Introduction to the Scorbot ER VII and the Eshed Robotec Pty. Ltd. Advanced Control Language (ACL)

Robert MAHONY,
Dep. Engineering, ANU, ACT, 0200, Australia.

The Department of Engineering, has two Scorbot-ER VII robotic manipulators and one Controller-A. There are 4 components in the system:

Scorbot-ER VII Manipulator arm: The Scorbot-ER VII arm is a 5 degree of freedom robotic arm designed for education purposes. The base/shoulder/elbow configuration is a standard industrial anthropomorphic manipulator configuration. The wrist mechanism is a two degree of freedom (Y-Z co-located axis) design. The gearing system is belt gear reduction followed by a harmonic drive gear driven by brush-commutated permanent magnet motors rated to 18A at 24VDC. This configuration ensures the system is relatively safe for operation outside a safety enclosure, however, the maximum workload that can be carried is 2kg (including the weight of the end effector).

Controller-A: The proprietary Scorbot controller has the ACL programming language burnt into EPROM. It has the potential to control 11 axes that are separated into 3 groups: Group-A is the first 6 axes corresponding to the Scorbot-ER VII and gripper, axes 1-5 are rated to 18A at 24V while axis 6 (for the gripper) is rated to 2A at 12V; Group-B is for peripheral devices, axes-7 & 8 are both rated to 18A at 24V for linear-slidebase, rotating-table or conveyer belt peripherals. Group-C axes are for a second peripheral device, axes 9 & 11 rated to 18A at 24V while axis 10 is rated to 2A at 12V. Only axes 1-6 have operable driver cards installed in the ANU controller unit.

Advanced Control Language: The controller-A unit is programmed in the advanced control language (ACL) developed by Eshed Robotec Pty Ltd. The ACL protocol allows both `direct` control commands and `edited` programs to be run. Direct commands are sent in serial ascii directly to the controller, edited programs are best written in an ascii text editor and downloaded. **ACL is not case sensitive.**

Advanced Terminal Software: The advanced terminal software (ATS) provides a good means of communication with the controller unit. In fact, any ascii serial connection could be used for direct communication with the controller unit, however, the ATS software offers backup and print facilities as well as a better command interface than is available in a standard serial channel.

i) **ATS.** The ATS software can be activated from the icon on the desktop, or from a DOS prompt window (change directory to the directory containing ATS and type `ATS`. Commands entered on the command line of the terminal are interpreted directly by the controller unit. Programs can be entered directly in edit mode using `edit` command.

ii) **ATS-offline**. An ascii text file can be downloaded directly to the controller unit using the download command in a DOS prompt

```
download /r /y filename.nbl
```

The .nbl file is written as though the commands were to be entered directly in the ATS.

Bug report: It appears that the download command fails to recognize the serial connection to the controller unless the ATS software has run first. When you wish to download data, first open a dos prompt window, CD into the working directory, start ATS from the command line (by typing ATS), exit ATS in the normal way without closing the DOS prompt, now you can execute the download command as written above.

1. Documentation:

There are four manuals produced by Eshed robotec:

Scorbot-ER VII User Manual: The User manual contains hardware and software specifications for the robot and controller systems. It contains a tutorial on operating the robot and on programming in ACL. In Appendix A it describes the joint controller design.

ACL reference guide: This manual contains comprehensive lists of ACL commands.

ATS reference guide: This manual contains an overview of the operation of the Advanced Terminal Software (terminal emulation software).

ATS-offline reference guide: This manual contains an overview of the operation of the offline capability of the ATS software. It is recommended that you use the command line version of the ATS software and this manual is unlikely to be of use.

2. Operation Protocol:

i) Check that the manipulator workspace is free of obstacles.

ii) Do not enter the robots safety range or touch the robot during operation.

iii) Make sure loose hair and clothing is tied back when you work with the robot.

iv) Verify that you can reach the red emergency button on the controller without difficulty and without passing within range of the manipulator. One person should always be in a position to abort control using the emergency switch during operation.

v) Switch on the controller - switch at rear right. Activate the motors (green switch).

vi) If the gripper is attached then open the gripper using the teach pendant.

vii) Start an ATS session.

viii) Home the robot using the command

HOME

Do not run the robot without homing. Details of the homing process can be found on pages 6-3 of the user manual.

3. Avoiding damage to the equipment:

i) Do not overload the manipulator arm. The combined weight of the workload and gripper should not exceed 2kg.

ii) Do not use physical force to move or stop any part of the robot arm.

iii) Do not drive the robot arm into any physical obstacle.

iv) Do not leave a loaded arm extended for more than a few minutes.

v) Do not leave any of the axes under mechanical strain for any length of time. Especially, do not leave the gripper grasping an object indefinitely.

vi) Do not drive the Scorbot axes continuously in one direction. Specifically, when using the ACL command: `SET ANOUT[n]= DAC` verify that the *DAC* value is in the range ±2500.

3.1 Parameters

Parameters are stored values of constants used by the controller-A unit. The value of a parameter can be seen by

`SHOW PAR (parameter #)`

The work envelope of the ER VII is shown in the following figure.

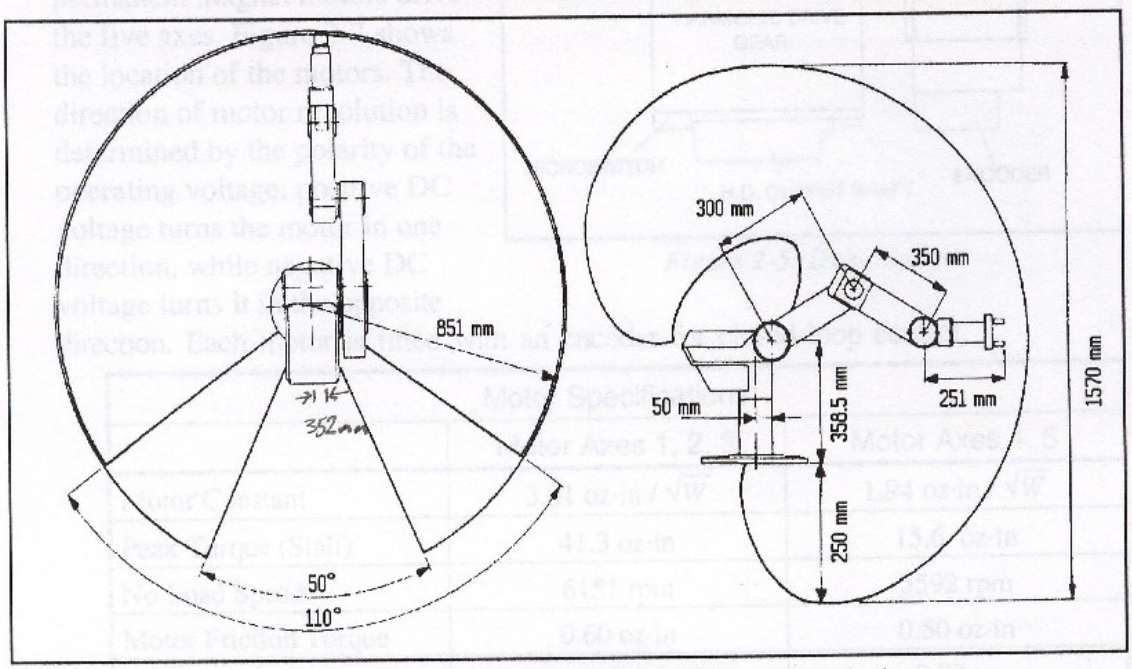


Figure 2-4: Operating Range With Gripper Attached

The home position of the robot is shown in the follow diagram
[Figure of Home Position to be inserted here.]

The following parameters for the Cartesian position of the robot are stored in the controller memory.

PAR 33	Number of encoder counts for +90_ rotation of axis 1
PAR 34	Number of encoder counts for +90_ rotation of axis 2
PAR 35	Number of encoder counts for +90_ rotation of axis 3
PAR 36	Number of encoder counts for +90_ rotation of axis 4
PAR 37	Number of encoder counts for +90_ rotation of axis 5
PAR 52	Value of encoder 1 at home position
PAR 53	Value of encoder 2 at home position
PAR 54	Value of encoder 3 at home position
PAR 55	Value of encoder 4 at home position
PAR 56	Value of encoder 5 at home position
PAR 92	Y -coordinate (offset from center along the Y -axis) of the gripper tip when robot is in the home position
PAR 93	X-coordinate of the rotation axis of arm link 2 when the robot is in the home position
PAR 94	Z-coordinate of the rotation axis of arm link 2
PAR 95	Length of the (upper) arm link from the first articulated joint
PAR 96	Length of the (lower) arm link from the second articulated joint.
PAR 97	Distance from the pitch axis to the tip of the gripper

Do not modify the controller parameters without asking the lecturer or lab technician.

4. Direct Control of the Robot Using the Teach Pendant

Unfortunately the teach pendant LCD screen is unreliable on the ANU controller unit. Appendix B is a reproduction of pages 5-4 to 5-7 of the user manual and use of the teach pendant.

5. Basic use of the manipulator

Servo control: The `CON` and `COFF` commands (as well as the `Control On/Off` key on the teach pad) turn on and off the servo control circuits on the robot. Movement is only possible with the control circuits on. Error events such as impact protection, overheating (thermic error) or use of the emergency switch will turn off the servo circuits.

Gripper Commands The gripper can be opened with command `OPEN` and closed with `CLOSE`. Alternatively the Teach pendant has an `OPEN/CLOSE` key.

Servo speed: The robot servos between points and along trajectories according to a trajectory profile. The maximum speed of each joint motion is governed by the following considerations.

Speed limits:

The actual robot speed setting for position servo-control is set using SPEED XX where XX is the percentage of maximum speed (1 to 100). Default speed should be 50. The present speed setting is seen by the command

SHOW SPEED

Note that when using the MOVES command (see below), the *duration* parameter takes precedence over SPEED setting. If duration is omitted, speed is governed by the *SPEED* setting.

Point-to-point control:
The robot plans a trajectory based on the velocity profile either in joint space (for the move,

bobat

individual desired speed is calculated based on standard PID error. Speed is saturated by the speed limit parameter (see above). The PID gains are factory set, and, while it is possible for users to change them, … don't!

Continuous path control:
The desired speed is calculated based on the standard PID error. Speed is saturated by the speed limit parameter (see above). The main difference between continuous path control (CP) and point-to-point control (PTP) is that in continuous path control many points are given to the controller, which then calculates a trajectory to move smoothly through the points.

6. Direct Commands

Several common commands are listed in the following table. See the ACL reference guide for a more detailed description of these commands and a complete list of all commands.

CON, COFF	Turns the servo control on and off.
A [*prog*]	Aborts a running program.
A	Immediately aborts all running programs.
DEFINE *var1 … var2*	Creates (defines) private variables.
DIM *var[n]*	Creates (defines) an array of *n* private variables.
DEFP *pos*	Defines position *pos*.
DIMP *vect[n]*	Defines a vector of *n* positions.
HERE *pos*	Records joint coordinates of current robot position in *pos*.
TEACH *pos*	Records Cartesian coordinates for a robot position.
SETPV *pos*	Records joint coordinates for a robot position.
SETPVC *pos coor var*	Changes on Cartesian coordinate of a previously recorded robot position (sets coordinate *coor* of position *pos* to *var*.
DIR	Displays all the names of all user programs.
LIST [*prog*]	Displays all lines of user program *prog*.
LISTP	Displays a list of all defined positions.
LISTPV *pos*	Displays the type of position and coordinates for *pos*.

MOVE *pos*	Moves axes to target position (*pos*).
MOVES *pvect p1 pn* [*duration*]	Moves the axes through a vector (*pvect*) of positions from point *p1* to point *pn*. (only works in joint coordinate space)
SPEED XX	Sets speed where XX is between 1 and 100.
RUN *prog*	Runs program *prog*.
MPROFILE PARABOLE A	Assigns a parabolic movement profile to control group *A*.
MPROFILE TRAPEZE A	Assigns a trapezoidal movement profile to control group *A*.
PVALC *pos coor*	Evaluates a joint position (*pos*) in Cartesian space.

7. Programming Commands

The direct commands listed above may also be inserted into a .dnl file and run as a program in ATS. A few commands that are more specifically used when writing an ACL program are listed below.

PROGRAM *prog*	Begins a program block.
END	Ends a program block.
FOR *var=n1* TO *n2*; ENDFOR	Creates a for loop.
IF *var1 oper var2*; ENDIF	Creates and if statement.

Appendix A: Coordinate systems for Scorbot-ER VII.

Coordinate Systems

The **SCORBOT-ER VII** can be operated and programmed in two different coordinate systems: Joint and Cartesian (XYZ) coordinates.

Cartesian (XYZ) Coordinates

The Cartesian, or XYZ, coordinate system is a geometric system used to specify the position of the robot's TCP (tool center point=tip of gripper) by defining its distance, in linear units, from the point of origin (the center bottom of its base) along three linear axes, as shown in Figure 6-1.

To complete the position definition, the pitch and roll are specified in angular units.

When robot motion is executed in XYZ mode, all or some of the axes move in order to move the TCP along an X, Y or Z axis.

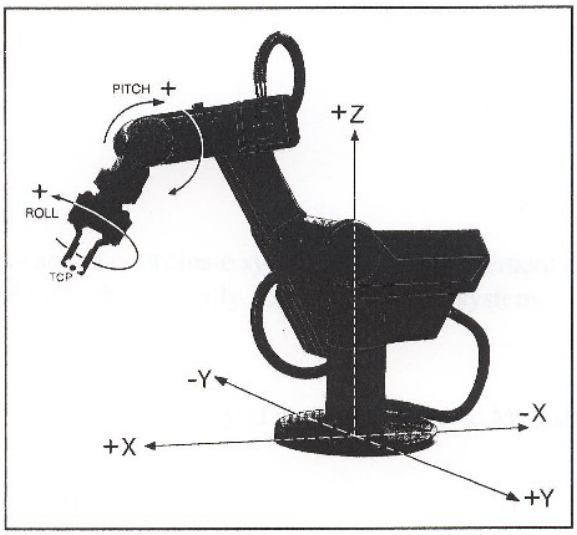


Figure 6-1: Cartesian Coordinates

Joint Coordinates

Joint coordinates specify the location of each axis in encoder counts. When the axes move, the optical encoders generate a series of alternating high and low electrical signals. The number of signals is proportional to the amount of axis motion; the controller counts the signals and determines how far an axis has moved. Similarly, a robot movement or position can be defined as a specific number of encoder counts for each axis, relative to the home position, or another coordinate.

When robot motion is executed in Joint mode, individual axes move according to the command.

If any peripheral devices are connected to the robotic system, the position of their axes is always stated in encoder counts.

Appendix B: Teach pendant key descriptions.

Accepts and/or executes the command which has been entered.

Starts execution of a program following a Run command.

A toggle key. Switches the command mode between Joints and Cartesian (XYZ).

When used following a numeric function, this key acts as a backspace function; it cancels the last numeric entry and moves the cursor one position to the left.

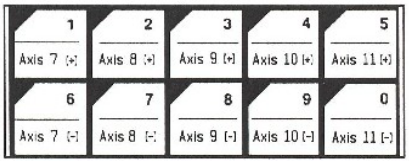
- Enables TP control of a specific axis group.

  Successively press for group A, group B, group C, and again for group A, and so on. When group C is displayed, enter the axis number on the numerical keys. Then press **Enter**.

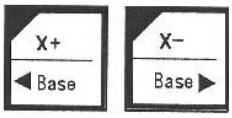
  The Record Position and Speed functions apply only to the currently selected group.

A toggle key. Enables (CON) and disables (COFF) control of the selected group.

- The **Axis** keys move axes 7 through 11 in two directions.

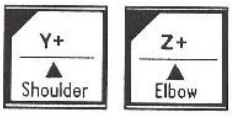


- The numeric keys are operative if one of the following functions has been activated: **Speed**, **Run**, **Record Position**, **Go Position**, **Group Select**.



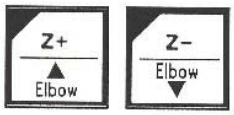
In Joint mode: the **Base/X** keys move the base axis in two directions.

In XYZ mode: the **Base/X** keys move the TCP (tip of gripper) along the X-axis; Y and Z coordinates do not change.



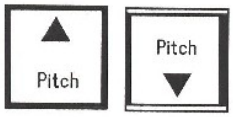
In Joint mode: the **Shoulder/Y** keys move the shoulder axis in two directions.

In XYZ mode: the **Shoulder/Y** keys move the the TCP (tip of gripper) along the Y-axis; X and Z coordinates do not change.



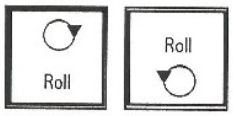
In Joint mode: the **Elbow/Z** keys move the elbow axis in two directions.

In XYZ mode, the **Elbow/Z** keys move the TCP (tip of gripper) along the Z-axis; X and Y coordinates do not change.



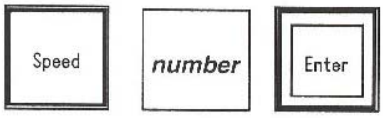
In Joint mode: the **Pitch** keys move the TCP (tip of gripper) up or down, without moving the other axes.

In XYZ mode: the **Pitch** keys move three axes (shoulder, elbow and pitch) in order to change the pitch angle without changing the position of the TCP (tip of gripper).



In both Joint and XYZ modes: the **Roll** keys move the roll axis in two directions.

A toggle key. Opens and closes the electrical gripper.



Sets the speed of manual axis movement of the current axis control group; that is, group A, B, or C. The speed is defined as a percentage (1-100) of maximum speed.

Press **Speed**. The current speed is displayed.

Press **Enter** to accept the displayed default speed. Or use the numerical keys to enter a different speed, and press **Enter**.

Defines and records a position.

Only numerical position names, of up to five digits, can be entered from the TP. The position is defined for the currently active group, and receives the current values of the axes in that group.

Press **Record Position**. Then press up to five digits for the position name. Then press **Enter** to record the position coordinates.

If you use a position name which has already been defined, the new coordinates will overwrite the existing ones.

This command is also used to record positions in a vector. The vector must first be attached to the teach pendant by means of the **ACL** command ATTACH.

Moves the axes to a target position.

Press **Go Position**. Then use the numeric keys to enter the position name. Then press **Enter** to execute the move.

In Joint mode: robot movement is by joints.

In XYZ mode: robot movement is linear.

To send the axes to their home position, enter the following commands:

Go Position 0 sends all the axes of group A to their HOME position.

Go Position 00 sends all the axes of group B to their HOME position.

Executes a program.

Press **Run**. Then press the program's identity number on the numerical keys. The program name will be displayed in brackets. Then press **Enter** to begin program execution.

The controller automatically assigns an ID number to each user program. The **ACL** command DIR lists the programs and their assigned (IDENTITY) number.

Aborts execution of all running programs. Stops movement of the robot and all peripheral axes.

The Display Panel

The LCD panel shows the current status of the controller, the current user command, and system messages.

A resident note shows the coordinates system currently active: **JOINTS** or **XYZ**.

Another resident note shows the currently active group: **A**, **B**, or the ***number*** of one of the independent axes in control group C.