USN | | | | | | | | | |

CMR INSTITUTE OF TECHNOLOGY, BENGALURU.
ACCREDITED WITH A+ GRADE BY NAAC

Internal Assessment Test 3 – DEC. 2022

Sub:	Automation and Robotics				Sub Code:	18ME732	Branch:	ME	
Date:	27.12.22	Duration:	90 min’s	Max Marks: 50	Sem / Sec:	VII/A&B		OBE	

Answer any FIVE FULL Questions

		MARKS	CO	RBT
1	Explain Potentiometer and Resolvers with a neat sketch	[10]	CO4	L2
2	Explain hydraulic actuators with suitable diagrams	[10]	CO4	L2
3	Explain tactile sensor and stepper motor with a neat sketch	[10]	CO4	L2
4	Define robot programming language and explain levels of Robot programming	[10]	CO5	L2
5	Explain Central issues in Robot Programming	[10]	CO5	L2
6.	Briefly discuss requirements in robot programming	[10]	CO5	L2

CI CCI HOD

USN | | | | | | | | | |

CMR INSTITUTE OF TECHNOLOGY, BENGALURU.
ACCREDITED WITH A+ GRADE BY NAAC

Internal Assessment Test 3 – DEC. 2022

Sub:	Automation and Robotics				Sub Code:	18ME732	Branch:	ME	
Date:	27.12.22	Duration:	90 min’s	Max Marks: 50	Sem / Sec:	VII/A&B		OBE	

Answer any FIVE FULL Questions

		MARKS	CO	RBT
1	Explain Potentiometer and Resolvers with a neat sketch	[10]	CO4	L2
2	Explain hydraulic actuators with suitable diagrams	[10]	CO4	L2
3	Explain tactile sensor and stepper motor with a neat sketch	[10]	CO4	L2
4	Define robot programming language and explain levels of Robot programming	[10]	CO5	L2
5	Explain Central issues in Robot Programming	[10]	CO5	L2
6.	Briefly discuss requirements in robot programming	[10]	CO5	L2

CI CCI HOD

Scheme of Evaluation

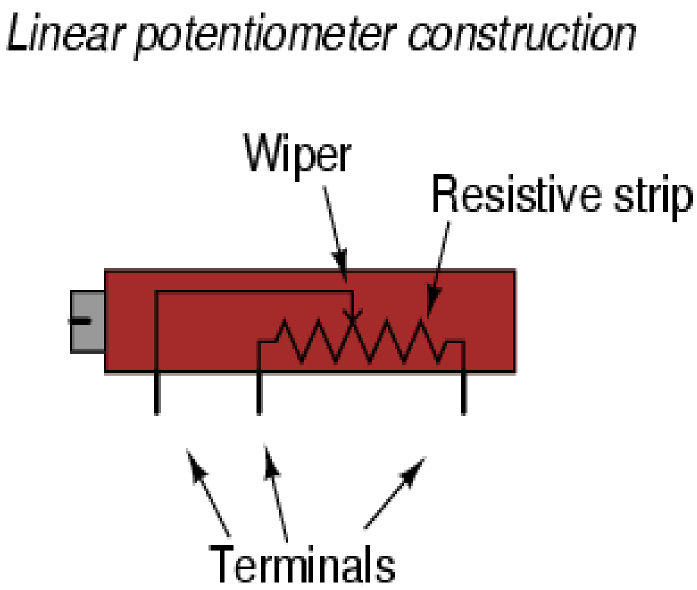
Question number	Particulars	Marks distribution
1.	Potentiometers Resolvers	5 marks 5 marks
2.	Hydraulic Actuators Explanation Diagram	 6 marks 4 marks
3.	Tactile sensor Stepper Motor	5 marks 5 marks
4.	Definition of Robot Programming Levels of Robot Programming	2 marks 8 marks
5.	Central issues in Robot Programming Any 5 issues	10 marks 5x2=10 Marks
6.	Requirements in robot programming 5 requirements	10 Marks 5x2=10 Marks

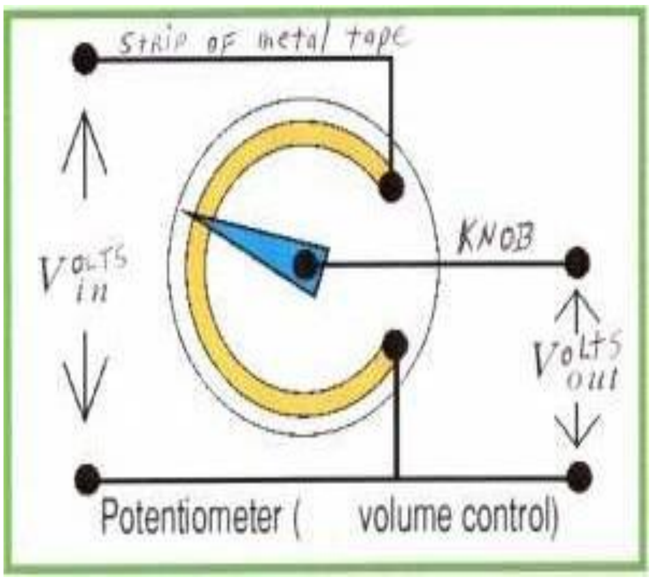
1.

- Potentiometers are analog devices whose output voltage is proportional to the position of wiper.
- Potentiometers offer a low cost method of contact displacement

measurement.

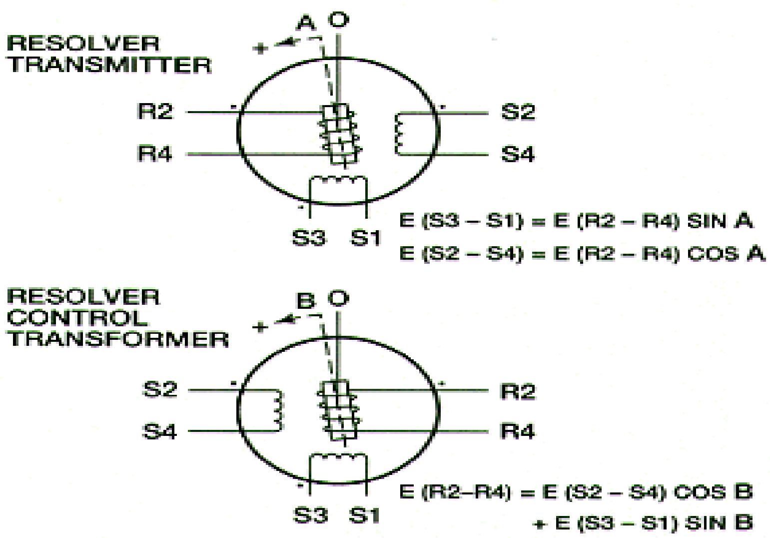
- Depending upon their design, they may be used to measure either rotary or linear motion.
- In either case, a movable slide or wiper is in contact with a resistive material or wire winding.The slide is attached to the target object in motion.
- A DC or an AC voltage is applied to the resistive material.
- When the slide moves relative to the material, the output voltage varies linearly with the total resistance included within the span of the slide.
- An advantage of potentiometers is that they can be used in applications with a large travel requirement.
- It is possible to use pots to provide a limited amount of feedback control in robots where high proportional resolution and accuracy are not required.





Resolvers

- A resolver is a type of rotary electrical transformer used for measuring degrees of rotation. It is an analog device whose output is proportional to the angle of rotating element with respect to fixed element.
- The primary winding of the transformer, located on rotor shaft, is excited by a sinusoidal electric current, which by electromagnetic induction induces current to flow through the secondary windings located on the stator.
- The two two-phase windings, fixed at right (90°) angles to each other on the stator, produce a sine and cosine feedback current by the same induction process.
- The relative magnitudes of the two-phase voltages are measured and used to determine the angle of the rotor relative to the stator.
- Since a resolver is a rotary transformer we must require an AC signal for excitation. If Dc signal is used there will be no output signal.

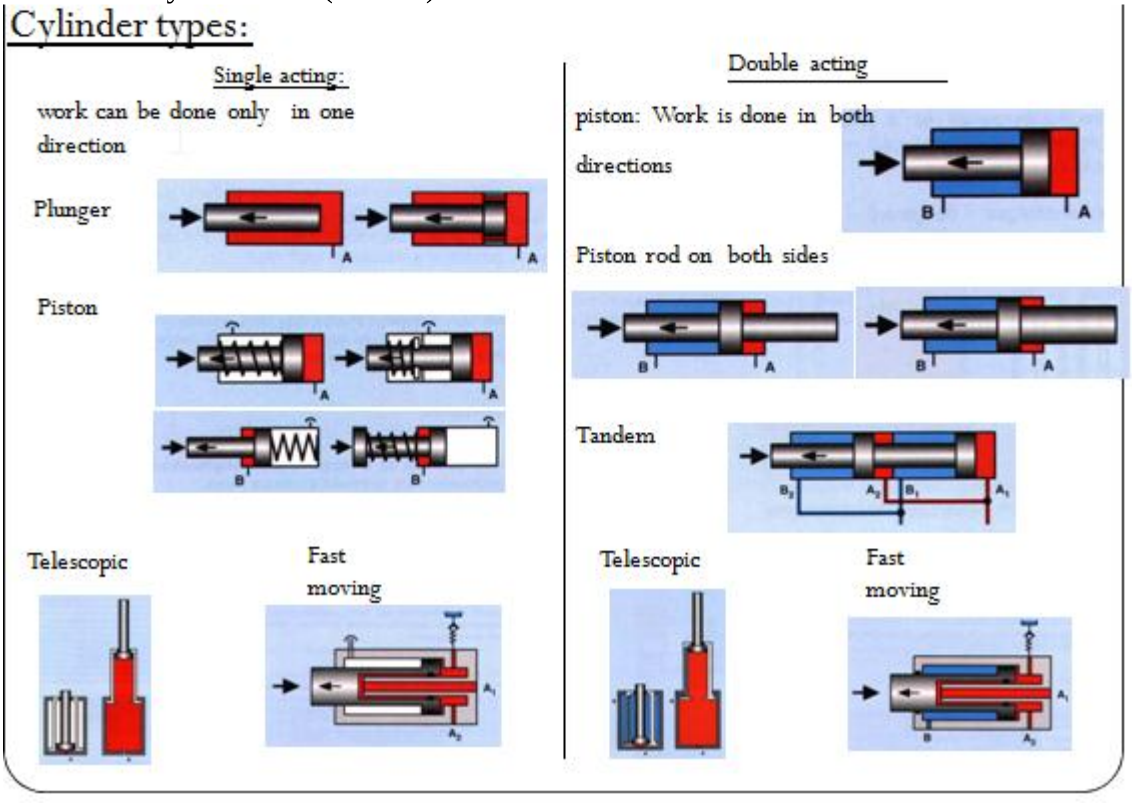


2. Hydraulic actuators

Hydraulic actuators allow a robot to move by the use of fluids moving under pressure through a series of valves by the use of pumps. The hydraulic fluids normally consist of oils which are reasonably non-compressible.

Hydraulic and Pneumatic actuators are classified as

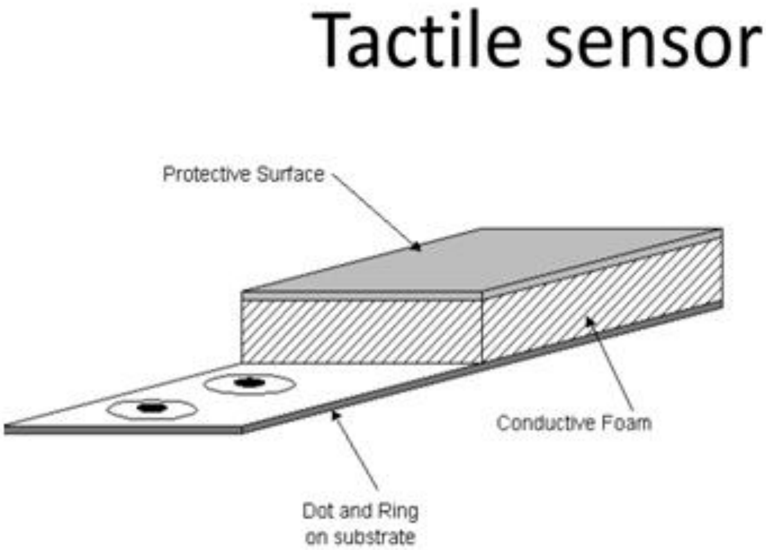
- Linear Actuators (Cylinders)
- Rotary Actuators (Motors)



- A Hydraulic motor is a mechanical actuator that converts hydraulic pressure and flow into torque and angular displacement (rotation).
- Several types of hydraulic motors:
  - Gear Motors
  - Vane Motors
  - Gerotor Motors
  - Radial Piston Motors

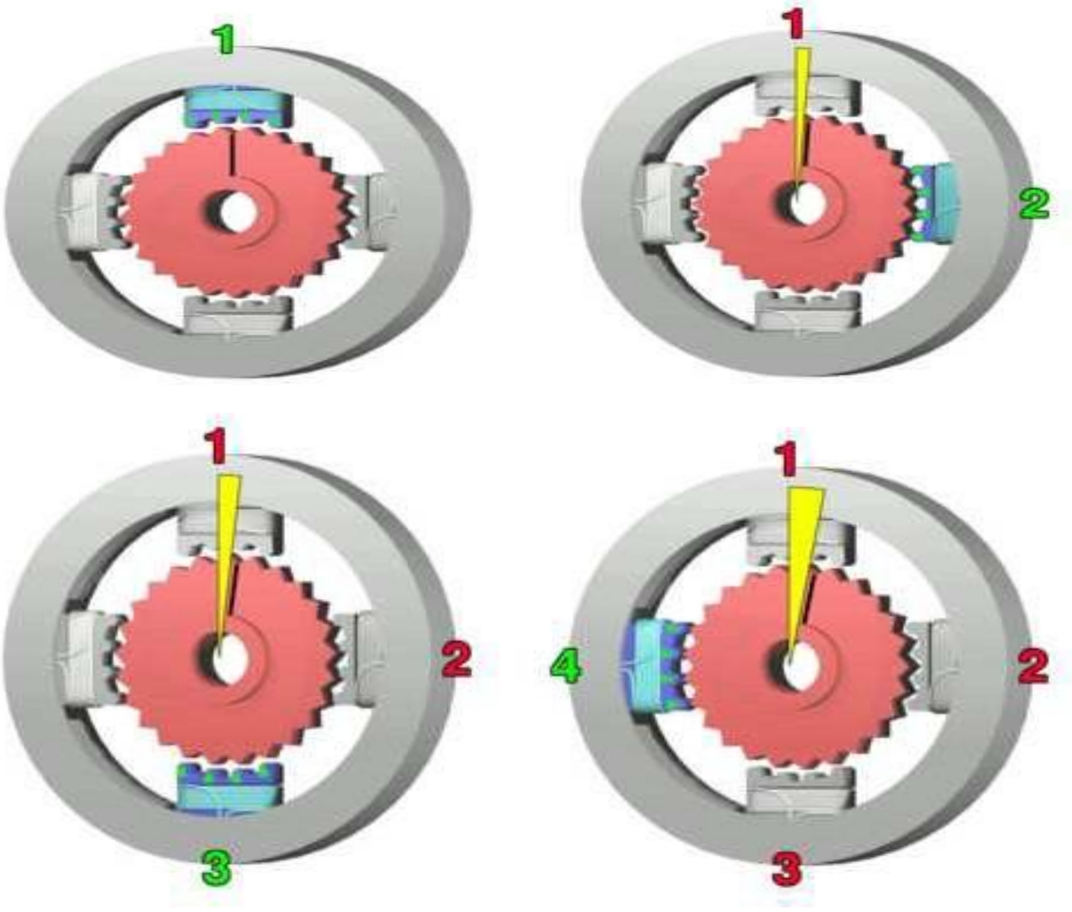
3. Tactile sensor

- A tactile sensor can measure any physical interaction with its environment.
- This sensor is capable to measure the parameters related to the contact of the sensor with an object.
- These sensors are designed according to the biological sense of cutaneous touch, which is able to detect the stimuli from mechanical stimulation, temperature, and pain.
- A tactile sensor will receive and respond to a signal and this signal could be a force or a physical contact.
- Basically a tactile sensor is a touch sensor that can provide information about the object which it.is in contact with it.
- The information about the object could be the shape, size, and type of material.



Stepper Motors

- When incremental rotary motion is required in a robot, it is possible to use stepper motors
- A stepper motor possesses the ability to move a specified number of revolutions or fraction of a revolution in order to achieve a fixed and consistent angular movement
- This is achieved by increasing the numbers of poles on both rotor and stator
- Additionally, soft magnetic material with many teeth on the rotor and stator cheaply multiplies the number of poles (reluctance motor)
- This figure illustrates the design of a stepper motor, arranged with four magnetic poles arranged around a central rotor
- Note that the teeth on the rotor have a slightly tighter spacing to those on the stator, this ensures that the two sets of teeth are close to each other but not quite aligned throughout.
- Movement is achieved when power is applied for short periods to successive magnets
- Where pairs of teeth are least offset, the electromagnetic pulse causes alignment and a
- small rotation is achieved, typically 1-2^{o}
- The top electromagnet (1) is charged, attracting the topmost four teeth of a sprocket.
- The top electromagnet (1) is turned off, and the right electromagnet (2) is charged, pulling the nearest four teeth to the right. This results in a rotation of 3.6°
- The bottom electromagnet (3) is charged; another 3.6° rotation occurs.
- The left electromagnet (4) is enabled, rotating again by 3.6°. When the top electromagnet (1) is again charged, the teeth in the sprocket will have rotated by one tooth position; since there are 25 teeth, it will take 100 steps to make a full rotation.
-



4. Robot Programming is a art of teaching a robot online or offline a series of tasks to be performed with required accuracy, velocity and repeatability.
Three levels of Programming

1. **Teach by showing**
2. **Explicit robot programming languages**
- Specialized manipulation languages
- Robot library for an existing computer language
- Robot library for a new general-purpose language
3. **Task-level programming languages**

1. **Teach by showing**
- Early robots were all programmed by a method that we will call **teach by showing,** which involved moving the robot to a desired goal point and recording its position in a memory that the sequencer would read during playback.
- During the teach phase, the user would guide the robot either by hand or through interaction with a **teach pendant.**
- Teach pendants are handheld button boxes that allow control of each manipulator joint or of each Cartesian degree of freedom.
- Some such controllers allow testing and branching, so that simple programs involving logic can be entered.
- Some teach pendants have alphanumeric displays and are approaching hand-held terminals in complexity.

2. Explicit robot programming languages

Ever since the arrival of inexpensive and powerful computers, the trend has been increasingly toward programming robots via programs written in computer programming languages.

- Usually, these computer programming languages have special features that apply to the problems of programming manipulators and so are called **robot programming languages** (RPLs).
- Most of the systems that come equipped with a robot programming language have nonetheless retained a teach-pendant-style interface also.
- Robot programming languages have likewise taken on many forms. We will split them into three categories:
- Specialized manipulation languages
- Robot library for an existing computer language
- Robot library for a new general-purpose language

3. **Task-level programming languages**

- The third level of robot programming methodology is embodied in **task-level programming languages.**
- These languages allow the user to command desired subgoals of the task directly, rather than specify the details of every action the robot is to take.
- In such a system, the user is able to include instructions in the application program at a significantly higher level than in an explicit robot programming language.
- A task-level robot programming system must have the ability to perform many planning tasks automatically.
- For example, if an instruction to "grasp the bolt" is issued, the system must plan a path of the manipulator that avoids collision with any surrounding obstacles, must automatically choose a good grasp location on the bolt, and must grasp it. In contrast, in an explicit robot programming language, all these choices must be made by the programmer.

4. Central Issues in Robot Programming

1. User interface

- A major motivation for developing an OLP system is to create an environment that makes programming manipulators easier.
- However, another major motivation is to remove reliance on use of the physical equipment during programming.

2. 3-D modeling

- A central element in OLP systems is the use of graphic depictions of the simulated robot and its work cell.
- This requires the robot and all fixtures, parts, and tools in the work cell to be modeled as three-dimensional objects

3. Kinematic emulation

- A central component in maintaining the validity of the simulated world is the faithful emulation of the geometrical aspects of each simulated manipulator.
- With regard to inverse kinematics, the OLP system can interface to the robot controller in two distinct ways.

4. Path-planning emulation

- In addition to kinematic emulation for static positioning of the manipulator, an OLP system should accurately emulate the path taken by the manipulator in moving through space.
- Again, the central problem is that the OLP system needs to simulate the algorithms in the employed robot controller, and such path-planning and -execution algorithms vary considerably from one robot manufacturer to another.

5. Dynamic emulation

- Simulated motion of manipulators can neglect dynamic attributes if the OLP system does a good job of emulating the trajectory-planning algorithm of the controller and if the actual robot follows desired trajectories with negligible errors.
- However, at high speed or under heavy loading conditions, trajectory-tracking errors can become important.

6. REQUIREMENTS OF A ROBOT PROGRAMMING LANGUAGE

1. World modeling

- Manipulation programs must, by definition, involve moving objects in three-dimensional space, so it is clear that any robot programming language needs a means of describing such actions.
- The most common element of robot programming languages is the existence of special **geometric types.**
- Given a robot programming environment that supports geometric types, the robot and other machines, parts, and fixtures can be modeled by defining named variables associated with each object of interest.

2. Motion specification

- A very basic function of a robot programming language is to allow the description of desired motions of the robot.
- Through the use of motion statements in the language, the user interfaces to path planners and generators of the style.
- Motion statements allow the user to specify via points, the goal point, and whether to use joint-interpolated motion or Cartesian straight-line motion.

3. Flow of execution

- As in more conventional computer programming languages, a robot programming system allows the user to specify the flow of execution—that is, concepts such as testing and branching, looping, calls to subroutines, and even interrupts are generally found in robot programming languages.
- More so than in many computer applications, parallel processing is generally important in automated workcell applications.

4. Programming environment

- As with any computer languages, a good programming environment fosters pro- grammer productivity.
- Manipulator programming is difficult and tends to be very interactive, with a lot of trial and error.
- If the user were forced to continually repeat the “edit-compile-run” cycle of compiled languages, productivity would be low.

5. Sensor integration

- An extremely important part of robot programming has to do with interaction with sensors.
- Integration with a vision system allows the vision system to send the manipulator system the coordinates of an object of interest.
- The interface to force-control capabilities, comes through special language statements that allow the user to specify force strategies.