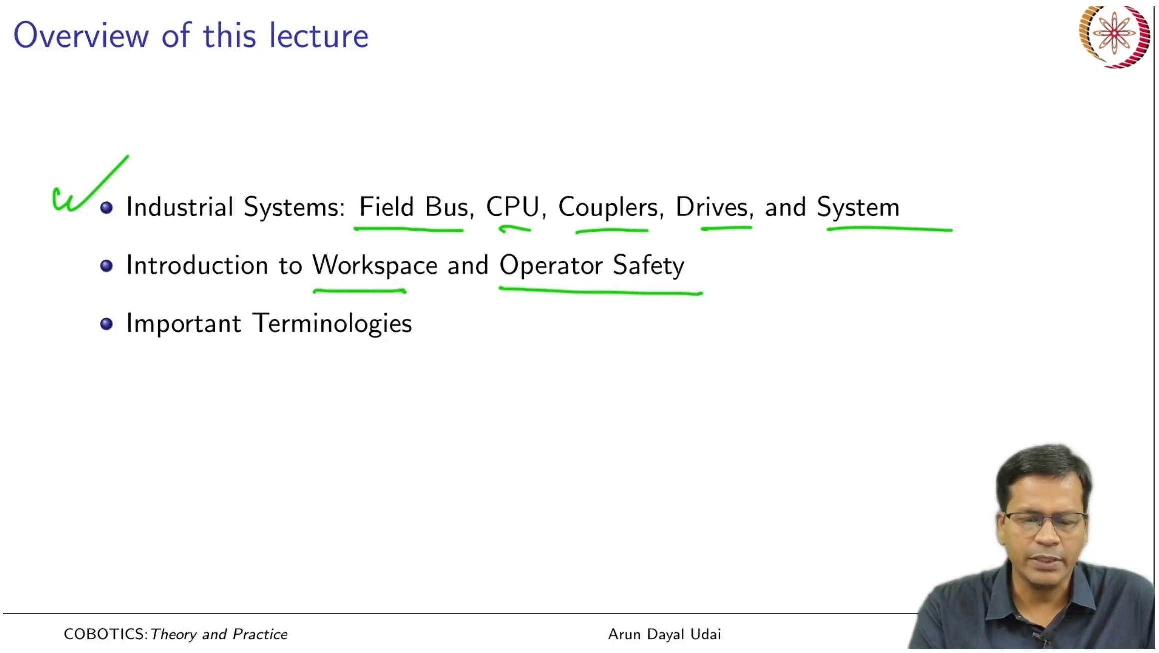
NPTEL Online Certification Courses
COLLABORATIVE ROBOTS (COBOTS): THEORY AND PRACTICE
Dr Arun Dayal Udai
Department of Mechanical Engineering
Indian Institute of Technology (ISM) Dhanbad
Week: 02
Lecture: 10

Industrial Field Bus, Drives and Devices, and Safe Workspaces

Welcome to the last lecture of this module, which is Actuators, Sensors, and Safe Workspaces. In this lecture, we will confine ourselves to Industrial Field Bus, Drives and Devices, Safe Workspaces, Safety Triggers, Workspace Monitoring, and Marking Forbidden Zones.

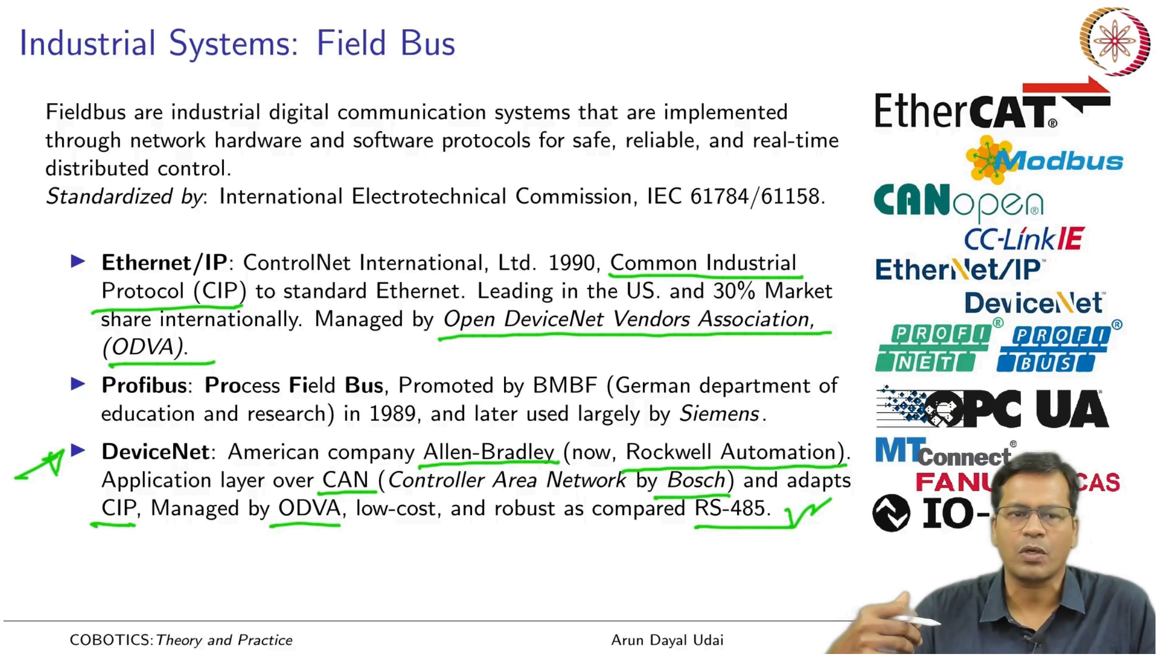


Overview of this lecture

- Industrial Systems: Field Bus, CPU, Couplers, Drives, and System
- Introduction to Workspace and Operator Safety
- Important Terminologies

COBOTICS: *Theory and Practice* Arun Dayal Udai

Specifically, we will be discussing Industrial Systems that cover Field Bus, CPU, Couplers, Drives, and Systems. We will discuss workspace and operator safety and various important terminologies, which will cover them all.



Industrial Systems: Field Bus

Fieldbus are industrial digital communication systems that are implemented through network hardware and software protocols for safe, reliable, and real-time distributed control.
Standardized by: International Electrotechnical Commission, IEC 61784/61158.

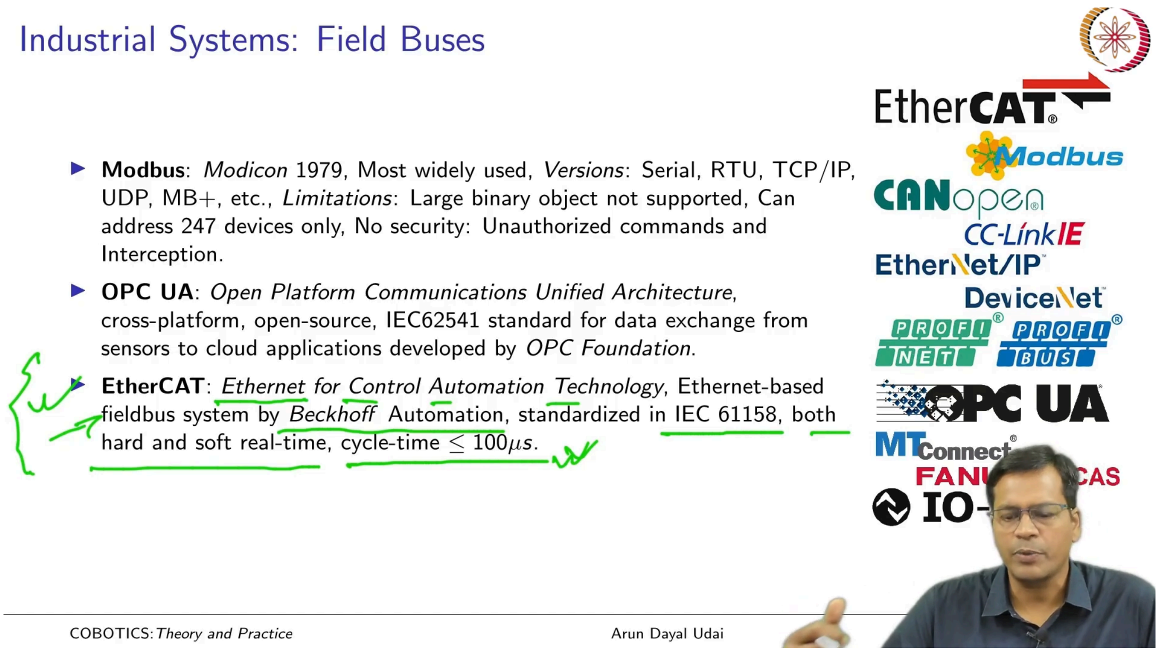
- **Ethernet/IP**: ControlNet International, Ltd. 1990, Common Industrial Protocol (CIP) to standard Ethernet. Leading in the US. and 30% Market share internationally. Managed by *Open DeviceNet Vendors Association, (ODVA).*
- **Profibus**: **Pro**cess **Fi**eld **Bus**, Promoted by BMBF (German department of education and research) in 1989, and later used largely by *Siemens*.
- **DeviceNet**: American company Allen-Bradley (now, Rockwell Automation). Application layer over CAN (*Controller Area Network* by *Bosch*) and adapts CIP, Managed by ODVA, low-cost, and robust as compared RS-485.

So, let us start with Industrial Systems, starting with the Field Bus. What are Field Buses, actually? Field Buses are industrial digital communication systems that are implemented through network hardware and software protocols for safe, reliable, and real-time distributed control, which is most widely used in the industry. The transmission can be as long as a kilometre at times. So, that distance can be very long, and it can be on-board systems also. Let us say you have a joint sensor and a controller; both can communicate with each other using such Field Buses. So, these are standardised by the International Electrotechnical Commission IEC 61784 and 61158.

Starting with the first one which is Ethernet or IP-based protocol. That is ControlNet International Limited 1990 came up with this idea, and this was a Common Industrial Protocol (CIP) to standard Ethernet. The standard Ethernet is the one that we are using on our desktop PC or laptop. This is meant for the industrial systems, leading in the US, and 30 percent of the share internationally was there, and it was managed by the Open DeviceNet Vendors Association (ODVA). But this didn't last longer because of some safety concerns and its performance.

Profibus: Profibus is the common name for Process Field Bus (PROFIBUS). It was promoted by BMBF, that is the German Department of Education and Research, in 1989, and later it was largely used by Siemens. Siemens, you know, is a very good manufacturer of motors, actuators, drives, systems, PLCs, and many more. So, they have come up with Profibus, and most of their Siemens systems use Profibus.

DeviceNet is the next one, which is given by an American company, that is Allen Bradley. Allen Bradley, again, is now known as Rockwell Automation. They make PLCs. Again, they also make various other input-output cards and drive systems. So, DeviceNet is an application layer over the CAN bus system. Controller Area Network was initially developed by Bosch, and it adapts the CIP. CIP is the Common Industrial Protocol. It adapts that, and this, again, is managed by ODVA, the Open DeviceNet Vendors Association. ODVA is robust compared to RS485. RS485 was an earlier serial protocol, and it became very popular because of its simplicity. However, it has many safety and security concerns. That is the reason it was modified and adapted into different other forms.



Industrial Systems: Field Buses

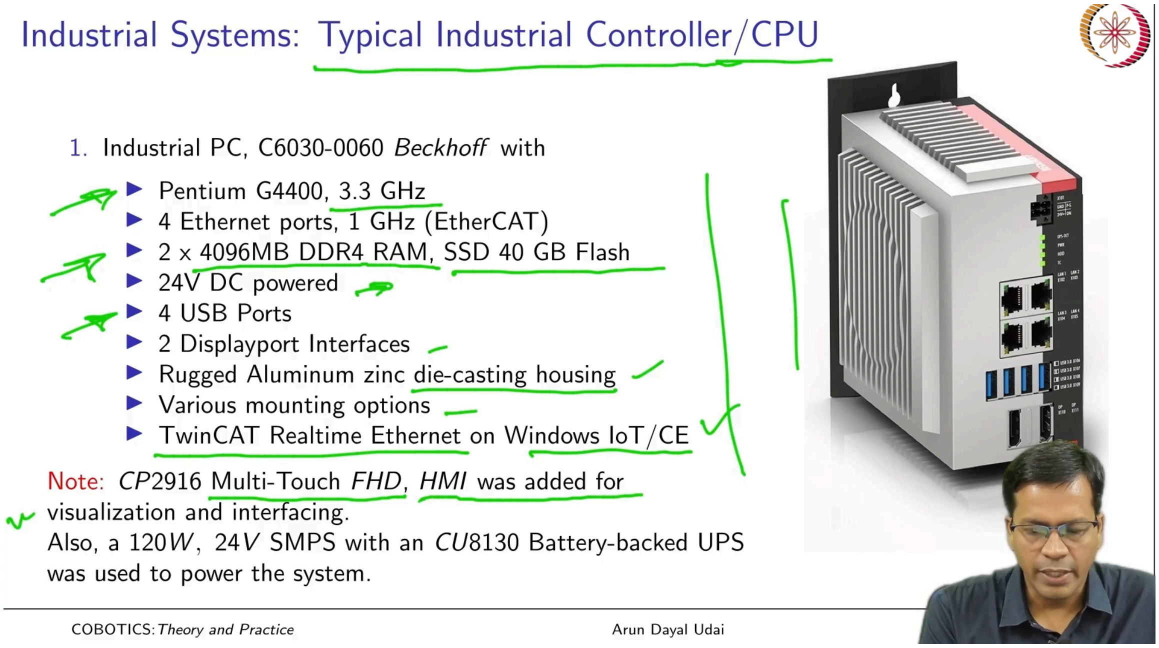
- **Modbus**: *Modicon* 1979, Most widely used, *Versions*: Serial, RTU, TCP/IP, UDP, MB+, etc., *Limitations*: Large binary object not supported, Can address 247 devices only, No security: Unauthorized commands and Interception.
- **OPC UA**: *Open Platform Communications Unified Architecture*, cross-platform, open-source, IEC62541 standard for data exchange from sensors to cloud applications developed by *OPC Foundation*.
- **EtherCAT**: *Ethernet for Control Automation Technology*, Ethernet-based fieldbus system by *Beckhoff* Automation, standardized in IEC 61158, both hard and soft real-time, cycle-time $\leq 100\mu s$.

COBOTICS: *Theory and Practice* Arun Dayal Udai

Now comes Modbus. It was introduced by Modicon in 1979. The most widely used versions of Modbus are Serial, RTU, TCP/IP, UDP, MB Plus, etc. It has many limitations, like a large binary format is not supported. That is the reason it can handle addresses up to 247 devices only. It lacks security; unauthorised commands can be given, and interceptions cannot be checked. This is the reason, due to safety, security, and limitations in its format, it is limited. But yes, it is still the most widely used in the industry because of its simplicity. This protocol is very popular.

The next one is OPC UA, which is Open Platform Communication Unified Architecture. This is open; it is not a proprietary format. It is a unified architecture for cross-platform communication. It is open source also, which is governed by the IEC 62541 standard and supports data exchange from sensors to cloud applications. It was developed by the OPC Foundation. So, this is one of the most advanced ones, the newer one, which is open source also. It is not a proprietary format. It would take some time to become very popular. It is this one. So, these symbols are for these standard formats. Those are just like logos and registered trademarks. They are used for different communication modes.

Here, EtherCAT is basically Ethernet for Control Automation and Technology: Ethernet and CAT. This is for Ethernet-based field bus systems. It was Proprietary of Beckhoff. Now, they have revealed the technology to the world, and there is quite a lot of adaptation of EtherCAT technology in multiple devices, even by Siemens, Allen Bradley, and many other manufacturers. That is standardised by IEC 61158. Both in hard and soft real-time, the beauty of this is it supports a wide range of devices with a very small cycle time. You see, the cycle time is less than 100 microseconds. It goes very fast, and that is the reason it can communicate in gigahertz. They are very fast, and not just fast, but also secure. And now, because of wide adaptation, it is very popular as well and most commonly used in cobots also for controlling their servo drives and for communicating with the sensors they have.



Industrial Systems: Typical Industrial Controller/CPU

1. Industrial PC, C6030-0060 *Beckhoff* with
   - Pentium G4400, 3.3 GHz
   - 4 Ethernet ports, 1 GHz (EtherCAT)
   - 2 x 4096MB DDR4 RAM, SSD 40 GB Flash
   - 24V DC powered
   - 4 USB Ports
   - 2 Displayport Interfaces
   - Rugged Aluminum zinc die-casting housing
   - Various mounting options
   - TwinCAT Realtime Ethernet on Windows IoT/CE

Note: *CP*2916 Multi-Touch *FHD*, *HMI* was added for visualization and interfacing.
Also, a 120*W*, 24*V* SMPS with an *CU*8130 Battery-backed UPS was used to power the system.

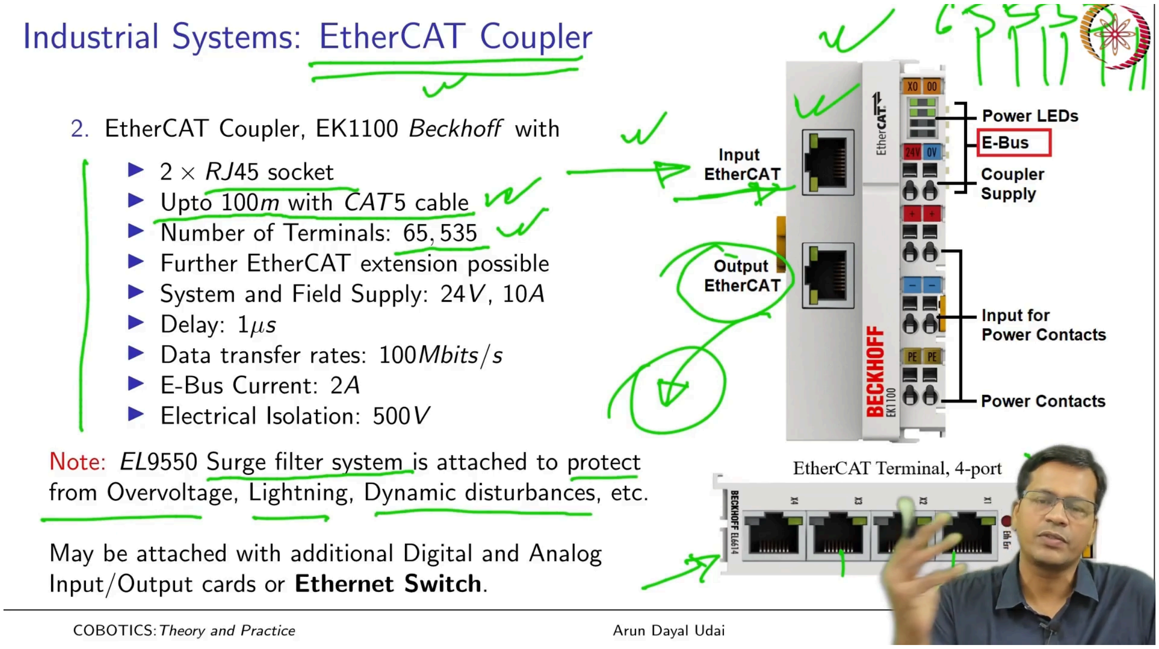
COBOTICS: *Theory and Practice* Arun Dayal Udai

So, now, what our system is also very popular in industrial systems, mostly for robots. Robots don't use a standard PC like our desktop or laptop. They are ruggedised PCs, which are known as industrial controllers or CPUs. They are just like standard computers. But they have quite a good amount of safety features and are ruggedised to be used in rugged environments like industries. See, the processors look quite old, but they are quite good enough to do real-time communications and to run standalone programs that are dedicated only to the robot. So, it is just one of the configurations that I have put. There are many such industrial controllers. I have just chosen one to demonstrate to you.

So, you see, it has standard DDR4 RAM, an SD card, 40 GB of flash, which can have an operating system like Windows Embedded, a 24-volt power supply, and four USB ports to take in pen drives and data. Display output ports are there to connect to external displays. Ruggedised aluminium zinc die-cast housing makes it very ruggedised. It can take up quite a good amount of load, unlike our laptop or desktop. Various mounting options are there. It can be mounted on Din-rails or within the controller cabinet of the robot.

It has TwinCAT real-time Ethernet, which is EtherCAT, and maybe on Windows IoT, which is CE. The multi-touch FHD (Full HD) monitor may be fitted. It can take up that, which may be used as an HMI for visualisation and interfacing. So, you can use a

120-watt, 24-volt SMPS with battery backup, just like a standard UPS, to supply this kind of CPU.



Industrial Systems: EtherCAT Coupler

2. EtherCAT Coupler, EK1100 *Beckhoff* with
   - $2 \times RJ45$ socket
   - Upto $100m$ with $CAT5$ cable
   - Number of Terminals: $65,535$
   - Further EtherCAT extension possible
   - System and Field Supply: $24V$, $10A$
   - Delay: $1\mu s$
   - Data transfer rates: $100 Mbits/s$
   - E-Bus Current: $2A$
   - Electrical Isolation: $500V$

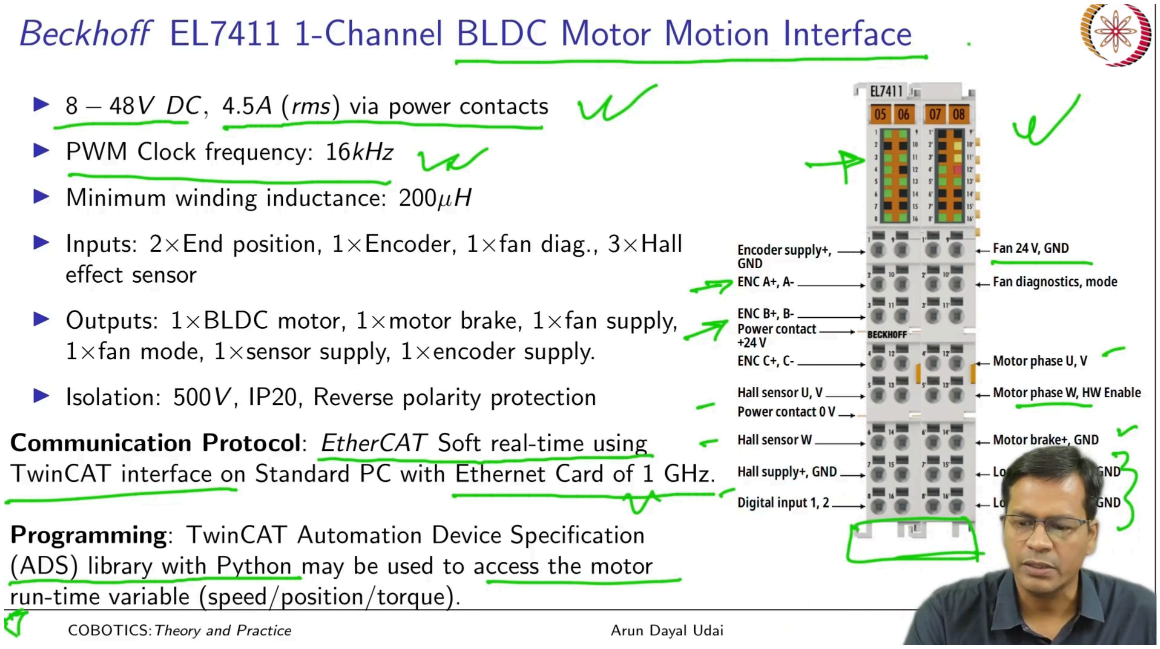
Note: *EL*9550 Surge filter system is attached to protect from Overvoltage, Lightning, Dynamic disturbances, etc.

May be attached with additional Digital and Analog Input/Output cards or **Ethernet Switch**.

COBOTICS: *Theory and Practice* Arun Dayal Udai

This is an EtherCAT Coupler. Basically, when you are doing all the robot kinematics on your computer, you have to transmit all the joint angles to the servo drives. So, that can happen in multiple ways. You can use devices like this. So, you provide this. So, you provide standard Ethernet with 1 GHz or higher frequency, which can be converted to EtherCAT, which can be used like an EtherCAT. EtherCAT is a software protocol in that case. so using that protocol, you can communicate with this device. This, in turn, can connect multiple cards after this. So, it can take up an EtherCAT signal, and it can connect other similar devices from this output also. They can be daisy-chained. I'll show you, or you can take in the input to an EtherCAT terminal over here, and you can have multiple such connections over here, just like a switch, any standard Ethernet switch. So, it looks like this. So, these are the parameters. It has two RJ45 sockets. One hundred meters, you see, with CATS 5 cable, it can take up such a long distance. The number of terminals that can be connected after this ranges up to 65,535. Further, an EtherCAT extension is also possible. So, each such device can connect to 65,535, and then after this, you can connect one more of these and connect another 65,535 devices.

You see, a huge amount of I/O cards, data cards, motor drives, and such EtherCAT-supported drives can be connected to this. A similar one is possible even with a Modbus coupler also. I have shown only the EtherCAT coupler. It can be a DeviceNet coupler, which would look very similar to this. This additionally may require a surge filter, which is attached to protect from overvoltage, lightning load or any dynamic disturbance, if it is there in the power supply, so as to protect this device.



Beckhoff EL7411 1-Channel BLDC Motor Motion Interface

- 8 – 48V DC, 4.5A (rms) via power contacts
- PWM Clock frequency: 16kHz
- Minimum winding inductance: 200μH
- Inputs: 2×End position, 1×Encoder, 1×fan diag., 3×Hall effect sensor
- Outputs: 1×BLDC motor, 1×motor brake, 1×fan supply, 1×fan mode, 1×sensor supply, 1×encoder supply.
- Isolation: 500V, IP20, Reverse polarity protection

Communication Protocol: *EtherCAT* Soft real-time using TwinCAT interface on Standard PC with Ethernet Card of 1 GHz.

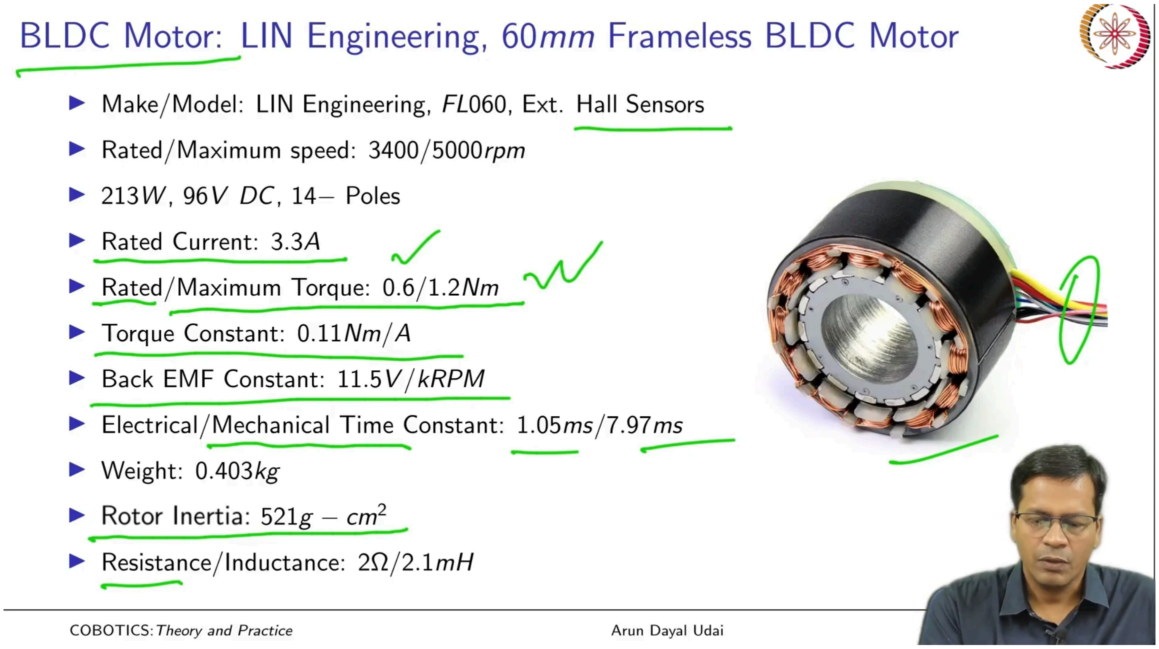
Programming: TwinCAT Automation Device Specification (ADS) library with Python may be used to access the motor run-time variable (speed/position/torque).

COBOTICS: *Theory and Practice* Arun Dayal Udai

The next comes a motion controller: BLDC Motor motion interface. This interface can be clubbed after this, so this can come here on the DIN rail. So, this, again, is a DIN rail-supported one. So, this is just a BLDC motion interface. It can run with a 48-volt DC. 4.5 ampere, so this is the maximum capacity this can handle with a PWM clock frequency up to 16 kilohertz. So, yes, you see, it has various connections to connect your BLDC motor. So, this is for the Hall Effect sensor, Hall Effect sensor, Hall Effect sensor, and motor phase wire U, V, and W goes here.

Motor brake supply can be given from here. This is a 48-volt supply that is for the motor. Fan supply is here. The fan may be connected over here so as to cool it and encoder feedback is fed here. So, this itself is hardware that has all the closed loops. Then, it needs the data from the EtherCAT coupler, which further transmits it to this to communicate the desired joint angle, position, velocity, and torque. So, accordingly, it can create multiple closed loops depending on the configuration. Communication protocol EtherCAT Soft real-time using TwinCAT interface is required for this, and an Ethernet card of one gigahertz is to be used with this. Programming can be done using API, that is TwinCAT ADS library, which supports Python. It supports various other programming languages also, like Visual Studio and all. It may be used to access the motor runtime variables to control speed, position, and torque.

So, you need to do everything on your computer and transmit the data to this real-time motion interface, which has all the closed loops high-frequency closed loops to control your motor. So, now, with a motor, it can be clubbed.

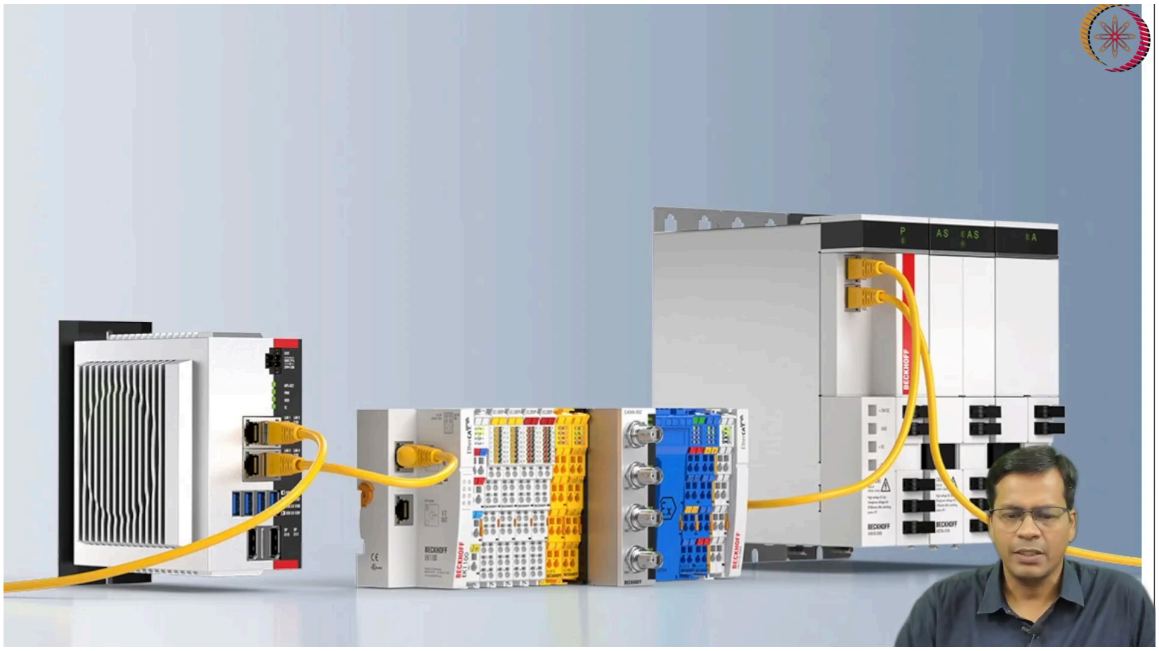


BLDC Motor: LIN Engineering, 60*mm* Frameless BLDC Motor

- Make/Model: LIN Engineering, *FL*060, Ext. Hall Sensors
- Rated/Maximum speed: 3400/5000*rpm*
- 213*W*, 96*V DC*, 14− Poles
- Rated Current: 3.3*A*
- Rated/Maximum Torque: 0.6/1.2*Nm*
- Torque Constant: 0.11*Nm*/*A*
- Back EMF Constant: 11.5*V*/*kRPM*
- Electrical/Mechanical Time Constant: 1.05*ms*/7.97*ms*
- Weight: 0.403*kg*
- Rotor Inertia: $521g - cm^2$
- Resistance/Inductance: 2Ω/2.1*mH*

COBOTICS: *Theory and Practice* Arun Dayal Udai

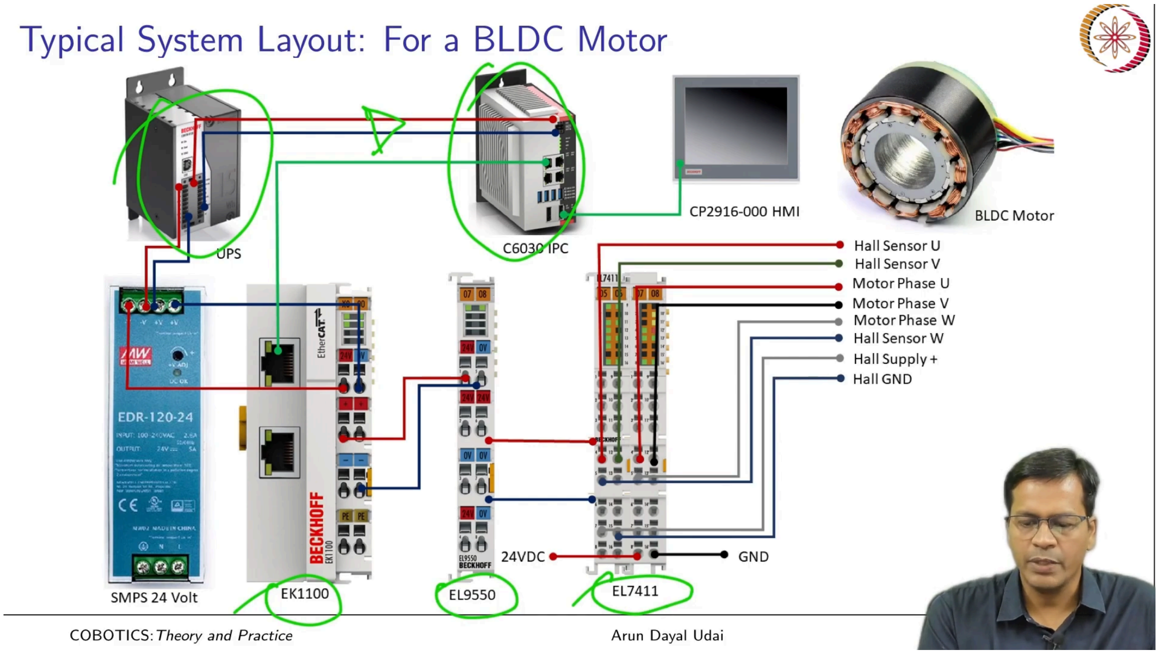
So, the motor may look like this. This is a simple BLDC motor with so many wires. So, it has wires for Hall Effect sensors. It has wires for the three phases here to supply these three phases to this motor. This is an open-frame motor so that it can be embedded inside the joint. So, these are the parameters. Parameters have almost everything that is required to control it. We will discuss various other parameters which are necessary to control such a device. A few of them are mechanical and electrical time constants back EMF constant, rotor inertia, resistance, and the inductance of the coil, which is here and rated current; the maximum current that can pass through this is 3.3 ampere. The maximum torque that it can handle is 1.2 newton meter and the rated torque is 0.6. So, accordingly, see torque constant, which is again very important. By using this, you can control the torque if you can control the current.



So, the EtherCAT Coupler finally couples all the cards that go here. So, all the cards come here like this. So, this is the first one you see; you have the CPU. Next is the EtherCAT coupler, and then you have all the cards that are attached one after the other. Next, finally, the servo drive can be an external servo drive. So, everything else can be connected in a daisy chain manner, as shown in this figure.

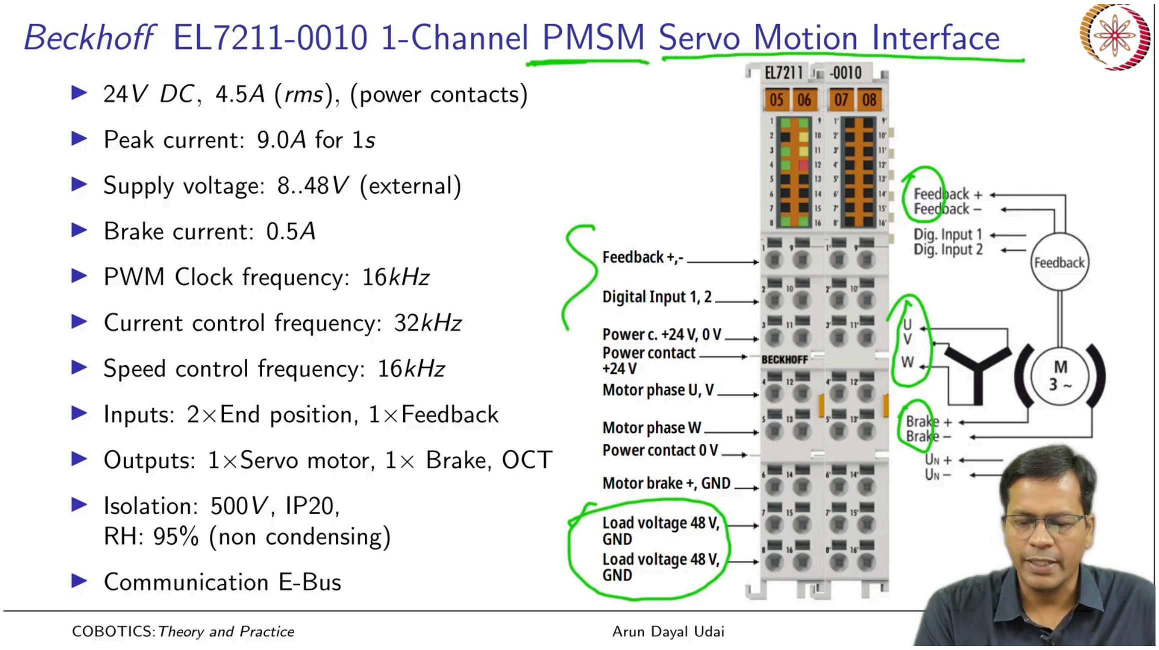
So, these are all the cards like this: BLDC servo motor motion interface, it comes here. Even all the cards that are shown here are clubbed one after the other. They are packed together in a manner like this, as I have shown here. The first thing is your IPC, which is the industrial PC. Next is your EtherCAT coupler. The third one is the power filter device, and then comes the other input-output card and the servo motor drives. So, everything else can go there. Altogether, one EtherCAT coupler supports 65535 such cards.

Typical System Layout: For a BLDC Motor



COBOTICS: *Theory and Practice* Arun Dayal Udai

So, overall, the layout may look like this: This is your CPU, the one that I have shown. This is the UPS to maintain a constant supply to this CPU and other devices. This is the EtherCAT coupler. This is the power surge protection device, and here comes your BLDC motor driver, and all the connections are shown here, which include motor phase wires U, V, W, and the Hall Effect sensor wires. If the motor has a brake wire, this servo drive also supports that.



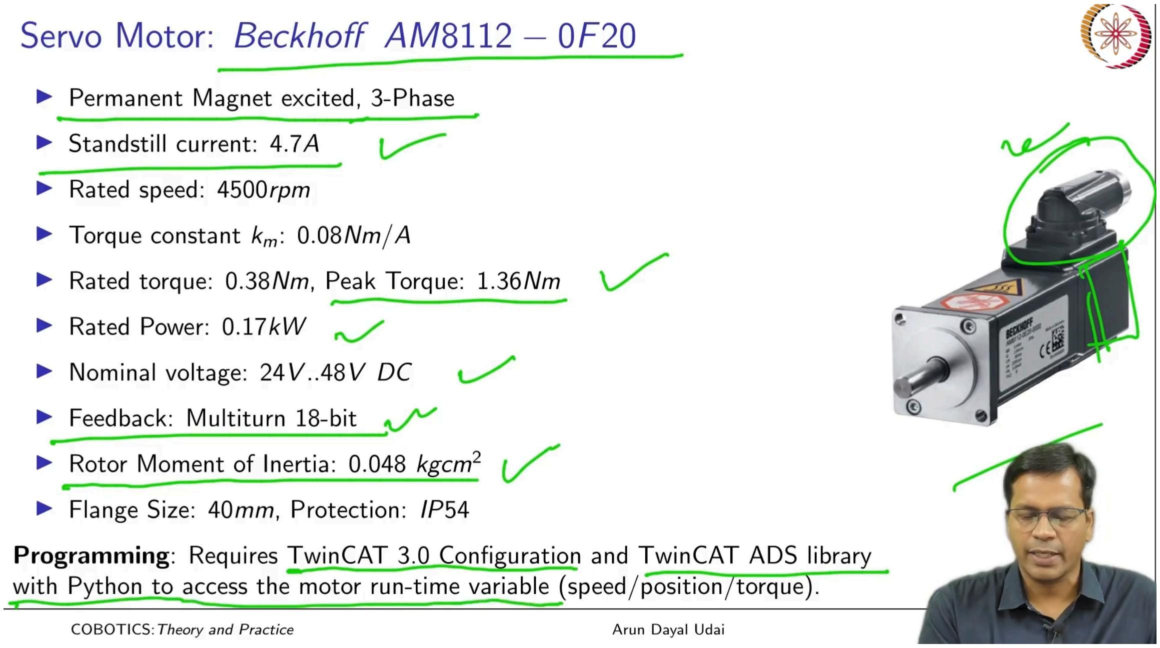
Beckhoff EL7211-0010 1-Channel PMSM Servo Motion Interface

- 24V DC, 4.5A (rms), (power contacts)
- Peak current: 9.0A for 1s
- Supply voltage: 8..48V (external)
- Brake current: 0.5A
- PWM Clock frequency: 16kHz
- Current control frequency: 32kHz
- Speed control frequency: 16kHz
- Inputs: 2×End position, 1×Feedback
- Outputs: 1×Servo motor, 1× Brake, OCT
- Isolation: 500V, IP20,
  RH: 95% (non condensing)
- Communication E-Bus

COBOTICS: *Theory and Practice*   Arun Dayal Udai

Similar is an arrangement for a permanent magnet synchronous machine, that is, a PMSM motor, also known as an AC servo motor motion interface. In this case, it looks like this when you have U, V, and W for the phases and brake inputs, And you also have the feedback input from here, which may be a resolver or some kind of that. So, it can take up the data for feedback and create a closed loop.

This is the supply for the PMSM system. So this is basically a variable frequency drive, maybe. This again needs to be connected with a similar EtherCAT coupler because this is an EtherCAT device that communicates via eBus. So important parameters are here with which you can control your motor.



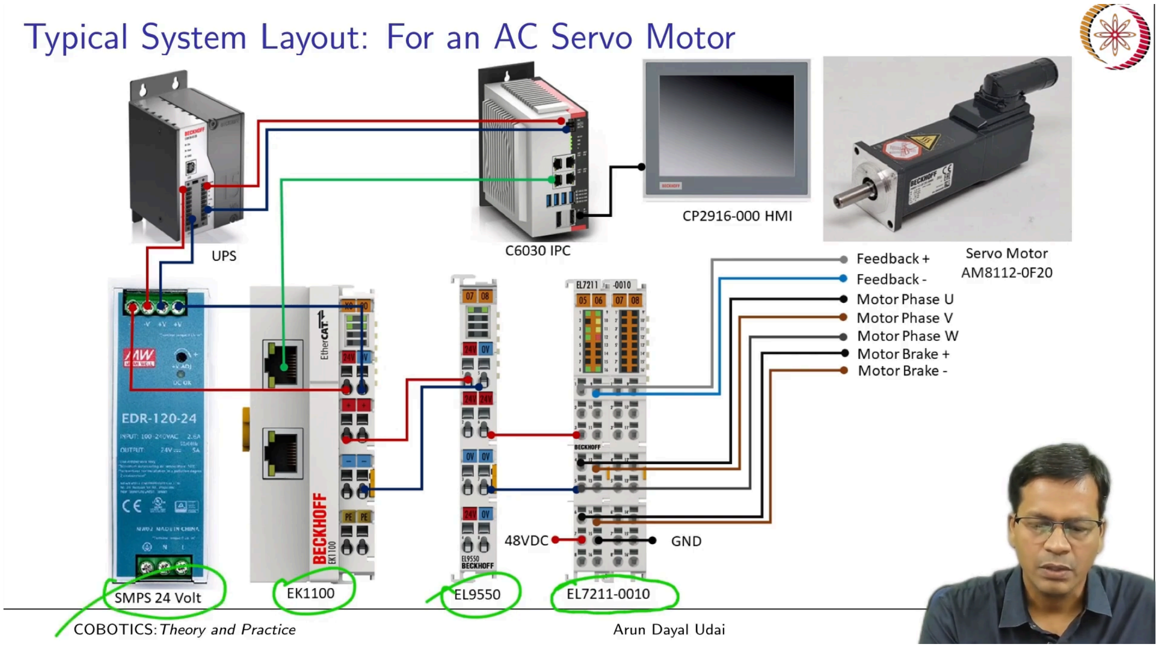
Servo Motor: *Beckhoff AM*8112 − 0*F*20

- Permanent Magnet excited, 3-Phase
- Standstill current: 4.7*A* ✓
- Rated speed: 4500*rpm*
- Torque constant k_m: 0.08*Nm*/*A*
- Rated torque: 0.38*Nm*, Peak Torque: 1.36*Nm* ✓
- Rated Power: 0.17*kW* ✓
- Nominal voltage: 24*V*..48*V DC* ✓
- Feedback: Multiturn 18-bit ✓
- Rotor Moment of Inertia: 0.048 $kgcm^2$ ✓
- Flange Size: 40*mm*, Protection: *IP*54

Programming: Requires TwinCAT 3.0 Configuration and TwinCAT ADS library with Python to access the motor run-time variable (speed/position/torque).

COBOTICS: *Theory and Practice* Arun Dayal Udai

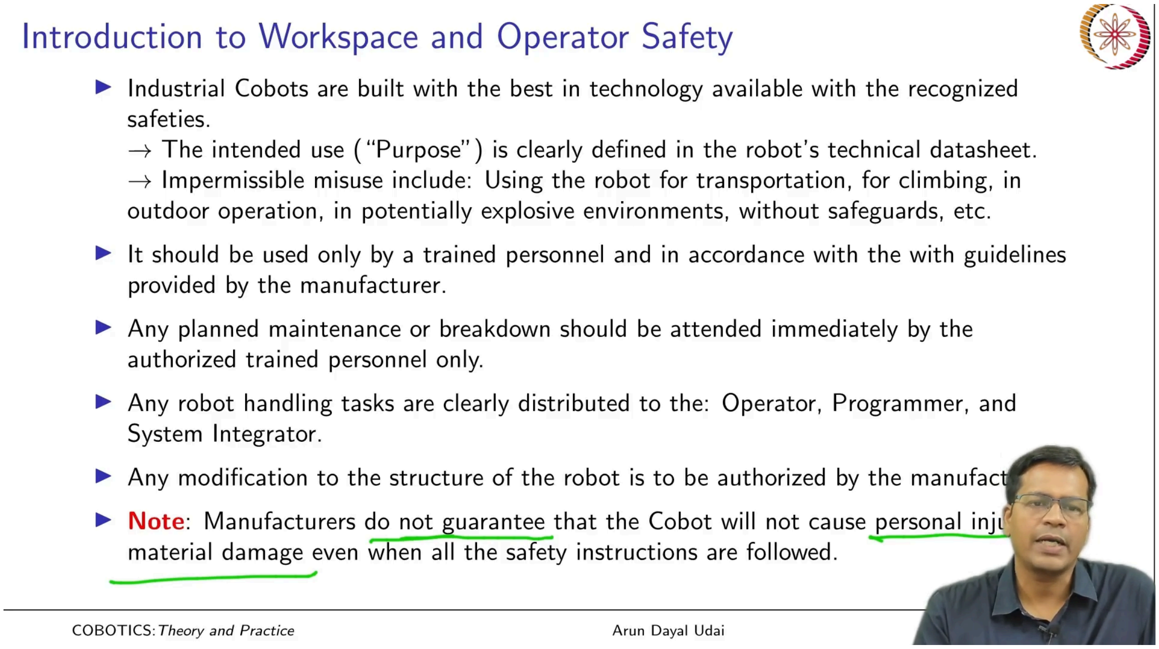
So again, a standard motor looks like this. This uses a single cable for data as well as power, U, V, W, as well as encoder values. Both are running from a single cable. So, this is a permanent magnet excited three-phase. Standstill current is given, so with that, you can choose the appropriate servo motor drive for this to run. This maximum peak torque is 1.36Nm, rated power maximum running voltage, and feedback is multi-turn 18-bit. So you may have some kind of encoder which is here. A rotor moment of inertia is required to precisely control such a motor programming requires. Again, TwinCAT configuration the TwinCAT ADS library with Python to access the motor runtime variable, which needs to be done with the drive that is shown here.



Typical System Layout: For an AC Servo Motor

COBOTICS: *Theory and Practice* Arun Dayal Udai

This is now can be connected like this. So here comes your AC servo motor drive, which is here, surge protector, EtherCAT coupler, power supply for 24-volt, continued power supply for using UPS, and this is the CPU and HMI. So, this may be a typical system to control such an AC servo motor.



Introduction to Workspace and Operator Safety

- Industrial Cobots are built with the best in technology available with the recognized safeties.
  → The intended use ("Purpose") is clearly defined in the robot's technical datasheet.
  → Impermissible misuse include: Using the robot for transportation, for climbing, in outdoor operation, in potentially explosive environments, without safeguards, etc.
- It should be used only by a trained personnel and in accordance with the with guidelines provided by the manufacturer.
- Any planned maintenance or breakdown should be attended immediately by the authorized trained personnel only.
- Any robot handling tasks are clearly distributed to the: Operator, Programmer, and System Integrator.
- Any modification to the structure of the robot is to be authorized by the manufact
- **Note**: Manufacturers do not guarantee that the Cobot will not cause personal inju material damage even when all the safety instructions are followed.

COBOTICS: *Theory and Practice* Arun Dayal Udai

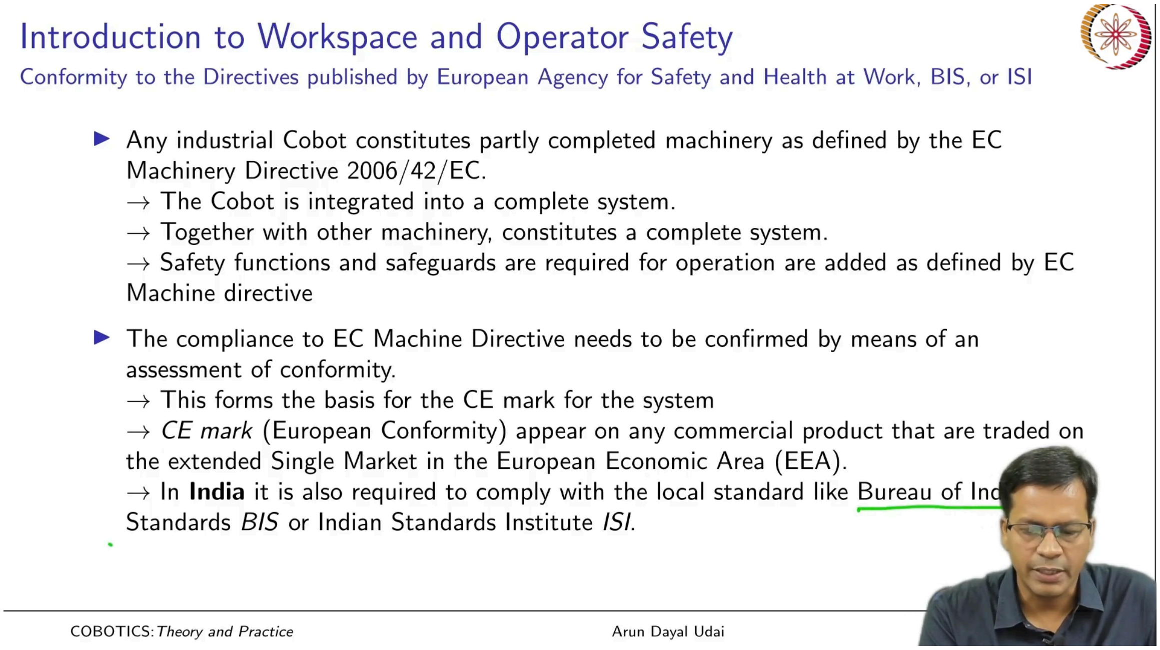
Now, let us go to workspace and operator safety. I have just introduced you to the workspace and operator safety here. Industrial Cobots are built with the best in technology available with the recognised safeties, unlike an academic setup, which may use a setup of cobots that may not be that rugged and is not safe to be used in industry with people all around.

The intended “Purpose” is clearly defined in the robot's technical data sheet. Suppose it is to be used for pick and place, for hand guiding, or maybe machine tending. So, a specific application is precisely given in the robot's technical data sheet. Impermissible misuse includes using the robot for transportation, climbing, or outdoor operation; potentially explosive environments without any safeguards are not recommended, and this is not an exhaustive list. There may be many such cases when it is not permissible for the robot to be used.

It should be used only by trained personnel and in accordance with the guidelines provided by the manufacturer. Normally, training is to be given beforehand. Any planned maintenance or breakdown should be attended to immediately by authorised, trained personnel only. Nobody can make any modifications to the hardware or to the program. Any robot-handling tasks are clearly distributed to the operator, programmer, and system integrator. The operator is the one who is actually on the assembly line or anywhere, who actually uses it. The programmer's jobs are done once the robot is put to use and handed over to the operator. The integrator basically does all the hardware jobs and configures the system to be used by the operator and the programmer.

Any modification to the structure of the robot must be authorised by the manufacturer. You are not supposed to make any modifications to the structure of the robot apart from just changing the tool end, which is documented in their manual as well.

Please note: Manufacturers do not guarantee that the Cobot will not cause any personal injury or material damage, even after all the safety instructions are followed. So, application to application, it depends quite a lot. Therefore, manufacturers do not guarantee such injuries if they are caused due to the mishandling or misuse of such robots.



Introduction to Workspace and Operator Safety

Conformity to the Directives published by European Agency for Safety and Health at Work, BIS, or ISI

- Any industrial Cobot constitutes partly completed machinery as defined by the EC Machinery Directive 2006/42/EC.
  → The Cobot is integrated into a complete system.
  → Together with other machinery, constitutes a complete system.
  → Safety functions and safeguards are required for operation are added as defined by EC Machine directive
- The compliance to EC Machine Directive needs to be confirmed by means of an assessment of conformity.
  → This forms the basis for the CE mark for the system
  → *CE mark* (European Conformity) appear on any commercial product that are traded on the extended Single Market in the European Economic Area (EEA).
  → In **India** it is also required to comply with the local standard like Bureau of Ind
  Standards *BIS* or Indian Standards Institute *ISI*.

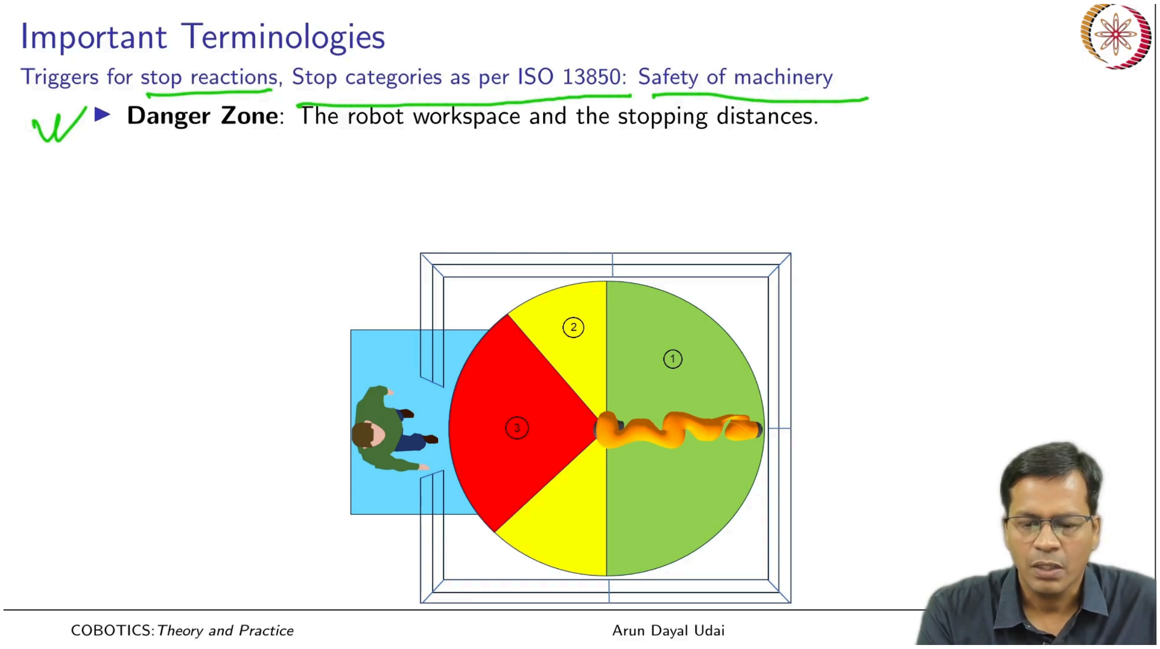
COBOTICS: *Theory and Practice* Arun Dayal Udai

So, yes, conformity to the directives published by the European Agency for Safety and Health at Work, based on ISI. Let us talk about that any cobot constitutes partly completed machinery as defined by the EC Machine Directive. It is just a partly completed machinery. It is not a complete machine. It has to be integrated with accessories, maybe a conveyor, a turntable, a linear rail, or some other accessories like a gripper or force sensor. So, it includes a lot of other peripherals. So, as per the EC Machine Directive, it is only partly completed machinery. The cobot is integrated into the complete system together with other machinery, which constitutes the complete system. Safety functions and safeguards are required. The robot in itself is not very safe, even if it is a cobot. It also needs some kind of safeguards for operation, which are added and defined by the EC Machine Directive.

Now, compliance with the EC Machine Directive needs to be confirmed by means of assessment of conformity. This forms the basis for the CE mark for the system. If you remember, the CE mark looks like this.

This is a very popular mark which you must have seen quite often, even on your laptop. The CE mark, which stands for the European Community, appears on any commercial product traded in the extended single market of the European Economic Area (EEA). Nowadays, even Indian products, if they intend to sell in European markets, must conform to the CE mark. In India, it is also required to comply with local standards like the Bureau of Indian Standards and the Indian Standards Institute (ISI).

Now, a few important terminologies are very important for you to know before you put any robot to use. Triggers for stop reaction stop categories as per ISO 13850, which is for the safety of machinery. What is the danger zone? A robot workspace and the stopping distance.



Important Terminologies

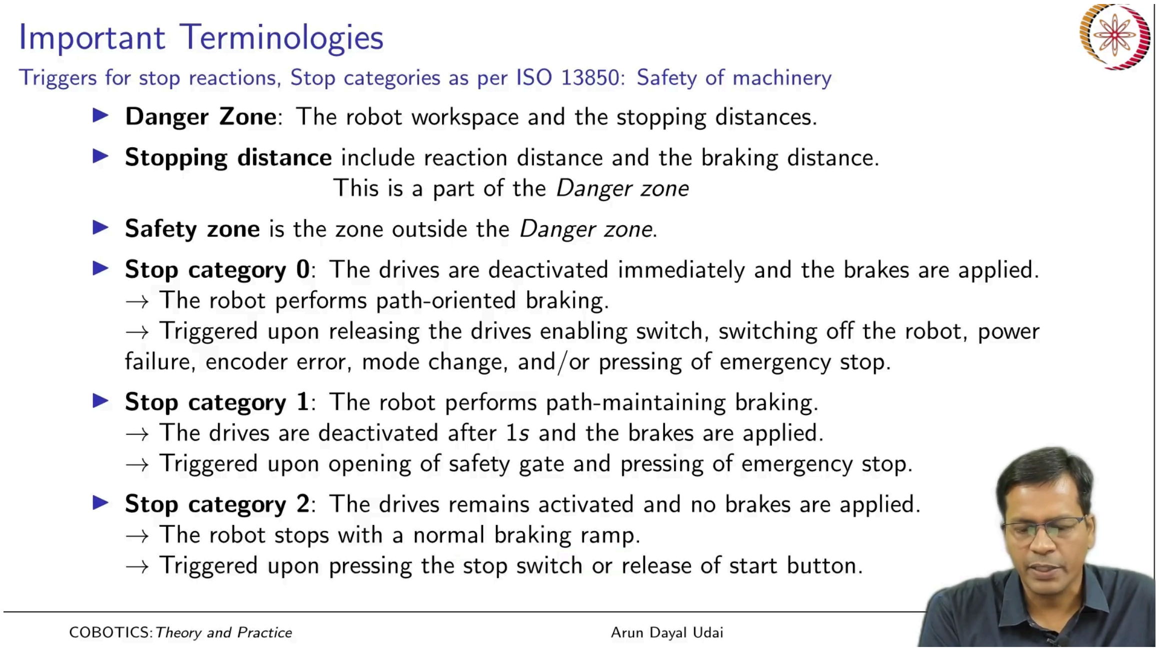
Triggers for stop reactions, Stop categories as per ISO 13850: Safety of machinery

- **Danger Zone**: The robot workspace and the stopping distances.

COBOTICS: *Theory and Practice* Arun Dayal Udai

Let me just display the figure here. If you can see this figure, you see there is a zone marked as one, which is the danger zone. That is where most of the time the robot works in that region, which is the danger zone. It also includes the stopping distance. The stopping distance is area 2, which is marked in yellow here. So, if you apply the brake, even in zone 1, it can go up to zone 2. So that also is a danger zone. The stopping distance includes the reaction distance and the braking distance. As soon as you apply the brake, what will happen? It will take some time to respond, and the brakes are applied.

Even after applying the brake, it may travel a little distance. So, this is part of the danger zone.



Important Terminologies

Triggers for stop reactions, Stop categories as per ISO 13850: Safety of machinery

- **Danger Zone**: The robot workspace and the stopping distances.
- **Stopping distance** include reaction distance and the braking distance.
  This is a part of the *Danger zone*
- **Safety zone** is the zone outside the *Danger zone*.
- **Stop category 0**: The drives are deactivated immediately and the brakes are applied.
  → The robot performs path-oriented braking.
  → Triggered upon releasing the drives enabling switch, switching off the robot, power failure, encoder error, mode change, and/or pressing of emergency stop.
- **Stop category 1**: The robot performs path-maintaining braking.
  → The drives are deactivated after 1s and the brakes are applied.
  → Triggered upon opening of safety gate and pressing of emergency stop.
- **Stop category 2**: The drives remains activated and no brakes are applied.
  → The robot stops with a normal braking ramp.
  → Triggered upon pressing the stop switch or release of start button.

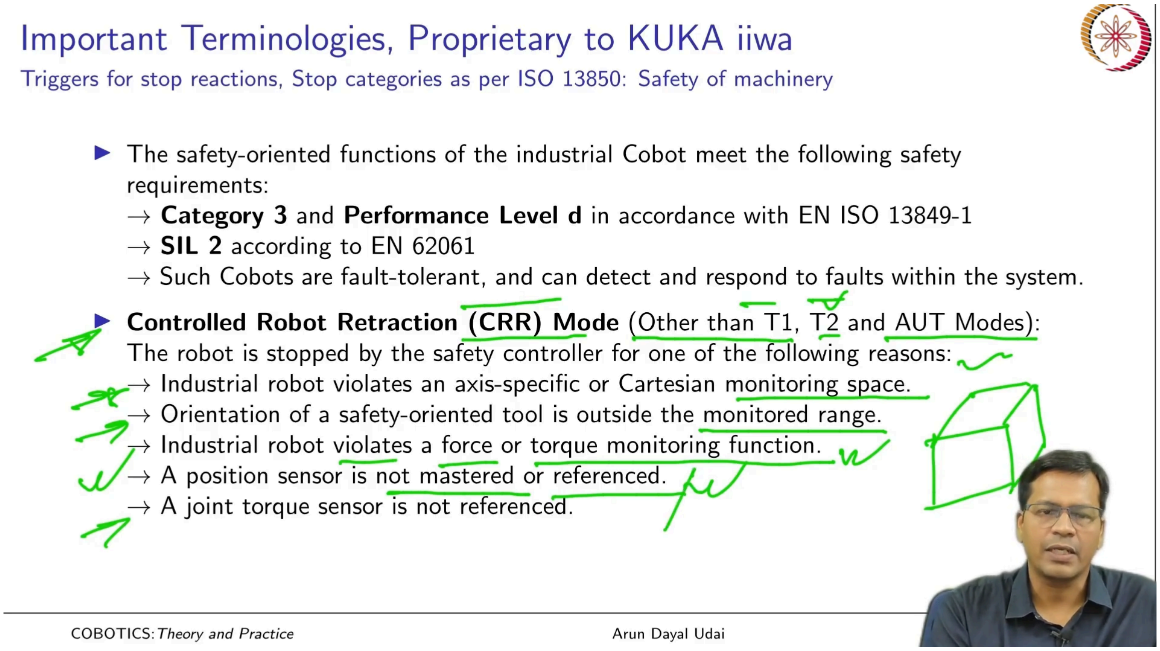
COBOTICS: *Theory and Practice* Arun Dayal Udai

The safety zone is the zone outside the danger zone, which is the red region that you can see. That is where the robot physically cannot go. So, that becomes the safe zone. But in the case of quite a lot of cobots nowadays, they can even go behind, which also is not very safe. Even if it is a cobot, it possesses some kind of inertia and can hit you very hard.

So, stop category zero. This is the stop category when the drives are deactivated immediately, and the brakes are applied. The robot performs the path-oriented braking. The trigger upon releasing the drives enabling switch. The drive enabling switch is the switch, which is: if you press that button, the drives are released. Switching off the robot, power failure. So, if the drive-enabling switch is released, what happens? The brake should be applied. As long as it is pressed, it is on. Sometimes, it is a three-state switch. When it is further pressed, it is again locked, which is known as a panic state. You hold the teach pendant. Grip it softly, and it will be on. Again, pressing very hard will switch it off, so in any of those cases, if the drive enabling switch goes off, encoder error mode changes, or pressing any emergency stop button on the teach pendant, the robot, or the safety fence, anywhere you press the emergency stop, stop category zero is executed.

Stop category one is when the robot performs the path-maintaining braking. In this case, the drives are deactivated after one second. That means it continues to be in its path up till one second, and then the brakes are applied. Triggered upon opening the safety gate. As soon as you open the safety gate, it will gradually come to rest. It will take maybe one second, and till then, it will maintain the path. Even after pressing the emergency stop, stop category one may be triggered.

Stop category 2: The drive remains activated. No brakes are applied. Unlike this one (Stop category one), Brakes are applied, and even in this case (Stop category Zero) breaks are applied. In this case, it is continuously under servo. The robot maintains its position or trajectory. The robot stops with a normal braking ramp. If it is a trapezoidal ramp, it will go gradually. It sometimes may be more than a second also, but it will stop ultimately. Triggered upon pressing the stop switch or releasing the start button.



Important Terminologies, Proprietary to KUKA iiwa

Triggers for stop reactions, Stop categories as per ISO 13850: Safety of machinery

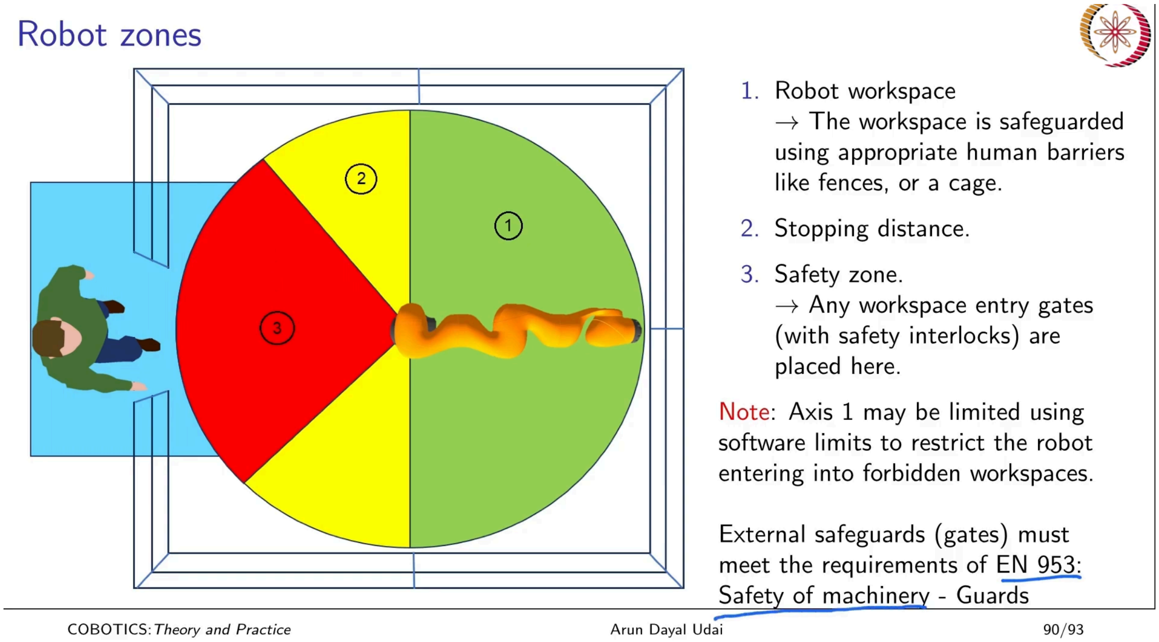
- The safety-oriented functions of the industrial Cobot meet the following safety requirements:
  → **Category 3** and **Performance Level d** in accordance with EN ISO 13849-1
  → **SIL 2** according to EN 62061
  → Such Cobots are fault-tolerant, and can detect and respond to faults within the system.
- **Controlled Robot Retraction (CRR) Mode** (Other than T1, T2 and AUT Modes):
  The robot is stopped by the safety controller for one of the following reasons:
  → Industrial robot violates an axis-specific or Cartesian monitoring space.
  → Orientation of a safety-oriented tool is outside the monitored range.
  → Industrial robot violates a force or torque monitoring function.
  → A position sensor is not mastered or referenced.
  → A joint torque sensor is not referenced.

COBOTICS: *Theory and Practice* Arun Dayal Udai

Now, safety-oriented functions in an industrial cobot meet the following safety requirements also. Normally, they are marked under category 3 performance level. In accordance with ISO 13849-1 or SIL2 according to EN 62061. such cobots are fault-tolerant and can detect and respond to faults within the system, that means it is less likely that they will cause any damage in case of any single fault source.

There is a new term that is only popular with cobots, especially in the case of KUKA iiwa I have seen. Controlled Robot Retraction Mode, apart from the programming mode that is T1 mode or T2 mode, is normal execution. You do it with T2, which is the full speed after programming. You can test it in autonomous mode when humans are not around. So, there is one other mode, which is the CRR mode, Controlled Robot Retraction Mode, the robot is stopped by the safety controller for one of the following reasons. An industrial robot violates an axis-specific or Cartesian monitoring space. Axis- Specific motion could be, let us say, you have limited your joint angle through software. In order to have some safety, you may have some objects lying beyond that angle. If it violates that, then CRR mode will execute robot retraction.

The orientation of a safety-oriented tool is outside the monitored range. Again, due to the orientation or due to the position, if it is outside the Cartesian range, you may also define a Cartesian range in the workspace where your robot is not allowed to go. Apart from fixing up the axial range where your robot should not go, you may also define it in a Cartesian space. So, even if your robot goes into that region due to the orientation or maybe due to the positioning error, CRR mode is executed. Industrial robots violate a force or torque monitoring function. If it violates the maximum force limitation, if you have put it, the robot can retract. As I have shown in one of the examples, I just touched upon the robot while it was lowering, and the robot retracted back. So, that was just a demonstration of that. A position sensor is not mastered or referenced. So, without mastering, if you put the robot to use, it works under CRR mode, and the robot may be retracted. A joint sensor is not referenced. So, all these are mastering errors or referencing errors; Calibration errors would cause this to be executed.



Robot zones

1. Robot workspace
   → The workspace is safeguarded using appropriate human barriers like fences, or a cage.
2. Stopping distance.
3. Safety zone.
   → Any workspace entry gates (with safety interlocks) are placed here.

Note: Axis 1 may be limited using software limits to restrict the robot entering into forbidden workspaces.

External safeguards (gates) must meet the requirements of EN 953: Safety of machinery - Guards

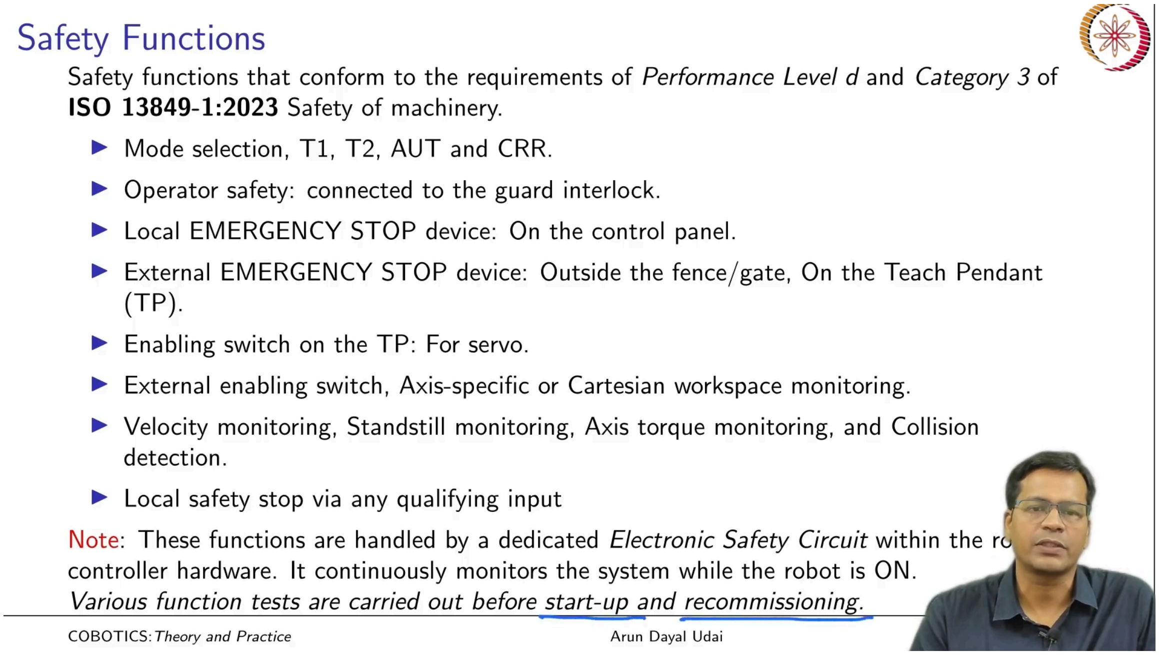
Let us look at the robot zones. We have discussed it earlier also. Let us make it more clear here. So, the robot workspace, workspace number one over here, is safeguarded using appropriate human barriers like fences or a cage. In the case of an industrial robot, that is mandatory. In the case of a Cobot, if it is put in human workspaces with overlapping workspaces, then you need to be very careful in this region.

Number two is the stopping distance that is here. Even if you apply the brake somewhere over here, it can further travel to this region because it takes some time to respond, and after braking, it would take some time to come to a complete stop. So, this is the stopping zone.

Number three is the safety zone. So, this zone is assumed to be totally safe because the physical limitation of the robot joint is exhausted here, and it cannot go further. But there may be some instances in which one of the joints may take it to that space from behind. Although maybe the first joint cannot go there. Using the second, third, fourth, or fifth joint, you can still go to that region. This is still not very safe. You may be a little attentive and still be here. You know the program execution cycle and everything else. You are literally, you are basically quite safe out here.

Any workspace entry gates within the safety interlocks are placed here. So, normally, the gates are placed here. Axis 1 may be limited using software limits to restrict the robot from entering into the forbidden workspaces. So, axis 1 is the first axis of the robot, that is, the vertical axis. If you check this robot, it looks like this. So, this is axis 1. That is axis 1. So, that is the first axis that may be limited. So, in this case, that would cause the robot not to even enter somewhere over here. So, you can always limit this angle.

External safeguards, that is, the gates, must meet the requirements of EN 953, that is safety of machinery, that is, for the Guards. So, there is a special safety of machinery standard EN 953, which decides upon the design of the gates and the type of interlocks that should be there. So, it dictates everything else.



Safety Functions

Safety functions that conform to the requirements of *Performance Level d* and *Category 3* of **ISO 13849-1:2023** Safety of machinery.

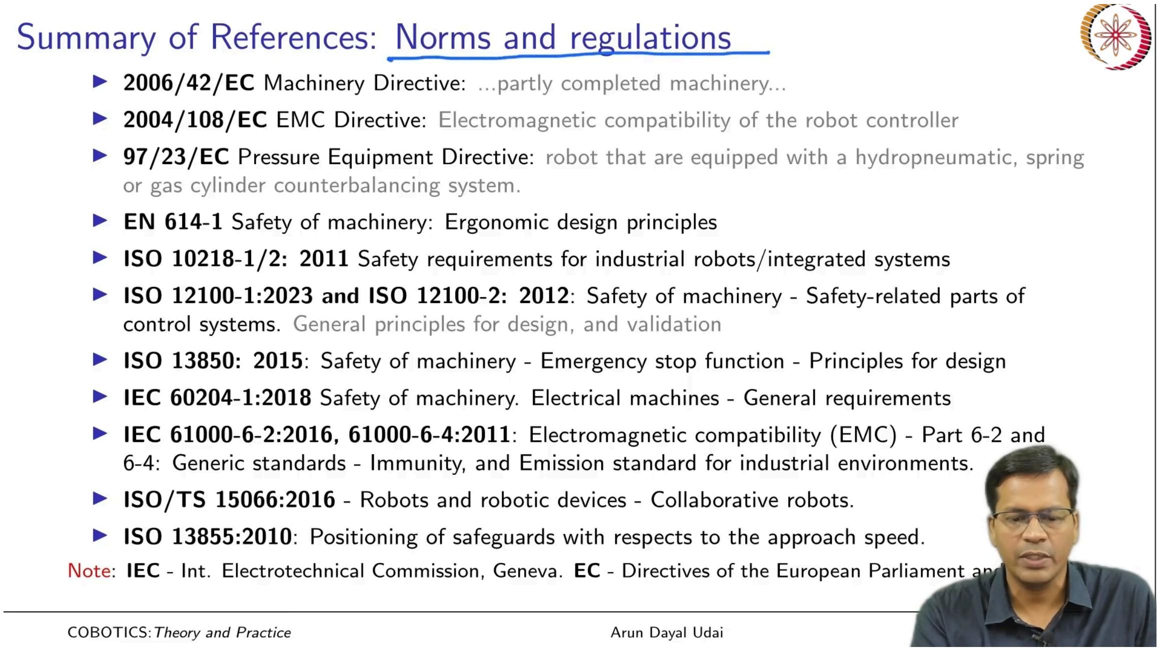
- Mode selection, T1, T2, AUT and CRR.
- Operator safety: connected to the guard interlock.
- Local EMERGENCY STOP device: On the control panel.
- External EMERGENCY STOP device: Outside the fence/gate, On the Teach Pendant (TP).
- Enabling switch on the TP: For servo.
- External enabling switch, Axis-specific or Cartesian workspace monitoring.
- Velocity monitoring, Standstill monitoring, Axis torque monitoring, and Collision detection.
- Local safety stop via any qualifying input

Note: These functions are handled by a dedicated *Electronic Safety Circuit* within the ro controller hardware. It continuously monitors the system while the robot is ON. *Various function tests are carried out before start-up and recommissioning.*

COBOTICS: *Theory and Practice* Arun Dayal Udai

Now, there are various safety functions that I have already discussed quite a lot of time, but I'll just summarise them here once again. So, safety functions conform to the requirements of performance level d, category 3 of ISO 13849-1:2023, that is, for the safety of machinery. From time to time, this year may be revised with some modifications to this that decide the mode selection T1, T2, automatic, or CRR mode that I have discussed. Operator safety is connected to the guards and the interlocks. Local emergency stop device on the control panel, on the teach pendant, also. External emergency stop device outside the fence or the gate on the Teach Pendant (TP) is for the teach pendant. , so the emergency stop on the teach pendant is an emergency stop device that is accounted as an external emergency stop. Enabling the switch on the teach pendant for the servo motor enabling as long as you do not press the enabling switch, the servos are off, and the position or velocity cannot be taken up. Any movement cannot be done. External enabling switch, axis-specific or Cartesian-specific monitoring, governs them all. External enabling switch, axis specific or Cartesian specific monitoring, governs them all.

Velocity monitoring for standstill monitoring Access torque monitoring and collision detection. , so all these safety functions are governed by ISO 13849-1:2023. Local safety stop via any qualifying input. It can be a failure of any attachment which sends feedback and that tends to stop the robot. Also, these functions are handled by a dedicated Electronic Safety Control Circuit within the robot controller hardware. It continuously monitors the system while the robot is on. This hardware is a dedicated one. It is apart from the main controller which executes all the programs. It does forward kinematics, inverse kinematics, and everything else. Various function tests are carried out before startup and after recommissioning of such robots.



Summary of References: Norms and regulations

- **2006/42/EC** Machinery Directive: ...partly completed machinery...
- **2004/108/EC** EMC Directive: Electromagnetic compatibility of the robot controller
- **97/23/EC** Pressure Equipment Directive: robot that are equipped with a hydropneumatic, spring or gas cylinder counterbalancing system.
- **EN 614-1** Safety of machinery: Ergonomic design principles
- **ISO 10218-1/2: 2011** Safety requirements for industrial robots/integrated systems
- **ISO 12100-1:2023 and ISO 12100-2: 2012**: Safety of machinery - Safety-related parts of control systems. General principles for design, and validation
- **ISO 13850: 2015**: Safety of machinery - Emergency stop function - Principles for design
- **IEC 60204-1:2018** Safety of machinery. Electrical machines - General requirements
- **IEC 61000-6-2:2016, 61000-6-4:2011**: Electromagnetic compatibility (EMC) - Part 6-2 and 6-4: Generic standards - Immunity, and Emission standard for industrial environments.
- **ISO/TS 15066:2016** - Robots and robotic devices - Collaborative robots.
- **ISO 13855:2010**: Positioning of safeguards with respects to the approach speed.

Note: **IEC** - Int. Electrotechnical Commission, Geneva. **EC** - Directives of the European Parliament an

COBOTICS: *Theory and Practice* Arun Dayal Udai

So, this is the summary of all the references that I have used. These are some norms and regulations which you may go through.

That is all for this lecture. With this, I would end here. In the next lecture, we will start with a new module, which is Transformation Matrices and Robot Kinematics.