4.1 ROBOT PROGRAMMING

According to the *consistent* performance by the robots in industries, the robot programming can be divided in two common types such as:

- Lead through Programming Method
- Textual Robot Languages

4.1.1 Lead through Programming Method:

During this programming method, the traveling of robots is based on the desired movements, and it is stored in the external controller memory. There are two modes of a control system in this method such as a *run mode* and *teach mode*. The program is taught in the teach mode, and it is executed in the run mode. The lead through programming method can be done by two methods namely:

- Powered Lead through Method
- Manual Lead through Method

a) Powered Lead through Method:

The powered lead through is the *common* programming method in the industries. A *teach pendant* is incorporated in this method for controlling the motors available in the joints. It is also used to operate the robot wrist and arm through a sequence of points. The playback of an operation is done by recording these points. The control of complex geometric movesis *difficult* to perform in the teach pendant. As a result, this method is good for *point to point* movements. Some of the key applications are spot welding, machine loading & unloading, and part transfer process.

b) Manual Lead through Method:

In this method, the robot's *end effector* is moved physically by the programmer at the desired movements. Sometimes, it may be difficult to move large robot arm manually. To get rid of it a *teach button* is implemented in the wrist for special programming. The manual leadthrough method is also known as *Walk Through method*. It is mainly used to perform continuous path movements. This method is best for spray painting and arc welding operations.

Textual Robot Languages:

In 1973, *WAVE* language was developed, and it is the first textual robot language as well. It is used to interface the machine vision system with the robot. Then *AL* language was introduced in 1974 for controlling multiple robot arms during arm coordination. *VAL* was nvented in 1979, and it is the common textual robot language. Later, this language was dated in 1984, and called as *VAL II*. The IBM Corporation has established their two own languages such as *AML*and *AUTOPASS*, which is used for the assembly operations.

Other important textual robot languages are Manufacturing Control Language (*MCL*), *RAIL*, and

Automatic Programmed Tooling *(APT)* languages.

Robot Programming Methods

There are three basic methods for programming industrial robots but currently over 90% are programmed using the teach method.

Teach Method

The logic for the program can be generated either using a menu based system or simply using a text editor but the main characteristic of this method is the means by which the robot is taught the positional data. A teach pendant with controls to drive the robot in a number of different co-ordinate systems is used to manually drive the robot to the desired locations.

These locations are then stored with names that can be used within the robot program. The co-ordinate systems available on a standard jointed arm robot are :-

Joint *Co-ordinates*

The robot joints are driven independently in either direction.

Global Co-ordinates

The tool centre point of the robot can be driven along the X, Y or Z axes of the robots global axis system. Rotations of the tool around these axes can also be performed

Tool Co-ordinates

Similar to the global co-ordinate system but the axes of this one are attached to the tool centre point of the robot and therefore move with it. This system is especially useful when thetool is near to the workpiece.

Work piece Co-ordinates

With many robots it is possible to set up a co-ordinate system at any point within the working area. These can be especially useful where small adjustments to the program are required as it is easier to make them along a major axis of the co-ordinate system than along a general line. The effect of this is similar to moving the position and orientation of the global co- ordinate system.

This method of programming is very simple to use where simple movements are required. It does have the disadvantage that the robot can be out of production for a long time during reprogramming. While this is not a problem where robots do the same task for their entirelife, this is becoming less common and some robotic welding systems are performing tasks onlya few times before being reprogrammed.

Lead Through

This system of programming was initially popular but has now almost disappeared. It is still however used by many paint spraying robots. The robot is programmed by being physically moved through the task by an operator. This is exceedingly difficult where large robots are being used and sometimes a smaller version of the robot is used for this purpose. Any hesitations or inaccuracies that are introduced into the program cannot be edited out easily without reprogramming the whole task. The robot controller simply records the joint positions at a fixed time interval and then plays this back.

Off-line Programming

Similar to the way in which CAD systems are being used to generate NC programs for milling machines it is also possible to program robots from CAD data. The CAD models of the components are used along with models of the robots being used and the fixturing required. The program structure is built up in much the same way as for teach programming but intelligent tools are available which allow the CAD data to be used to generate sequences of location and

process information. At present there are only a few companies using this technology as it is still in its infancy but its use is increasing each year. The benefits of this form of programming are:-

- Reduced down time for programming.
- Programming tools make programming easier.
- Enables concurrent engineering and reduces product lead time.
- Assists cell design and allows process optimization