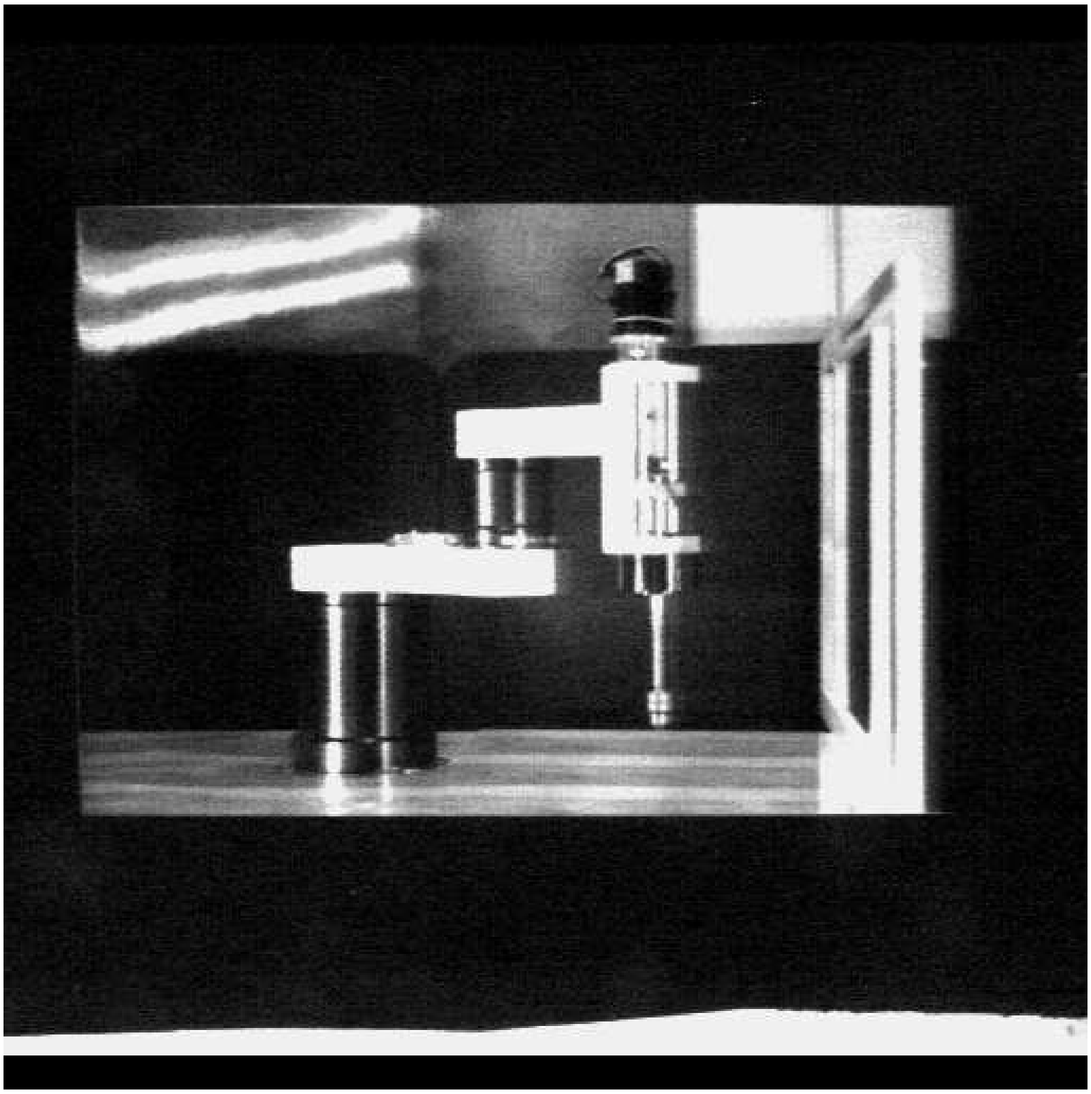


Figure 2: **ED I robot. June 1993.**.

The ED I arm was constructed with the facilities and technical skills available within the Robotics and Manufacturing Laboratory of NYU. The construction required only simple standard machine shop methods. The construction can be speeded up considerably if one uses the CNC machines to produce a small batch of the components. No special manufacturing practices would be necessary for either methods of construction.

Most of the components are commercially available. The main-links that required in-house construction are simple aluminum ($3 \times 5''$) box tubings and are rather easy to construct. Some of the components of the z axis required some skilled machining. The actuators, the sensors and the electronics are based on off-the-shelf components.

2.1.1 ED I DD Arm

ED I is 4 degree-of-freedom robot arm with SCARA like architecture. The main links—shoulder, θ_1 axis, and elbow, θ_2 axis—form a planar kinematic chain. At the end of the second link, the z axis provides a linear actuator to move the end-effector in the vertical direction and the distal joint, ρ axis allows the end-effector to be oriented at any angle. The workspace of the ED I robot is shown below.

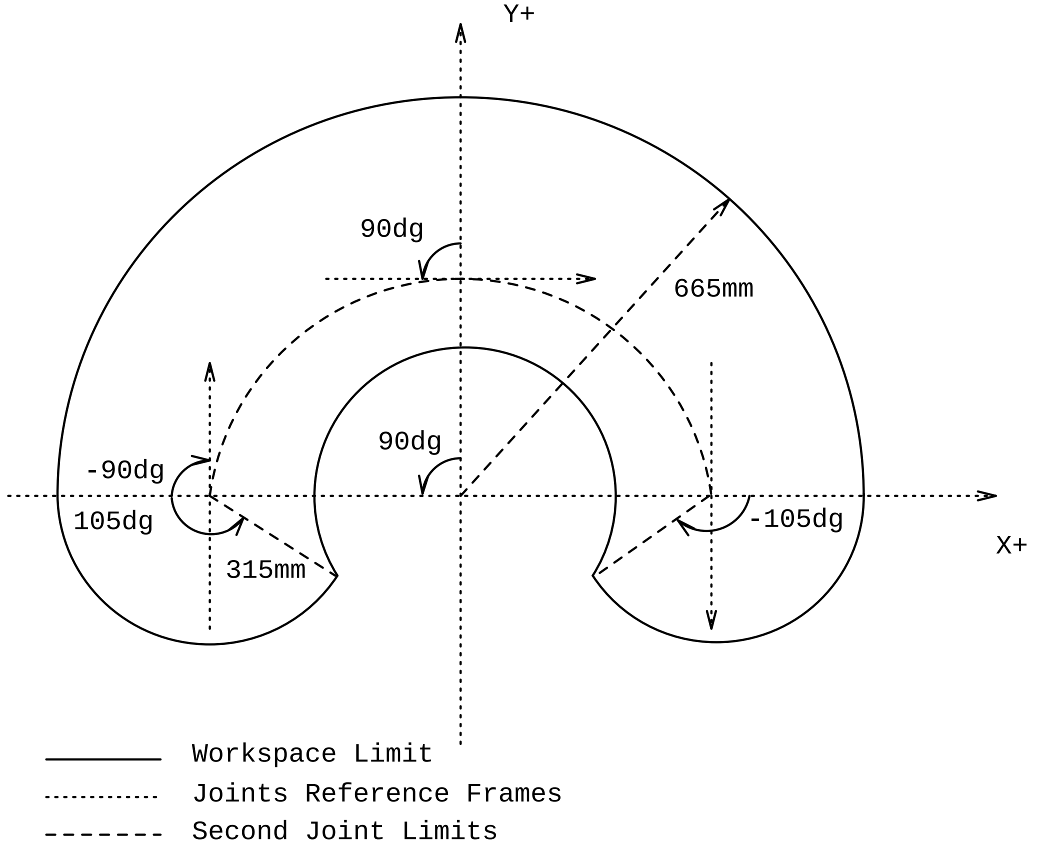


Figure 3: **Workspace of ED I.**

• **θ_1 and θ_2 Axes** The θ_1 and θ_2 joints are actuated by two Yokogawa direct-drive AC servo motors with very high torque as well as high accuracy capabilities. The motor housing is connected to one end of the preceding link (or base) and the armature is directly connected to one end of the succeeding link. There is no additional coupling or transmission mechanism. The motors for the joints have built-in encoders which provide the positional sensor-values directly. The mechanical specifications of the θ_1 (Yokogawa DR1400A) and θ_2 (Yokogawa DM1060B) are given below.

θ_1 **Axis Motor Specification**	
Manufacturer	Yokogawa
Model	DR1400A
Rated Output	200W
Rated Torque	400 N·m
Rated Speed	30 rpm (0.5 rpsec)
Inertia J_M	400×10^{-3} kg·m^2
Related Servo Amplifier Values	
Model	SR1400A-2SN
Encoder Feedback	$102,400 \times 8$ pulses/rev

Table 2: θ_1 Axis Motor Characteristics

θ_2 **Axis Motor Specification**	
Manufacturer	Yokogawa
Model	DM1060B
Rated Output	60W
Rated Torque	48 N·m
Peak Torque	60 N·m
Rated Speed	90 rpm (1.5 rpsec)
Torque Constant	N·m/A
Inertia J_M	23×10^{-3} kg·m^2
Related Servo Amplifier Values	
Model	SD1060B
Encoder Feedback	655,360 pulses/rev

Table 3: θ_2 Axis Motor Characteristics

• **z and ρ Axes** The structure of the z and the spindle (ρ) axes of the ED I is some what unique in many ways. The z axis is a linear actuator that moves the end-effector vertically by a ball-screw mechanism connected to a Yasukawa (USAREM 01 BE 2 K) motor housed at the top of the z axis. There are four members that guide the linear axis. Two of them are *precision ground shafts* with linear bearings that provide significant amount of rigidity to the spindle axis, while letting the entire mechanism to travel up or down with ease and precision. Third member is a *precision ball-screw drive* that transforms the rotary motion of the Yasukawa motor to the linear motion of the z axis. The fourth linear guiding member is the spindle itself and is composed of a *precision linear ball-spline*, holds the end-effector device, and is actuated by the spindle motor (Inland Motor—Model QT 3801 E) using direct-drive technology. The spindle motor rotates the ball-spline freely thus orienting the end-effector at any desired angle.

z **Axis Motor Specification**	
Manufacturer	Yasukawa
Model	USAREM 01 BE 2 K
Rated Output	100W
Rated Torque	0.318 N·m
Continuous Max. Torque	0.367 N·m
Peak Torque	0.955 N·m
Rated Current	1.7 A
Rated Speed	3000 rpm
Max. Speed	4000 rpm
Torque Constant	0.198 N·m/A
Inertia J_M	0.125 kg·cm^2
Power Rate	8.09 kW/s
Related Servo Amplifier Values	
Model	CACR-SR 01 AB 2 ER
Optical Encoder Feedback	1500 pulses/rev

Table 4: z Axis Motor Characteristics

ρ **Axis Motor Specification**	
Manufacturer	Inland Motor Kollmorgen Corp.
Model	QT 3801 E
Rated Output	187 W
Rated Torque	1.75 N·m
Peak Torque	1.75 N·m
Rated Current	9.41 A
Rated Speed	$\approx$ 21600 rpm
Max. Speed	$\approx$ 21600 rpm
Torque Constant	0.185 N·m/A
Inertia J_M	0.202 kg·cm^2

Table 5: ρ Axis Motor Characteristics

• **The End-Effector** The linear ball-spline member of the linear axis is threaded to accept various end-effectors. Following is a short list of the devices that have been used as end-effectors:

1. 3-Jaw Chucks: for manufacturing purposes.
2. Parallel-Jaw Grippers and Pinch-Type Grippers: for robotics pick-and place tasks.
3. Zebra Force-Sensing Wrists.
4. Hall-Effect Sensor: for magnetic measurement purposes.
5. Camera: for computer vision and graphics applications.

• **Safety** In addition to the safety mechanisms necessary to protect the users, we also had to provide several mechanisms to protect the unit from self-inflicted damage. First of all, all the motors have been kinematically constrained via *limit switches* so that none of the motors (except the spindle) can make more than a full revolution. Among other things, these constraints protect the electrical connections. The only self-collision that is possible (assuming that the robot goes past the limit switches) in ED I is between the spindle motor and the first link. In order to cushion against such an accidental self-collision, we have installed hydraulic shock absorbers on the first link. Other safety issues are discussed in detail in Section 5.

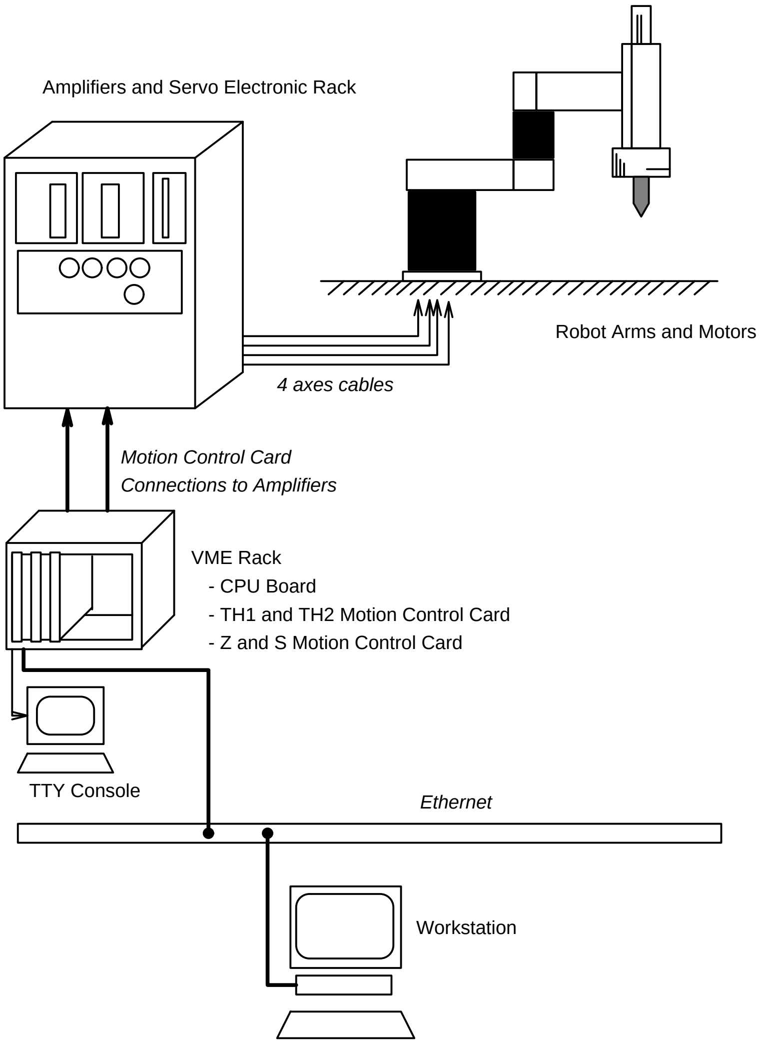


Figure 4: **Lay out of the ED I Robot System.**

2.2 Electronic Hardware

The robot is operated by a digital controller which consists of a CPU (Motorola MVME147S), and two motion control cards (Creonics 2-axis MCC), one for each pair of motors (i.e. the one for θ_1 and θ_2 axis-pair and the other for z and *spindle* axis-pair). The CPU and the MCC's communicate via a high bandwidth (40 Mbit/sec) VME bus. (See figure 2.) The digital controller receives commands from a workstation running the interface (a real time operating system and a C/UNIX programming environment) and generates appropriate commands for the actuators (the joint motors). In addition to the actuators that the motion control cards control, we also have *servo amplifiers for the actuators*, *digital encoders* and *encoder and drive power supplies* in order to run a closed-loop servo control scheme on the motion control cards.

Clearly, the heart of the digital control of the robot resides in the Creonics motion control cards. The VME bus only facilitates transfer of data between the master CPU and the multiple "slaves" (MCC's) without contention by means of a central arbiter. The master communicates with the MCC's in order either (1) to cause motion, set and modify parameters, etc., using *direct commands* or (2) to request status information using *request for reply commands*. We shall not delve into a lengthy discussion of the VME bus or the Motorola master CPU here as additional informations can be found in [36] and [29].

Bus	
Manufacturer	Motorola
Model	VME
Bandwidth	40 Mbit/sec

Table 6: **VME Bus Characteristics**

CPU Board	
Manufacturer	Motorola
Model	MVME147S
Microprocessor	Motorola 68020 @ 20 MHz
RAM	4Mb
Connectivity	Ethernet and SCSI through a MVME718 module

Table 7: CPU Board Characteristics

2.2.1 Creonics MCC

The Creonics VMEbus Motion Control Card (MCC) is a microcomputer-based, 2-axis digital motion controller, designed to work in conjunction with the VMEbus standards. The Creonics MCC is ideal for ED I control applications as it provides for each axis a closed-loop control for point-to-point positioning with *velocity* and *acceleration* control. The closed-loop control uses a PID (proportional, integral and derivative) feedback control mechanism, thus providing enormous flexibility in choosing the dynamic behavior of the controlled system. The MCC interface also allows for incremental optical encoders that provide both position and velocity informations.

The additional advantages provided by the Creonics MCC for our applications are as follows:

Motion Control Card	
Manufacturer	Creonics
Model	VMEbus MCC
Microprocessor	Intel 80C186 @@ 12.5 MHz
Pulse Width Modulation Interface	Custom Motion Control IC
Control	Digital Feedback control with nested Proportional, Integral, Velocity and Feedforward servo loop
Sample and Update	1KHz for position and velocity loops
Positioning	32 bits $\pm 2.147 \times 10^9$ encoder edges
Velocity	32 bits, 0.015 to 1 million edges per second
Acceleration	32 bits, 15.259 to 3.28×10^{10} edges per second square
Encoder Edge Rate	1 MHz
Command Execution	2 msec or less
Data Transfer Buffer Access Time	460 nsec
Interrupt Acknowledgement	510 nsec
Power	1.5 amps @@ 5V, 200mA @@ $\pm$ 12V from VMEBus

Table 8: Creonics Motion Control Cards Characteristic

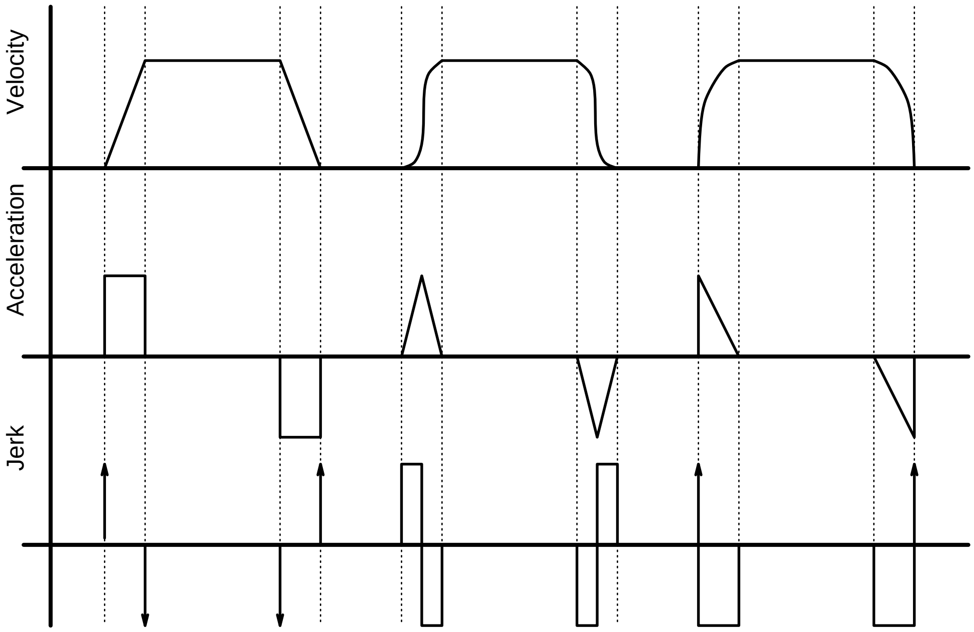
- Creonics provides an *exclusive automatic servo setup* which performs a quick and easy servo tuning, thus making the system ideal for the novice users.
- Its *direct tracking* capability allows complex motion profiles (for velocity and acceleration) to be computed and loaded by master CPU program in real time.
- Real time variable gain; four programmable software timers; packet type command and communication structure, etc.

The MCC's provide memory-mapped I/O by allowing indirect access to the 256 byte Command/Response Buffer that appear as eight registers to the VMEbus master. Additional access is also provided to various control and status registers. All *commands* and *requests* are transmitted via these registers.

Some of the frequently used *direct commands* that generate motion are discussed below briefly:

- **Move Command**

The Move command causes a motion of the designated axis using one of the following three profile types: *Trapezoidal Motion Profile*, *S-Curve Motion Profile* and *Parabolic Motion Profile*.



	Trapezoidal	S-Curve	Parabolic
Motion Stress	Fastest	2x slower	2x slower
Motor Stress	Worst	Better	Best
Mechanical Stress	Worst	Best	Better

Figure 5: **Profiles for Move command**

For most of the applications, the Trapezoidal profile is used, as it provides the fastest move for a given acceleration and deceleration limit. However, often S-curve profile is used, at the expense of the optimal speed, in order to minimize the stress on the mechanical system. Parabolic profile has the advantage as it matches the torque-speed characteristics of most motors and thus optimizes the motor stress.

The Move command can either specify the *distance* to be moved (relative/incremental) or the *endpoint* to move to (absolute) at a designated speed, acceleration and deceleration.

- **Home Axis Command**

Initializes the designated axis by enabling feedback and performing the predefined homing sequence.

- **Track Command**

The Track command allows direct control of the designated axis by the host frontend processor. In this case, the host generates the move profiles by controlling velocity and acceleration dynamically.

- **Interpolated Track Command**

The Interpolated Track command, like the Track command, allows direct control of the designated axis by the host processor, where the axis positions are controlled by incremental position values and the move time. Unlike the previous command, there is no direct control over the velocity/acceleration profiles; instead, the MCC interpolates intermediate points in order to accomplish a smooth motion over the given time interval.

A double buffering scheme is used so that the host can generate and update the next incremental position value even before the previous track motion is complete.

- **Abort Motion Command**

Immediately stops any motion in progress without disabling feedback—useful for safe execution of basic operations.