Robotic Kinematics Laboratory.

The Esched Scorbot-VII is a five degree of freedom robotic manipulator. The controller unit is typical of commercial controller systems for robotic manipulators. Information on the robot, its programming environment and safety considerations (both for the robot and yourselves) is contained in the document

Overview of the Scorbot ER-VII.

Each robot is worth a considerable amount of money. Please take care of the equipment. We would prefer it is you didn't hurt yourselves either.

Part 1 Goals

The object of the laboratory is use the Scorbot with the custom pen holder to draw a design on a page of paper in less than 5 minutes.

Picture Requirements:
Each lab group is to design their own picture to sketch on paper. The picture must contain at least 5 letters of text - usually a name. Use of script fonts, underlining, borders, patterns, 3D-effects and perspective views to add to the interest and complexity of the design.

Pilot felt tip pens are available from the departmental secretaries.

Programming Requirements:
The design must be achieved in a single active cycle of the robot that takes less than 5 minutes to complete. The design must be achieved under automatic control of the robot via programming of the controller unit. The initial and final positions of the robot must be the 'homed' position. This is the position in which the robot is left after a homing routine.

Part 2 Initialisation of the robot:

i) Turn on the controller and motors.

ii) Open up a DOS prompt and change directory to `C:\ATS`

iii) Run ATS by typing ATS.

iv) Turn the controller on by typing `CON` in the ATS window. Alternatively the control can be activated or deactivated from the Teach-pendant by pressing *Control On/Off*.

v) Home the robot by using the `HOME` command in ATS environment. The robot can not be homed using Teach-pendant commands.

It is important that the robot is homed every time that the controller is turned on or after any situation in which it is possible that the encoder count registers are invalidated. This happens when any of the controller motor overrides or emergency switches are thrown as well as when the thermal overloads or impact shutdowns occur.

vi) After homing the robot is in the 'homed' position or 'parked home' position. Define a reference point at this position. Using the teach pendant:
Press: *Record position*
Press: *01*
Press: *Enter*
Using the ATS command line interface:
`DEFP 01`
`HERE 01`

When working on the worktable, return the manipulator to the parked home position (using the move command not the homing routine) to avoid knocking the manipulator arm.

The manipulator arm should always be left in the parked home position at the end of each lab session to avoid long term strain on the joint drive transmissions.

vii) There is a second home position, `0`, defined automatically in the controller memory during the homing routine. We refer to this home position as the 'active home' position. To see where this home position is use the teach pendant to move to the active home position
Press: *Go position*
Press: *0*
Press: *Enter*
or using the command line in ATS
`MOVE 0`
Return the robot to the 'parked home' position `01` you defined earlier.

Part 3 Downloading and running a program:

i) Activate and home the robot. Leave the robot in the parked home position.

ii) Sticky tape a piece of paper on the worktable with its centre approximately 500mm from the base axis of the robot.

iii) Put a felt pen in the pen holder. (You may want to move the robot to the active home position to make this easier.) Only use the Pilot felt tip pens that are available from the departmental secretaries (or identical pens that you buy yourself.) Remove the pen after the experiment.

iv) Ensure that you have saved the 'justSdat.dnl' and 'justS.dnl' files into the ATS directory on the computer. These data files are plain ascii text files and can be edited with any ascii text editor.

Have a look at the two files.

The data file justSdat.dnl first defines a new vector of trajectory points `PTS` with 104 points using the command
`DIMP PTS[104]`
Then it individually assigns joint coordinates to each point in the vector `PTS` using the command
`SETPV PTS[1]`
3832 ; Axis 1
6812 ; Axis 2
-6861 ; Axis 3
-13213 ; Axis 4
0 ; Axis 5
The semi-colon ';' indicates a remark and text after a semi-colon (on the same line) is not parsed by the controller unit. Joint axis references are given in encoder counts.

The program file justS.dnl defines the program `S`. The initial command
`PROGRAM S`
defines the program. The second line
`MOVE PTS[1]`
moves the robot to the initial point of the trajectory using the standard move command. This is done to avoid an initial fast transient that will cause the end-effector to bounce into the table. The motion command
`MPROFILE PARABOLE A`
is used to select a parabolic velocity profile for the motion. And finally the trajectory itself is tracked using
`MOVES PTS 2 104 1000`
moving along the vector variable `PTS` from element 2 to element 104 in 10 seconds. The program is terminated with the command
`END`

v) Download the data file onto the robot controller.
Make sure that you have exited ATS and are in the DOS terminal.
type 'DOWNLOAD /R /Y justSdat.dnl'
It will take a minute or so for the points to download due to the slow serial link. There may be an initial error message due to the `DIMP` command if the variable `PTS` is already defined in the controller memory.

Download the program file
type 'DOWNLOAD /R /Y justS.dnl'

You must run ATS at least once before you run the download routine to establish the serial link with the controller unit.

If an error occurs while downloading your data or program, check the REPORT.DLD file in the ATS directory. This will tell you more precisely where the error occurred. If the program or data that you are downloading already exists in the controller RAM you might get an error when downloading. This is OK as the programs should be overwritten if you use the /r /y modifiers with the download command. However, you may want to delete the program using the `REMOVE <prog>` *command and the data using the* `DELP <pvect>` *command within ATS prior to downloading a modified program and data just to make sure the controller is running the latest version.*

vi) Run ATS by typing ATS.

vii) Turn the controller on by typing CON.

viii) Has the main control unit of the robot been turned off. Was the motor switch turned off. If you are in any doubt as to whether the robot is properly homed and the encoder registers are correct then home the robot again.

ix) Move the robot down until the pen tip is only a centimetre or two off the centre of the paper. This is done to avoid an initial fast move from home to the paper that might lead to the end effector bouncing into the paper. Use the Teach-pendant to move the robot axis.

x) List programs in the controller unit using
`DIR`
to verify that program 'S' has been loaded.

xi) Run program 'S' by typing
`RUN S`

xii) Move the robot back to its home position using the `MOVE` command.

Part 4 Cartesian trajectories for letters:

The lab requires you to program the robot to draw letters. You can find suitable fonts on the web, or you can create your own. As a starting point the Hershey vector font is provided. Have a look at the web page for more details
`http://astronomy.swin.edu.au/~pbourke/other/hershey/`
The vector fonts (for the occidental characters) are contained in the data file `Hershey_occ.txt`. Each line of the file represents a different letter or symbol referenced by a number. There is a MATLAB program `Hershey2XY.m` provided that will decode the Hershey vector fonts into an unscaled Cartesian trajectory stored in a variable `data`.

Once you have font trajectories, you will need to translate and scale the data to correspond to the robot's coordinate system and units of measurement. You will need to fully understand the forward and inverse kinematics of the robot including units of measurement etc. to achieve the scaling of the trajectories. The final joint trajectories should be defined as a series of points specified in joint encoder counts. *(There may be ways to achieve the task working solely in Cartesian coordinates, however, it is a required part of the laboratory to work in the joint coordinates for the final trajectory specification.)* Use MATLAB as a scripting tool to process the font data.

Finally, take a look at justSdat.dnl and justS.dnl for an example of how to specify points and write a program in ACL code.

Lab Report.

A single lab report is to be completed by each lab group.

The report is due at the latest by Friday 10am, October 28.

The report should be a maximum of **ten** A4 pages, 10 point Times-Roman font. Your report need not be this long to gain full marks - I don't enjoy reading dross.

Your Report should contain the following parts:

Header and author citations: Provide a title. Give names and student numbers of lab partners.

Abstract: A short 1 paragraph abstract providing an overview of the report.

Kinematics of the Scorbot Give a full presentation of the forward and inverse kinematics of the Scorbot. Write the kinematics in terms of encoder counts with respect to set point offsets on each joint.

Picture Design and trajectory planning Provide full details of how you chose the design of your picture and how you took that design and converted it into a trajectory for the robot to use. Include a discussion of particular choices that you made to overcome failings in the hardware or software environments. I will be looking for evidence of how you designed the trajectory to obtain the cleanest trace on the paper. For example, once the basic form of trajectory is decided, you will need to decide how to quantize the trajectory in order that the velocity profile of the end effector best achieves a clean, wobble free sketch of your design. Should the pen ascend smoothly from the paper at the end of a line while still moving forward or should it stop forward motion first and then ascend cleanly.

Programming the Scorbot Provide full details of the approach you took to program the Scorbot.

Results and Discussion Include a copy of the page that you successfully drew. The final cycle should be observed and signed by either myself, Padma Chakma, Rob Gresham, or any other staff member who you can find when you are ready to do the cycle.

Discuss the results and comment on what problems you encountered and their causes as well as possibilities to overcome them.