ROBOTIC **MACHINE TENDING**

How to improve **CNC Machine Production**

START PRODUCTION FASTER

LEAN ROBOTICS

Contents

Lean robotics: simplify robotic cell deployment

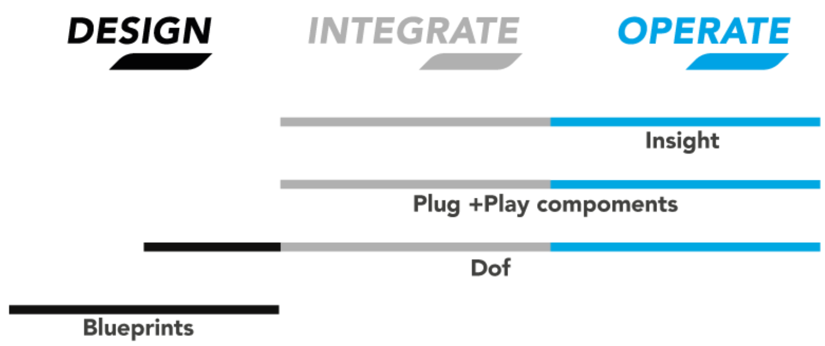
If you ask whether robots could work in your factory, the answer you'll get is probably a hesitant "It depends." It depends on your factory, your team, which robot you choose, what you want it to do... and a whole lot more.

So if you're a first-time robot user, how can you get started? How do you get from your initial idea to a productive, working robot? And if you've already got a few robotic deployments under your belt, how can you scale up your robotics efforts throughout your factory—or across multiple factories?

The answers can be found in **lean robotics: a methodology for simplifying robotic cell deployments**.

Lean robotics is a systematic way to complete the robotic cell deployment cycle, from design to integration and operation. It will empower your team to deploy robots quicker and more efficiently than ever before.

Lean robotics divides robotic cell deployments into three phases: design, integrate, and operate.



Robotiq's library of eBooks covers each phase of robotic cell deployment, giving you access to advice from robotics experts each step of the way.

Learn more about lean robotics at **leanrobotics.org**.

What is machine tending?

Machine tending is the most popular collaborative robot application. Before going too deep into details, let's start with simple definitions.

The basic definition of **tending** is providing treatment for someone or something. In our case, machine tending means to **load and/or unload a given machine with parts or material**. Currently, most machine-tending applications are done by humans. Modern machine shops often use CNC machines (such as lathes and milling machines). These machines must be tended by workers, who place the raw material (usually called the raw or blank part) in the machine and remove it once the machine has done its work. However, since qualified workers are becoming harder to find, companies are introducing robots into their workshops to make up for the lack of employees.

A robotic machine-tending process can be repeated endlessly, assuming the robot continually receives raw parts and the machine produces quality parts. Some industries use robots for a single step of production, like emptying injection-molding machines or CNC machines. When production is running around the clock, robots let you minimize cycle time and run the process continuously by removing parts from the machine's working area.

Read more:

- Getting Started with Collaborative Robots: Machine Tending
- What the Heck is Robotic Machine Tending Anyway?

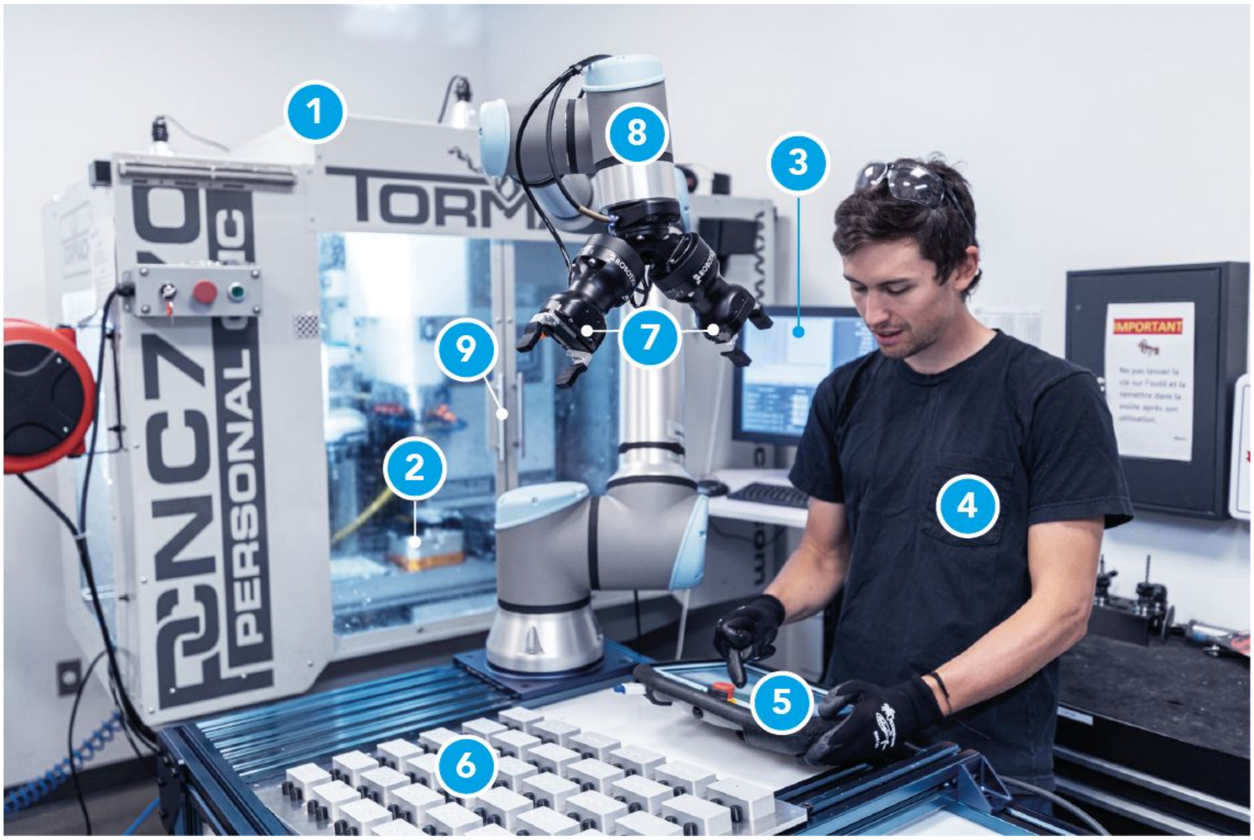
Machine tending with a cobot

Collaborative robots, or **cobots** for short, are power- and force-limited robots. These are robots that can be used without machine guarding or that include other safety features. Cobots can be placed beside a machine or person and set to perform a certain task without needing to be fenced off from the surrounding environment.

That being said, introducing a cobot around a CNC machine and programming it to load and unload parts is not as simple as it sounds. It requires choosing a robot that can accomplish the job and that can reach a level of performance comparable to a human worker.

Anatomy of a robotic machine-tending cell

A CNC cell is pretty complex. However, there are recurrent items that represent the core of a robotic machine tending cell. The anatomy is broken down in the following terminology.



1. **CNC machine**: Autonomously machines parts (CNC stands for computer numerical control).
2. **Workholding**: Holds parts while they are being machined.
3. **Controller**: Coordinates the machine's motions.
4. **Operator**: Operates the machine (in a human-operated cell); operates the robot (in a robotic cell).
5. **Teach pendant**: Handheld device used to program the robot.
6. **Parts**: Material that go into the machine as raw parts—generally one at a time—and come out as machined parts.
7. **Gripper**: Grasps the raw material, places it in the machine, and collects the machined part after the transformation process.
8. **Robot**: Performs the tedious actions of loading and unloading parts for the operator, including opening and closing the CNC door.
9. **CNC door**: Keeps metal debris confined within the CNC machine and prevents part projection while the machine is running.

You may want to look at the following video to get a better idea of the different steps necessary to accomplish machine tending with a Universal Robot and a Robotiq 2-Finger 85 Gripper.

Read more:

- [Anatomy of a Machine Tending Cell]
- [How to do Machine Tending Using Collaborative Robots]
- [Getting Started with Collaborative Robots: Machine Tending]
- [Machine Tending Playbook]

Safety

One reason why it's easy to use collaborative robots in machine-tending applications is their built-in safety features. Not only will the robot automatically stop when it comes in contact with something (or someone), but it can also be limited to moving in a certain space. It only takes a few minutes to configure a safety plan that will restrain the robot's arm movements to a defined workspace. This reduces the danger of using a robot and also reduces the need to buy fencing or other safety devices to prevent collisions with workers.

If people haven't seen a collaborative robot at work before, they might be nervous about the idea of being so close to one. Demonstrating safety in action is a great way to relieve anxiety. Use your first project to showcase cobots' safety features. Word will spread about how smoothly the robot moves and how easily it stops, and people will relax.

Who determines cobot safety guidelines?

The most widely recognized source of safety guidance for cobots is the ISO/TS 15066. This is a technical report produced by the International Organization for Standardization (ISO). ISO/TS 15066 (Robots and robotic devices — Collaborative robots) is not actually a standard yet. It is a technical specification (TS) because more technical development is required before it can be turned into a full standard.

For our purposes, this means that the safety limits it prescribes are likely to change. As we better understand the practicalities of working alongside collaborative robots, the limits will be updated. However, even though it's not a full standard yet, we'll still refer to this report as a "standard," because it's the closest thing we have to a standard for cobot safety. You can find out all about the standard in our eBook, *ISO/TS 15066 Explained*.

Read more:

- Should we fence the arms of Universal Robots?
- Collaborative Robots Risk Assessment, an Introduction
- 5 Cobot Applications Where Safety is Critical
- How Safety Levels Are Decided for Cobots
- Cobot Safety: Are You Too Close to Your Robot?

Business examples

Companies are using cobots (collaborative robots) to accomplish three things: expand production, redirect human workers to higher-value tasks, and ensure everyone's safety. In this section, we'll look at several examples of how they're making it happen.

Trelleborg: Expanding production

Video: Watch now

At Trelleborg Sealing Solutions, a Danish manufacturing company, collaborative robots are being used as smaller, safer alternatives to traditional industrial robots. Each cobot has effectively become an accessory to each CNC machine. As Jasper Riis, the company's head of production, says, “Whenever we buy a new machine, we also order a robot for it.” This is a completely valid approach, and it is clearly very effective for Trelleborg's purposes.

Machine operators in the company are able to program the robots at a basic level, but that seems to be all the robotics training they have at this stage. The operators' jobs have remained pretty similar to the machine tending they did previously. The main difference is one of scale. Before, each worker would tend three machines simultaneously; now, they tend eight.

Read more:

- How Robotics Training Helps Create New Applications
- Why You Need Robotics Expertise

Lowercase NYC: The 5-day Integration Project

To produce its stylish eyewear, Lowercase NYC imports raw material in sheets. The sheets are cut down into small tablets and then turned into glasses frames in a CNC machine. The machine is tended by a UR5 cobot from Universal Robots and a Robotiq Gripper. The robot picks up the raw material, loads it into the first position, closes the door, presses the start button, and then does it all again for the second position. After machining, the robot removes the finished pair of glasses and places it in a bucket. The eyewear then goes through many steps of fine-tuning before being ready to ship.

Each production batch in the CNC machine averages 500 units, with around 30 to 40 units of each style. But even with such a small volume, Lowercase NYC co-founder Brian Vallario says the automated process is by far the best solution. "I only have a few manual tweaks to do on the vices every time. The rest is all automated," he explains. "This automation is important for us. Eyewear production is a very labor-intensive process, and we are such a small team that any improvement we can make in our efficiency is huge. Having a product that allows me to sit at the computer and work on design or go work on the more labor-intensive stuff that can't be done by machines is a big plus for us."

Read more:

- Case study: Lowercase NYC's 5-day Integration Project

Tegra Medical: Redirecting human workers

Some people might read the stories above and think, "Sounds like the robot took that machinist's place and now that guy is out of a job!" But that's not true. The point of collaborative robots is to do more manual tasks and free up human workers for more valuable tasks.

Instead of providing physical labor, workers can spend more time on quality control, part verification, process optimization, and making sure production is running smoothly. Tasks like equipment maintenance can be done more often, and all without having to worry about whether the machine is being fed. Cobots enable employees to:

- Focus on process optimization.
- Focus on product quality and verification.
- Be redirected to more important or pressing tasks.
- Work in a more stimulating environment.

Instead of standing in front of the machine waiting for the next part, workers can keep learning new things. Ultimately, employees get more stimulating jobs and managers get better production. It's a win-win.

Tegra Medical started small when it introduced collaborative robots, beginning with a single UR5 robot before scaling up to three UR5s on different machines. A culture of training and innovation is a key part of Tegra's approach. As Hal Blenkhorn, director of manufacturing, says: “No one is going to lose their job to a robot. We're trying to [add more value to] employees, to train them in new skills, whether it's a different operation or making them the robot supervisor in that area.”

The culture of in-house expertise let Tegra get creative with its use of robots. Taking what they learned from their first three integrations, the team members embarked on an ambitious application for their next robot, a UR10. As Blenkhorn explains, “The challenges in this last cell were running three different products on three different operations. It's unusual for us—and it's unusual in the industry—to have a mixed-model cell like that’s feeding three [different] products simultaneously.”

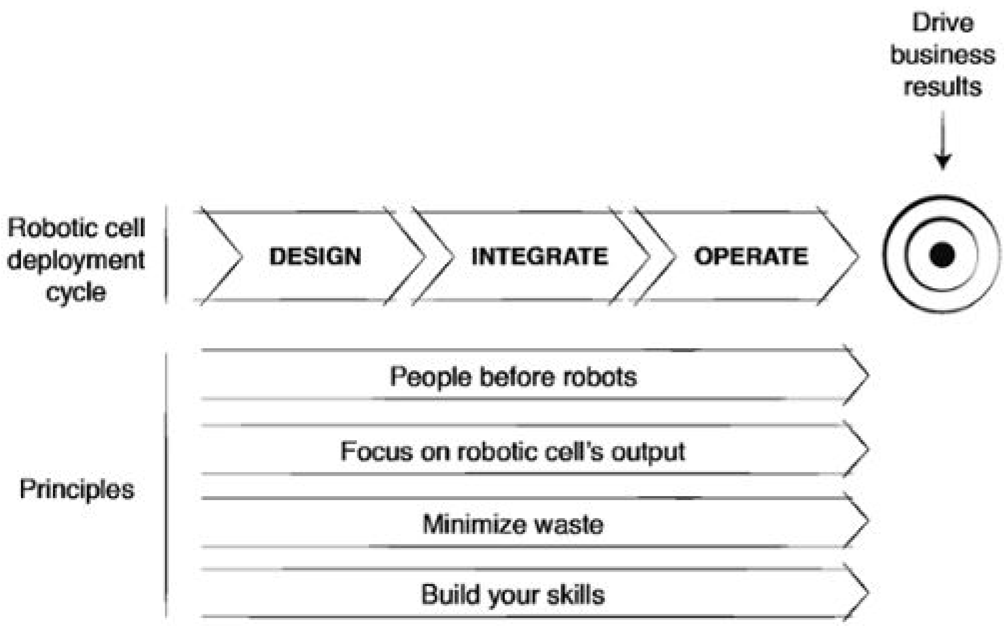
By taking over basic programming functions, the manufacturing team has given Blenkhorn more time to develop new applications. The operators are even able to get involved in developing these new applications.

This is clear when Senior Manufacturing Engineer Paul Quitzau talks about Tegra’s next application: “We are very excited. We’ve purchased our fourth robot and it's going to go into another area of our company where we perform laser marking.” As for how easy it was to install, Quitzau says, “I was able to piece together the entire application without much issue, and I'm not a programmer, I'm a mechanical person.”

Read more:

- 3 Ways to Keep Your Robotic Cell Project Simple
- How Robotics Training Helps Create New Applications

DESIGN a robotic machine-tending Cell



The lean robotics methodology at a glance.

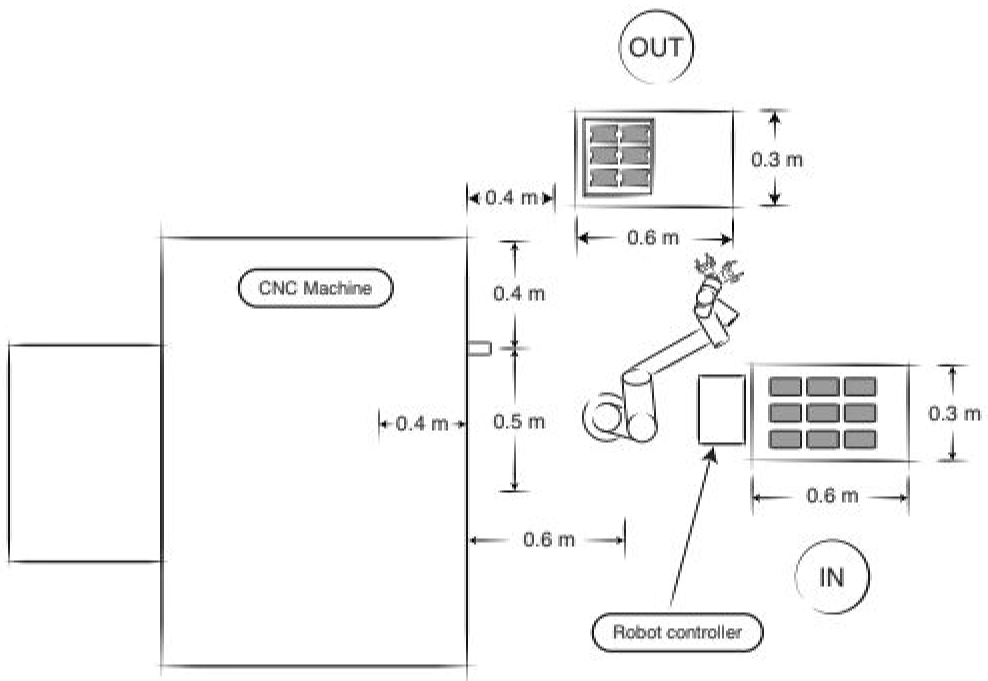
With your first robotic cell deployment, your primary objective should be getting a positive response to the project along with bottom-line return on investment numbers. Your first project is an opportunity to dispel myths and highlight benefits regarding safety, job security, programming ease, and other concerns. By implementing a robotic cell that's safe, frees people up for other value-added work, and does what it's supposed to do without requiring excessive programming expertise, you'll gain enthusiastic support for future projects.

How do you set the first project up for success? Keep it simple. Our book *Lean Robotics: A Guide to Making Robots Work in Your Factory* advises, "If you need to choose between a simple, low-ROI application and a complex, high-ROI one, it's best to go with the simple one... Far better to start simple, and make sure the first cell deployment is a success, so you can start creating value with it while building momentum for more ambitious future projects."

For your first project:

1. Implement a simple, proven solution.
2. Get it up and running as quickly as possible with no unnecessary extras.

Set a short timeline so you can quickly get back on track if the project fails.



Example of a robotic cell concept layout.

Read more:

- [3 Ways to Keep Your Robotic Cell Project Simple]()
- [The *Lean Robotics* Book: A Guide To Making Robots Work In Your Factory]()

Manual process definition

The design phase of a machine-tending application starts by looking at your actual process. Monitor your human-operated cell for a few days or weeks to get an idea of whether your investment in robotics will be worthwhile.

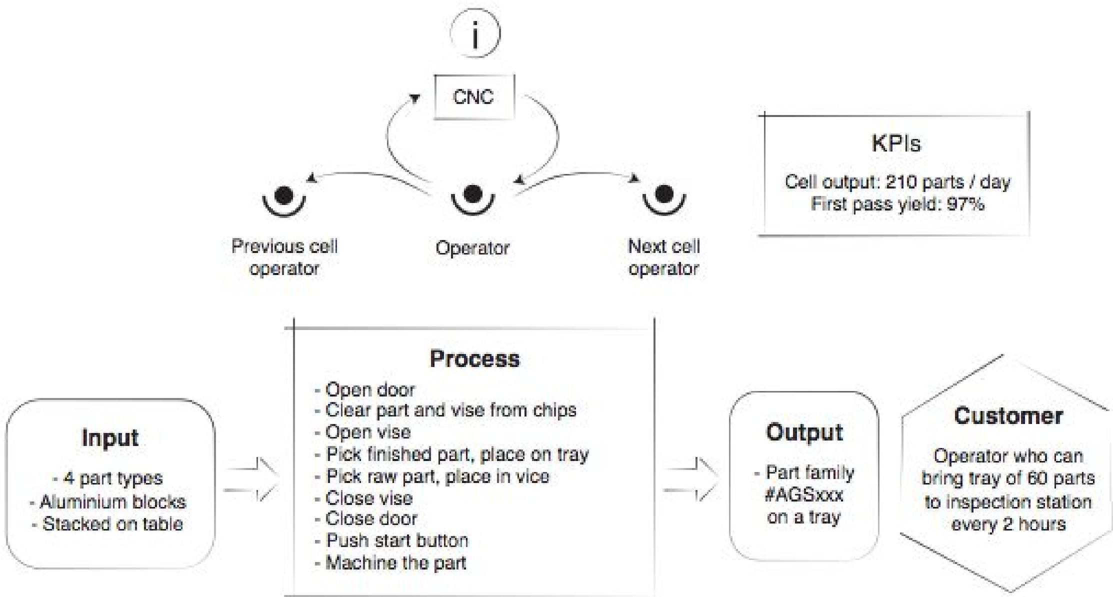
To track the current cell's production, record the following data:

- **Production rate**
- **Spindle time**
- **Part defects**

Next, record photos and videos of your current machine-tending process. Ask your operator to narrate their task out loud, and pay attention to hidden tasks (steps that may seem trivial, for example) to see exactly what's being done. We need to capture the information necessary to develop a concept of how a robot could do these tasks.

Ask:

- Where do the parts go when the station is done with them?
- What are the qualities of a "good input" for the next station?
- What is coming into this station?
- How are the parts processed?
- What information is used at the station? Where does it come from, and where does it go?
- What are the KPIs and how are they measured?
- What is the current spatial arrangement of the station?



Value-added time (s)	75
Non value-added time (s)	18
TOTAL (s)	93

Example of a manual task map summary.

In lean robotics, we start the robotic cell design by creating a manual task map: a static snapshot of the starting point from which we will work to improve. Capturing this information will save you time in the later phases.

The manual task mapping template will help you gather all the relevant info about your manual process. We've also included an example manual task map so you can validate your work.

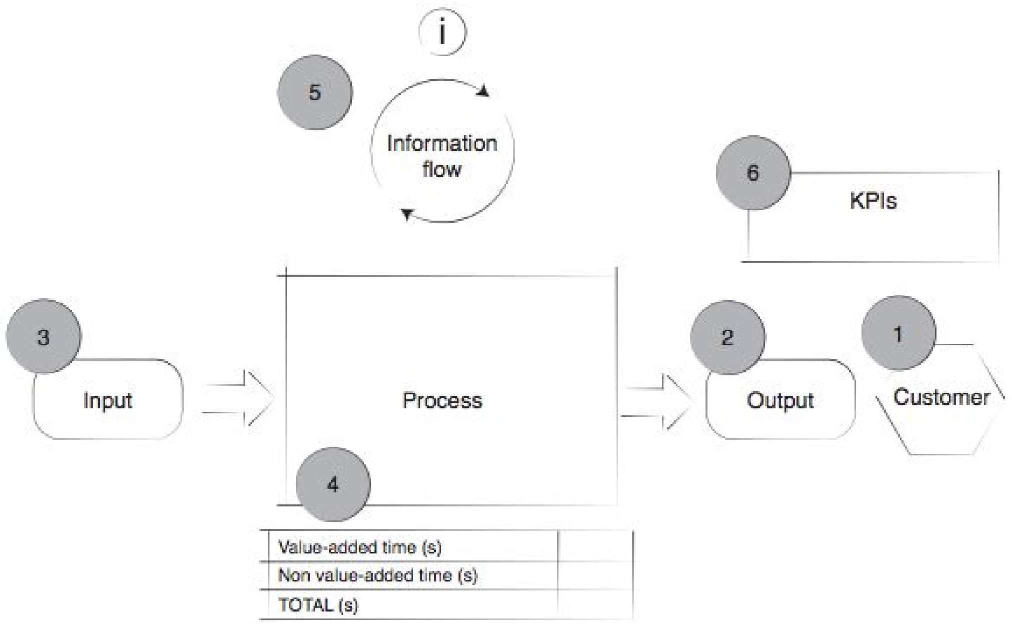
Read more:

- Machine Tending Playbook
- Lean Robotics
- Manual Task Mapping Resources

Now that you have clearly defined the problem by taking a static snapshot of the current process, you can start looking for a technical solution.

Robotic process definition

Once your manual task map is done, you are ready to figure out its robotic equivalent: the robotic task map. The thing about robotic task mapping is you can't complete it without knowing the layout—but you also can't complete the layout without knowing the task map! That's why we suggest working on these two components together.



Robotic task map template. Numbers in circles indicate the order in which the information is to be filled.

The robotic task mapping template follows the same structure as the manual map. Complete this step before moving on to the manual-robotic comparison. We've included an example of a robotic task map so you can validate your work.

Read more:

- [Robotic Task Mapping Resources]

Before doing the robotic task map, you first need a high-level robotic cell concept that describes the cell constituents and task sequence. You start your high-level robotic cell concept work by identifying the main parts of the robotic cell that are not present in the manual cell. These components include the robot, tooling, sensors, safety measures, and software.

You will have to evaluate the specifications of every component that will be added to the robot, including the tooling, sensors, safeguards, and software. You will also have to consider how these things will interface with your chosen robot.

The next step is describing the high-level sequence of steps that the robot will perform. The sequence description should include the part infeed technique, the part outfeed technique, the process sequence, and the information flow.

The robot

Which brand and model of robot has the right specifications for the process—considering reach, payload, speed, repeatability, compatibility with tools, etc.? The cobot eBook is designed to help you determine which robot best suits your needs.

(By the way, a new version is about to be released; sign up to our newsletter to be the first to know when.)

Read more:

- Getting Started With Collaborative Robots—Cheat Sheet
- How to Do Process Analysis for Machine Tending
- Getting Started with Collaborative Robots: Machine Tending
- Machine Tending Playbook

Reach

The robot's reach, or **work envelope**, is the robot's range of movement. If you want your robot to grasp parts, use a tool, and open a door, all these items must be located within the robot's work envelope.

A machine-tending robot needs to take a blank part from somewhere (point A) and place it in the machine (point B). To calculate the minimum reach your robot needs, measure the distance between these two points and divide it by two. Keep in mind that the result is a rough approximation of the minimum reach with the robot mounted in the middle (which may or may not be possible for your cell design).

Remember:

- A good rule of thumb the larger the reach, the higher the payload. Larger robots tend to be stronger (because they need to lift heavier payloads). A robot's reach is determined by its number of axes (degrees of freedom) as well as by the length of the segments in between the axis.

Payload

The robot's payload is one of the most important specifications. The payload is the total weight the robot can carry. Since your robot will carry different types of parts, you will need to determine which parts will be the heaviest and select your robot accordingly. To determine your required payload, sum the weights of your tool and the heaviest raw part you want the robot to carry.

Calculate part weight by multiplying the part's volume (W x L x H) by its density (g/mm^3). Or use CAD software to determine the weight of the part; that way you might as well weigh your part and arrive at the same results.

To avoid possible robot stops, it's important not to weigh it down with a load that's too close to the maximum payload. Plan to carry less than 90–95% of the payload.

Read more:

- DoF community discussion: Increasing speed and acceleration on a UR robot

Repeatability

Although CNC machines are super repeatable and you want your parts to be precise, you don't need your robot to be as precise as your machine. Machine-tending robots are typically used to do first setups where the part is usually larger than the finished part, which allows for small variations in positioning.

Collaborative robots are usually repeatable within 0.1 mm. If your process requires more repeatability, you can add a mechanical stopper on your vise and use the robot's force feedback to ensure precise placement in your fixture.

> ***Quick tip:*** *In the event of a displacement of the part in the vise jaws, the robot will force against the vise. Since the two components are forcing against each other there might be premature wear on the robot joint, or misplacement. That being said, by letting one of the robot joints "go free", the robot will adapt to the force of the vise and let the vise do the work. In Force Copilot, the Force Control node enables such freeing/release of force presence.*

Tooling

Robot end effectors (grippers)

Your gripper is the robot's "hand," but it's not nearly as versatile as your hand. Grippers work best when parts have at least two parallel surfaces. The stroke of the gripper will limit the range of parts you can handle.

Try to choose an adaptive gripper that can handle different shapes and, more important, different sizes of parts without modifying fingers or the robot's programming.

> ***Quick tip:*** *Make sure your gripper can handle 85–95% of the parts. The rest can be loaded manually if need be. To handle heavier payloads, consider a tool changer or dual-gripper setup.*

Gripper payload

Like robot payload, gripper payload is the amount of weight the gripper can handle. Make sure to respect this payload to ensure gripper longevity.

> ***Quick tip:*** *In addition to respecting payload, be careful not to max out the gripper's allowable torque. If the bulk of the object's weight is always grasped by the gripper's fingertips, the gripper will wear prematurely.*

Gripper dexterity

Grippers often have very limited dexterity. In fact, if your application requires the operator to handle several different parts at the same time and perform operations with both hands, proceed carefully. Robots can only do one thing at a time. This doesn't mean you can't use robots, but it does mean you will most likely have to redesign your process.

Here are some of the most important robot dexterity factors, along with questions to ask when defining your needs:

- **Object size.** How big are the parts? Are they identical, or a variety of sizes? How does this compare with the reach required of the robot?
- **Object shape.** What shape are the parts? Do they have many complex edges or a simple geometrical shape? Are they spherical or otherwise difficult to grasp?
- **Grip strategy.** How could the parts be grasped (with an encompassing grip, internal grip, or suction)? Are there different ways to grasp the same parts? Are the parts delicate enough to require a particular gripping strategy?
- **Reach.** How much does the robot have to "stretch" to reach all important locations in the workspace? Will the robot use its entire workspace or just a small part of it? Does it need to approach locations from many different angles?
- **Speed.** What cycle time is required for each action?

People sometimes think the first three factors above (object size, shape, and grip strategy) only relate to the robot's gripper, and the latter two (reach and speed) only relate to the manipulator, but they're actually all interrelated! One factor in isolation does not necessarily make a robot "dexterous." You can only get a full picture of dexterity when you consider all the factors together.

Read more:

- How Much Dexterity Does a Robot Need?

Gripper stroke

Your choice of gripper will depend on the parts to be handled. If your parts vary in size, you will need a flexible gripper. Robotiq's 2-Finger 85 Gripper can accommodate parts between 5 mm and 80 mm without any additional programming. If you will always be handling parts of the same size, you may want to go with a more rigid custom gripper that will perfectly fit your part.

Quick tip: *A flexible gripper can grasp both raw and finished parts. A 2F85 Gripper will also allow you to adapt the force of the gripper in cases where the finished product is more fragile than the raw part.*

Dual-gripper setup

The most common reason why people will use a dual-gripper configuration is to improve cycle time. This is possible because a dual gripper lets you pick up multiple objects at once. When choosing the gripper for your application, you must consider:

- Process (cycle time, required repeatability, environment, motion, etc.)
- Part (weight, material, shape, size, etc.)
- Price

Because CNC machining environments generally have harsh industrial conditions, such as chips, cutting oil, and coolant, we recommend using the Robotiq Hand-E Gripper because it has an IP rating of 67 and is easy to customize with custom fingertips.

Read more:

- 2F-85 and 2F-140 Grippers
- Hand-E Adaptive Gripper
- Robotiq eLearning course on Adaptive Grippers: How to install and set up dual grippers
- 6 Things That Are Possible With Dual Grippers (+ 3 That Aren't)
- Automate 2017: Machine Tending with a Robotiq Dual Gripper

Fingertips

Robotiq Gripper with fingertips designed to grasp cylinders in two different axes (as well as flat/square geometries).

In machine-tending operations, customers often want to use the same gripper for several operations. Obviously, these applications involve placing raw material in the CNC machine with a regular (round or square) shape and picking it up again when it comes out in a totally different shape. That's why some fingertips are designed to grasp different types of parts.

We've seen many machine-tending applications using custom fingertips to better match the production process. As mentioned, the raw material is more often than not delivered in square or round shapes. But whereas a square is quite intuitive to grasp, round shapes are more complicated. To increase your contact points, V-shaped custom fingertips can help. They let the robot get four points of contact on the cylinder. This results in grasps that are way more stable than with just two points of contact. The cylinder will also be centered, meaning that the center of the cylinder will always be in contact at the same four positions on the gripper. This increases positioning accuracy and makes the process much easier to program.

Read more:

- How to Design Custom Fingertips for Adaptive Grippers | Solidworks Tutorial
- How to Design Gripper Fingertips
- How to Design Custom Fingertips for Your Application

Prehension

Prehension is probably the most complicated aspect of the machine-tending process. In fact, prehension includes the choice of an end-effector and robot. You need to know what kind of parts can possibly be handled in the cell.

You also need to ask yourself if the cell will be dedicated to a single product or if the product will change every month (for example). This will affect the amount of grip force and payload needed, and determine whether you will need a flexible or rigid cell.

If your production will stay steady for years, feel free to go with a simple two-position gripper. If, on the other hand, your production changes relatively often, you will want to use a flexible robot gripper instead. Grippers like our 2-Finger Adaptive Robot Gripper and 3-Finger Adaptive Robot Gripper are ideal for flexible applications.

Automated vise

If your robot will be loading and unloading parts on an unattended machine, you should use a vise or chuck that will open and close automatically. If you are automating a lathe machine, you're already covered: most turning machines have hydraulic chucks that can be controlled either by the machine or the robot itself.

With conventional milling machines, very few of them have automated vises. Most have manual vises, which will need to be replaced or automated in order to open and close when the robot is tending the machine.

Control of the vises or chuck can be done through the robot or machine M codes, depending on your configuration.

Cleaning

Cleaning the working environment is an important part of the process. In most cases, the first thing a human operator does when opening the CNC door is blow air over the part to remove metal chips and coolant.

The same applies to the robotic process. The robot has to blow air over the part not only to clean the part itself, but also to ensure it gets a good grasp on the machined part. Once the part is removed from the vise/chuck, common practice is to blow air on the jaws to remove any remaining chips or coolant. This will allow the next part to be inserted straight into the vise jaw.

Video: Machine-tending cell that uses pneumatic cleaning to clear debris.

Sensors

Do you plan to do closed-loop control or logic-based programming with sensor data? Sensors can be simple—like limit switches—or more complex, like vision systems and force-torque sensors. Sensors are your robot's eyes and ears. They'll give you a lot of information, and you must manage all the data correctly if you want the robot to interact with other devices in your cell.

Proximity switches and other physical sensors are commonly found in robotic machine-tending cells. If you can't manage them with the remaining I/Os in your robot controller, you will need to use a PLC.

Also, even though collaborative robots have many safety features, they can benefit from additional safety sensors. You will most likely need a safety-specific PLC to manage all your different safety sensors.

Read more:

- Find the Ideal Force Torque Sensor for Your Application

Vision

Adding vision capabilities to your cobot opens up a new world of possibilities, from part inspection and bin picking to counting and sorting by color. With so many applications out there, it's critical to evaluate your project up-front to ensure you choose the right technology for the job.

Four key elements of your vision setup are lighting, image capture, image analysis, and image application.

Read more:

- Vision Systems for Collaborative Robots
- Wrist Camera

Measurement and sensing

While robots are great at following instructions and doing repetitive tasks, they're usually not so good at adapting themselves to changes in the environment. But by using various sensors, it's possible to lend "senses" to robots that allow them to better interact with their environment.

One such sense is **the sense of touch**. By using a force/torque sensor, the robot can "feel" the environment and act on it. Whereas some robots use torque sensors directly in their joint to sense force at the effector, others use a 6-axis force-torque sensor directly at the effector. Both methods have advantages, but the force-torque sensor only enables readings at the tool and not through the whole robot.

Equipped with a sense of touch, the robot can now perform actions such as finishing or assembly. It also allows the robot to find objects in the environment by touching them.

Video: Robotiq Force Copilot

Video: Fusion OEM Case Study: Automate Machine-Tending Applications with Hand-E Adaptive Gripper and Force Copilot

Video: Find Machine-Tending Surface

Video: Program Linear Search in Minutes

Video: Program Insertions in Minutes

Read more:

- eBook: Robot Force Torque Sensors: An Introduction
- Find the Ideal Force Torque Sensor for Your Application
- FT 300 Force Torque Sensor

PLCs

A programmable logic controller, or PLC for short, is typically used in industrial robot applications. As a cobot user, you most likely won't need a PLC at all.

Cobot controllers usually have enough configurable digital I/Os to manage an entire CNC machine-tending cell. In fact, we often see robots sending and receiving signals from two or three different CNC machines, while still having room for other I/Os.

That being said, when selecting a robot you should still check that the number of I/Os available on the robot matches the number of sensors you want to use.

Safety measures

Collaborative robots are designed to be inherently safe, so most of the tooling that is sold for them is also safe. However, that doesn't necessarily mean the entire application is safe.

For example, you can carry sharp knives or styrofoam pellets with the same robotic tooling, but the risk will be very different. Even if all the cell components come with top-notch safety certifications, the application itself can still be dangerous.

That's why it's important to do a risk assessment before installing the robot on the shop floor.

Bear in mind that the risk assessment needs to cover every aspect of the cell, not just what is on the robot arm. For example, a pneumatic part feeder can be considered "dangerous" and would need to be protected by a casing.

Check out our resources for more details on collaborative robot safety, how to understand the relevant standards, and how to manage the risk of having a robot on the shop floor.

Read more:

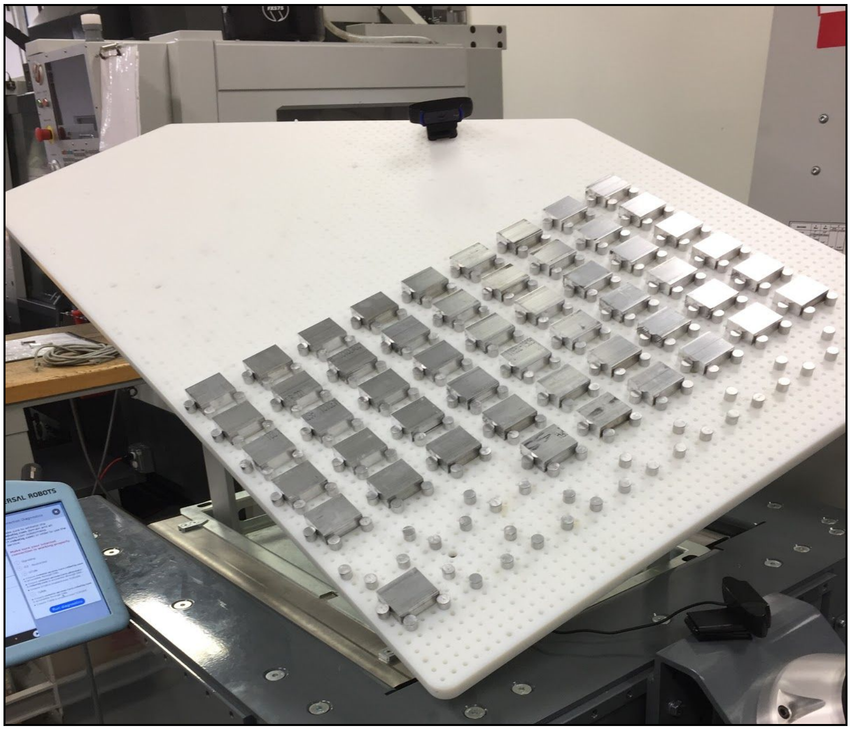
- ISO/TS 15066 Explained
- What is ISO/TS 15066?
- Collaborative Robot Risk Assessment
- How to Perform a Risk Assessment for Collaborative Robots

- [Collaboration Robot Risk Assessment: An Introduction]

Part infeed and outfeed techniques

In the design phase, you need to make a few preliminary decisions about the robot's infeed and outfeed. Will the robot cell accept parts presented the same way as in the manual cell, or do you need to build specific part-presentation devices?

The same question goes for the outfeed. Can the robotic cell output parts the same way as the current manual process, or are modifications required?



Outfeed

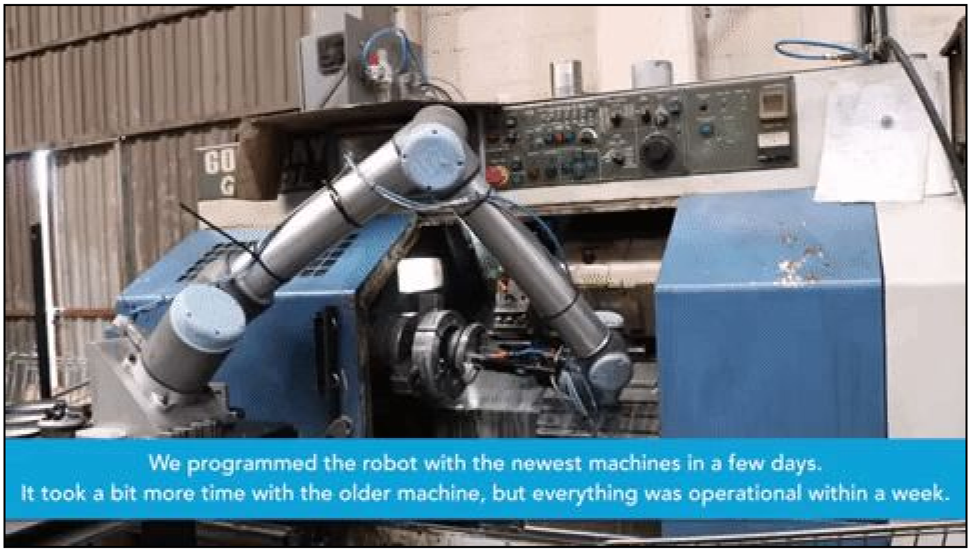
There are many ways to output parts. The best one for you depends on your application. For instance, aerospace machining will require a more structured part outfeed than general machining.

When designing your cell, keep in mind that part output techniques that seem simple for humans, like nestling delicate parts inside their packaging, might be complicated for robots. You don't want to add another operation after the robot has unloaded a finished part from the machine. If the part is ready to go, place it in its final destination. A tray or box is ideal.

Some of the best machine-tending cells we've seen have used trays. If you program the robot to leave a specific amount of space between the parts, you can ensure there's always enough space for the robot's fingers to open once the part is placed on the tray, and prevent the parts from being damaged during transport.

Example: ordered outfeed for part tracking.

Example: structured outfeed.



Example: unstructured outfeed.

Infeed

When designing the infeed, you need to consider how parts will be transported to the cell and how the robot will pick them for reliable insertion in the machine. As with the outfeed, you must choose between structured and unstructured part presentation.

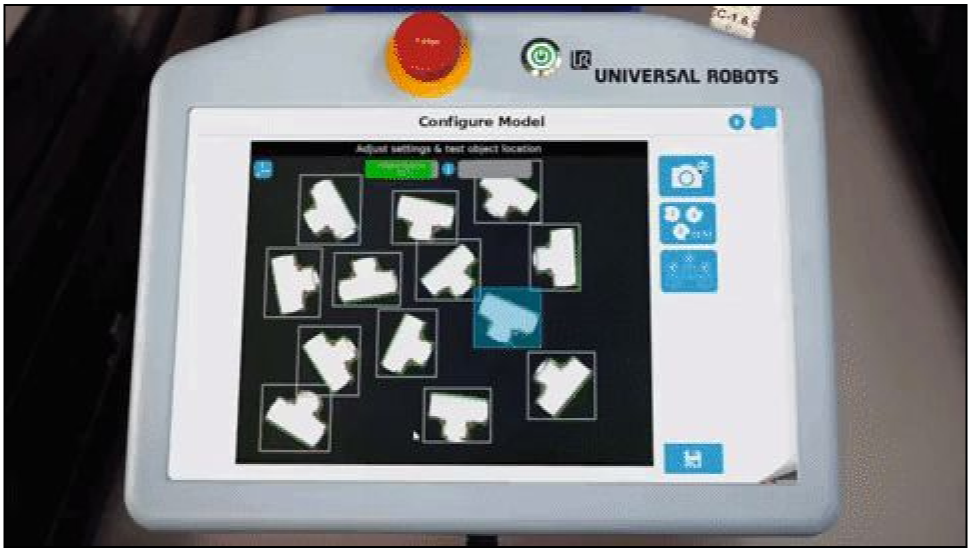
At human-operated machine-tending stations, we often see raw parts arrive all thrown together in a bin (unstructured). This works fine for humans, but it's more difficult for robots. Grasping is easier when there's some space around each part, so make it easy for your robot by placing raw parts on simple trays.

What we usually see is a matrix of parts (structured). This type of part placement is easy for human operators to prepare, and is fairly easy to program for the robot. If you change the type of part being machined, you can simply change a couple of points and you're good to go.

Another structured method is a classified racking system. By classifying each part and programming its coordinates, it's relatively simple for the robot to know exactly where each part should be. However, you might have to put some time and effort into the racking design. This is worthwhile if the raw parts will remain the same for a long time, but if the production needs to be changed and a new raw part needs to be classified often, the system will quickly become frustrating.

Example: structured infeed.

With unstructured part infeeds, you need [vision](). Raw parts can be placed in a conveyor, bin, or drawer, depending on their size. Since the parts are not positioned precisely, the robot program must be paired with a vision system. The system takes photos to find exactly where to pick up the part. If the robot needs to grasp parts while they're moving (such as on a conveyor belt), you can set up a dynamic vision system by pairing the camera with an encoder on belt.



Example: unstructured infeed.

Video: Incoming raw parts ordered on a conveyor belt.

The bottom line: vision systems work best in simple contexts like picking up a part from a flat surface. They're more trouble than it's worth for complex visual contexts with lots of colors and reflections (like the inside of a CNC machine), which make it hard for the vision system to recognize the part.

That's why we recommend starting with a simple device that will position the part in the same spot every time. This is relatively easy to program and highly repeatable, especially since most cobots have built-in wizards to simplify path programming.

Process sequence

Design for manufacturing

It's safe to assume that most of your products are for customers with very specific requirements. You probably can't change a lot of features in the final product. However, you can still adapt your setup to the robot. If your gripper works best for grasping parts of a certain width or shape, you may want to leave a feature of this width on the part at the first setup, or simply ensure it has two parallel surfaces to allow for a better grasp. Doing so will help the robot load the part for a second machine setup without any problems.

Video: Watch How Scott Fetzer Electrical Group customized parts for automated machine tending.

Testing the new process

If you have new infeed and outfeed processes, you'll want to test them during off-the-line setup.

The act of picking up or dropping off the part from the ordering device is critical. Make sure you have a consistent process that will raise your robot repeatability and keep track of your parts. Better to spend a couple more hours testing and bring your an infeed/outfeed process to a higher level than lose time while the robot is running.

Information flow

Communicating with the CNC

Many signals can be sent between the robot and the CNC, but the PLC can only process a limited number of I/Os. For that reason, it's a good idea to limit the amount of data you would like to access. Prioritize the signals that are necessary for efficient machine tending.

As explained in a discussion on DoF, it's easy to use a simple signal from the CNC machine to start the robot program—such as a light signal—but it's much harder to extract enough precise data from the CNC to trigger your robot. Keep communications between the robot and CNC machine as simple as possible. This will simplify your integration and prevent you from having to deal with complex logic chains.

> ***Quick tip:*** *To trigger the start or end of a program, use discrete values like end position, a proximity sensor signal, or a light signal, not variables like distance, position, current, or force. Enforce strict barriers between "go" and "no-go" by leaving no room for interpretation.*

Read more:

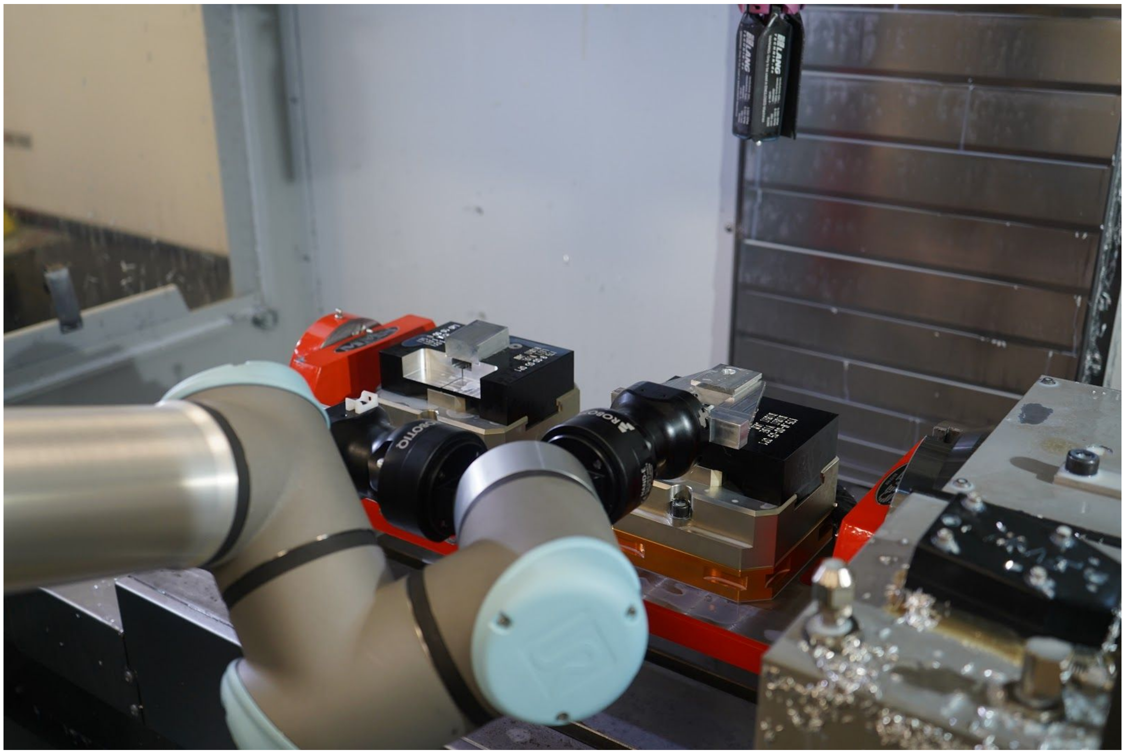
- How to Do Process Analysis for Machine Tending
- Machine Tending Playbook

How to communicate with your CNC machine

The challenge is not a robot, it's the application!

In machine-tending applications, the challenge is not to make the robot work, but to make everything around it work correctly. Integrating a robot effectively requires some sort of handshake between the machine and the robot.

Given that there are hundreds of machine brands, many of which offer several different models, it would be hard for anyone to put together a perfect robot integration on their first try. Starting small is the first step.



Some brands have published guides to the best way to integrate a robot in front of their machines. One of the most detailed documents comes from Haas Machinery.

Their guide, called Robot Integration Aid, lets us dive deep into the process of integrating robots for machine tending. With its detailed diagram and robot sequence, it gives you a better idea of how complex (or simple) a robot integration can be.

Here are the main steps for a clean robot-machine integration:

1. Contact your machine service provider and ask them about your options.
2. Buy the expansion package (your machine company might have a different name for this).
3. Connect everything together.
4. Program a sub-program to send signals to the machine. (You can re-use this subprogram in a bunch of other programs.)
5. Program the robot to perform the necessary actions, like opening and closing the machine door and vise.

Stick to one step at a time in order so you can debug each step separately. If you try to do everything at once, you'll have a much harder time figuring out which segment is responsible for which problem.

For a more detailed example, check out one of our customer's explanations of how to integrate a robot on a Haas VM3 milling machine.

Does this sound too complicated? Some integrators will offer integration services only for the robot/machine handshake.

Open your CNC door using a robot

One of the most important steps of automating your machine tending is getting your robot to open the CNC door.

Before going too deep into the solution, what you need to know is that one of the main KPIs of CNC machine-tending cells is the machine's output or productivity. In other words, you want your robot to get into the machine, change the part, and get out in as little time as possible

You have two main choices: use the robot to open the door, or have something else to open it.

Robot moving the door

The easiest (read: cheapest) way to automate your door-opening is to have the robot do it. The robot can basically grab the door handle and slide the door open until it has enough room to get in.

Once the machine is ready to start, the robot just needs to close the door and everything can run normally.

Video: Watch how Whippany Actuation Systems faced increased demands for specialized parts.

The main advantages of this solution are simplicity and cost-effectiveness. Think of it like this: if you own a robot, you also own a CNC machine door-opener. There's no need for a robot-machine handshake, and you can be up and running in no time.

The downside is that the process is usually slow, and it takes more time for the machine to get back at it. If you're not very concerned about cycle time, this can be a good option. However, this method may not be the best for optimizing your production.

Built-in door opener

You're not the first to want to automate a CNC machine. It's such a popular application that CNC machine builders are now offering automatic doors as add-on features. If you're buying a new machine, you can probably get an embedded door opener for few thousand dollars extra.

One major benefit is that you can place G or M codes in the program to make the door open at specific times. No handshake is needed between the door opener and the robot.

This convenience comes at a cost, though: it's relatively expensive, and if you ever want to operate your machine without a robot, you'll have to press a button to open the door every time.

Off-the-shelf door openers

Many automation companies are starting to offer automated door openers for CNC machine-tending applications. Brands like EasyRobotics and CNC AutoDoor specialize in these devices. They're generally easy to install and need minimal customization to get started.

These doors are built with the robot in mind, so they're easy to integrate with the robot. You can also remove the door opener if you don't need it anymore.

We think these devices provide the best value for your time and money.

DIY door openers

On a tight budget? It's possible to build your own door opener.

In fact, with a double-action cylinder (strong enough to move your door) and a two-way valve, you can make a door opener for less than $1000. Add a couple of sensors and some tubing and connect everything in the robot's configurable I/Os and you're good to go.

When the machine sends a ''ready'' signal, you can trigger the corresponding I/O and open the valve to move the pneumatic cylinder. When the part is placed correctly, you can trigger the I/O back and move the door back to a safe position. It's simple and cost-effective.

The bottom line: If you are buying a brand new machine, take the option of an automated door. It's generally worth the price. If you're keeping your old machine, consider opening and closing the machine door with the robot instead of an automated signal. It's slower, but it works.

Video: Robot-operated door.

Video: Actuated door.

Connect your robot and machine

Somewhere in the fields of Mississippi is a man with a CNC machine job-shop. Tommy Caughey has found the solution to his problems with Universal Robots. He was able to double daily production by running unattended shifts with his UR10. We already showcased his story in the WALT Machine case study. Today, Tommy brings us to an earlier stage in his automation journey, at the exact moment he got the robot and established communication with the CNC machine by himself.

Machine interface

In machine tending, the interface between the robot and the CNC machine is an important part of the integration. This is not always a given for different machines produced by different manufacturers. In order to optimize your cycle time, you should make sure the two machines can talk with each other so they know each other's status. They will need to communicate statuses such as that the CNC program is done, the door can be opened, the vise is closed, the robot is in motion, and the part is in place.

> ***Quick tip:*** *Make sure your robot and CNC or other machines can share the same interface. Some robot models may not speak the same language as your machine.*

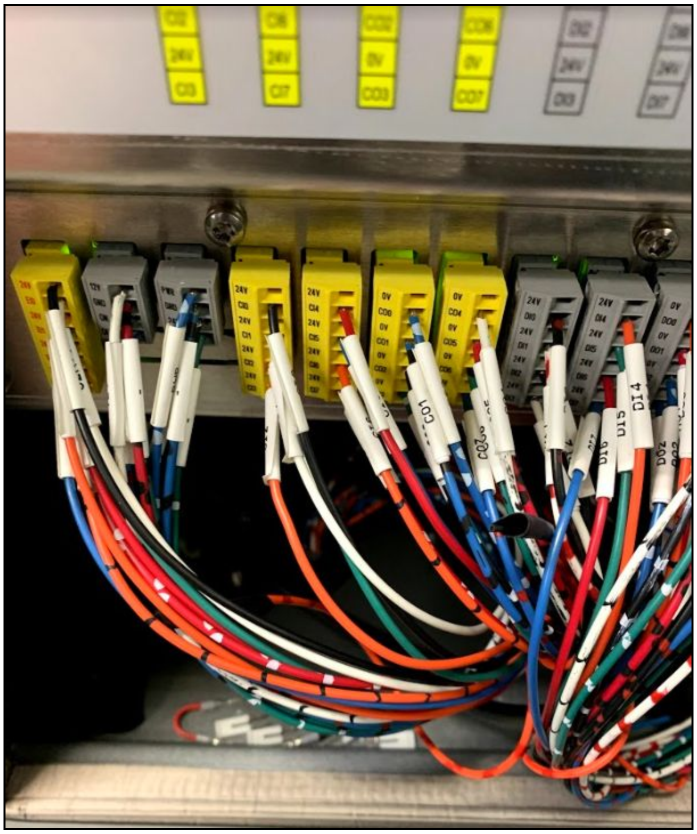
Signals for CNC operation

CNC machine tending is a widespread application because of its simplicity. Compared to other applications, the steps are straightforward and there are very few hurdles along the way. However, the biggest question mark when it comes to integrating a robot in front of a machine is communication. How do you get your machine to communicate with the robot? But in fact, the concept is very simple. You can break down communication to a couple of discrete I/Os and be good to go.

Type of communication

Industrial robots usually handshake with a CNC machine using a PLC or an external "robot-ready" board. In fact, this requirement is usually due to the complexity of the cell. The robot and machine are talking to each other to manage safety sensors, complex tooling, and other external components. Since the collaborative robot is easier to use and safer, there is a lot less communication to have in between the robot and the machine.

Universal Robots offers a couple of communication options. Ethernet/IP is a more complex communication protocol that can be joined to a PLC and manages a different component of the CNC machine tending cell. But what we see more often is simple I/O communication with very limited signals. The robot is usually the master and the machine the subordinate. In other words, the robot sets the different states of the program and manages what goes on.



Signals can seem confusing, but don't worry: we've got you covered.

Essential signals

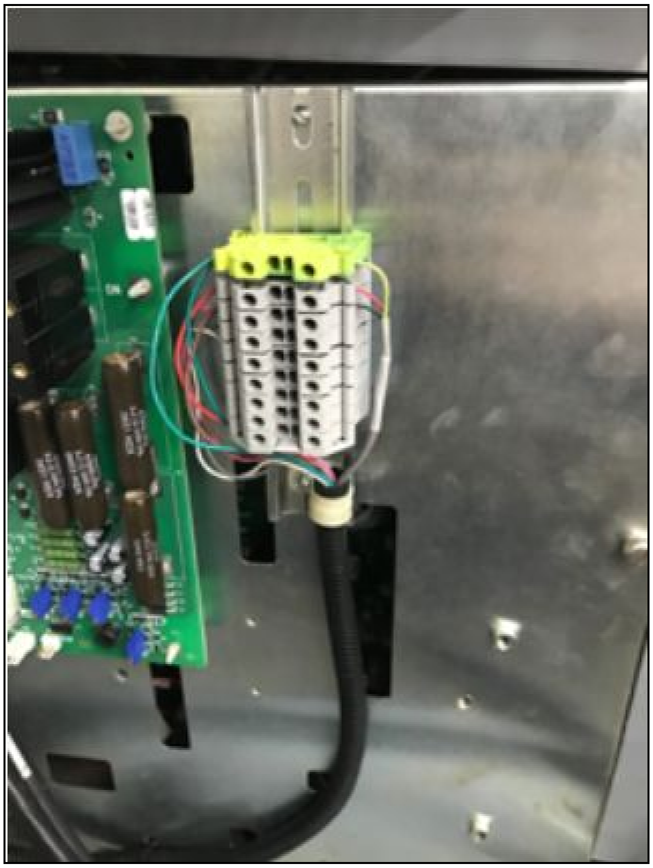
At the simplest level, here are the signals that must be exchanged with the robot:

- Machine cycle end
- Part picked by robot
- Robot out of machine
- Machine ready to start

Only two I/Os are needed to operate the robot in front of the machine. What we typically see in the industry is a subprogram waiting for the state to change, like this one:

```
Sub_Program_CNC_HANDSHAKE
        Wait_CNC_M21Inp=TRUE
        Set Safety_signal=TRUE
```

Here, the program is calling a subprogram that will first wait for M code number 21 to trigger, then set an I/O called Safety_Signal to true. This means that the robot won't run until the Safety_Signal isn't triggered.



Couple relays inside the CNC controller

Once the steps inside the machine are done, and the robot is out of the machine, another subprogram sets the Safety_Signal to false and reverses the state of M code 21 so the machine can go back in production.

```
Sub_Program_CNC_HANDSHAKE_BACK
    Set Safety_signal=False
    Set CNC_M21Inp=False
```

What we usually see is that the M code available is plugged into a relay, and the relay is connected to one of the robot's configurable I/Os. Depending on the machine setup, you might need to add a relay to jump-start your machine start button.

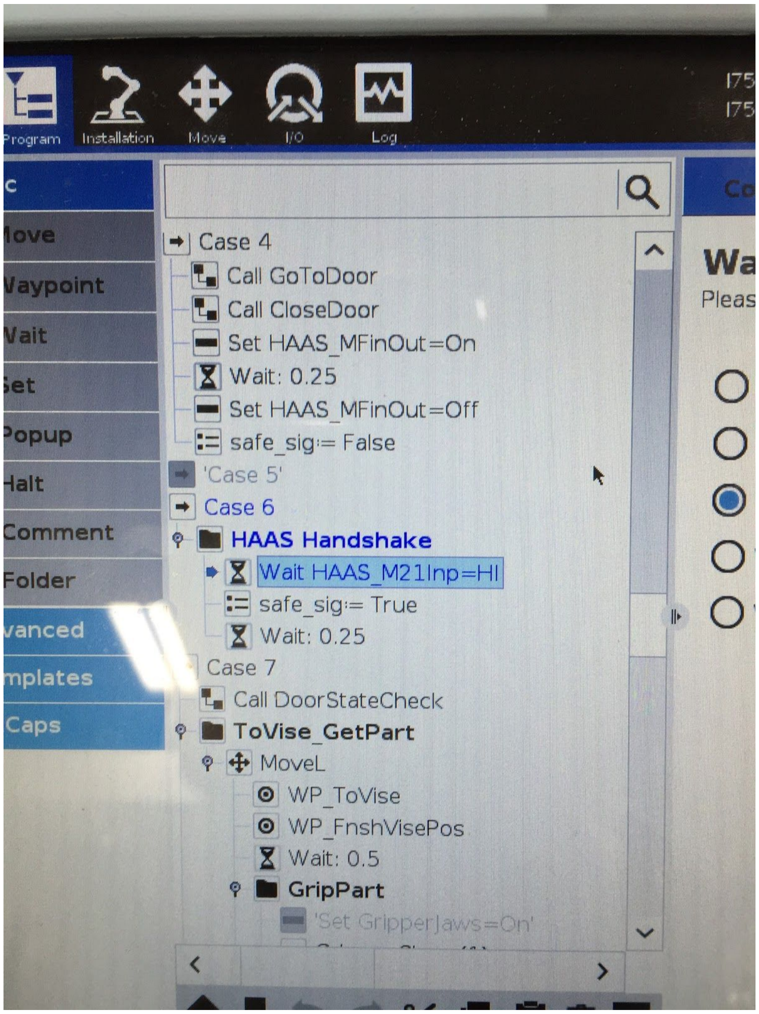
Other signals

Above, we covered the basic signals. Obviously, if you are controlling the machine vise passing by the machine controller you will need to read the state of the vise. To confirm that the doors are correctly opened or closed. You can exchange a few other signals between the robot and the machine, but don't add more just for the sake of it; stick to signals that are actually priorities for you.

Other signals typically use the same kind of connectivity. Make sure you choose a relay that can be triggered under 24V voltage, since that is the operating current of the UR robot.

Those are the basics. If you're getting into a more complex setup you might need an I/O expansion for the machine or robot. However, most of the installations we've seen out there have been done with the basic setup.

We recommend starting with the robot I/Os outside of the machine. Make sure you understand what you're doing before you dive too deep into the machine signal configurations. Note that most machine manufacturers provide instructions on which M code to use to trigger an external relay. Make sure you go through that documentation before connecting everything together.



Error signals

In a human-operated cell, whenever a CNC error pops up on the controller screen, the operator knows to stop and check the machine. This is not the case for a robot-operated cell. It's dangerous to leave the CNC machine running despite an error warning, so you will need to set up the proper reactions to any error message in your program.

> ***Quick tip****: You can track an error IO signal and trigger an Insight Tag. This tag can be linked to emails or SMS notifications sent to the operator, who can manage the situation and keep the production running.*

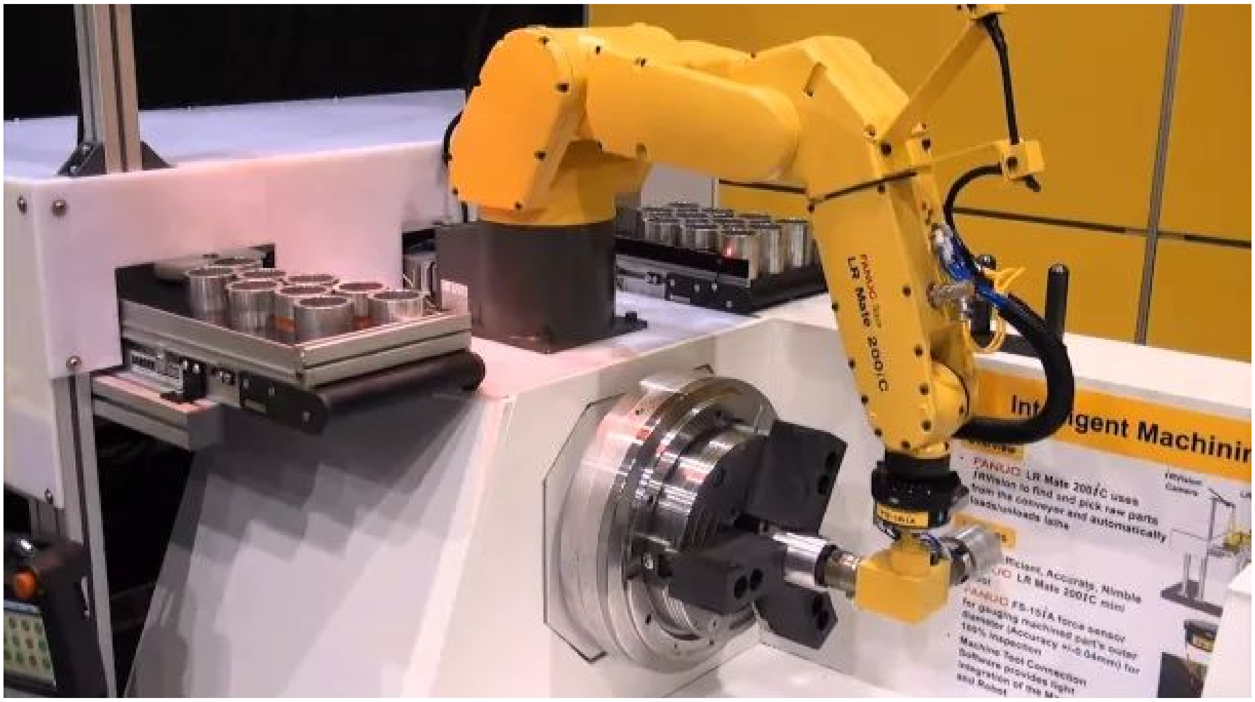
> ***Quick tip:*** *Make sure you have a way to confirm that the part has been correctly grasped by the vise and gripper. We've seen a few cases where such conditions were not respected, and the machine was damaged due to an unclamped part flying around the CNC.*

> ***Quick tip:*** *Did you know your cobot can expect the unexpected? Set up the Grip Check function on Robotiq adaptive grippers to not only detect problems, but solve them—it works with object detection, size validation, operation reiterations (grip retrials), classification, and much more. In manufacturing, robust programming helps you start production faster.*

Validate finished parts

Depending on the part's complexity, you might be able to insert a metrology device in the cell. Whether that's a camera that measures certain diameters, or probes placed directly on the end effector, use a measuring method that verifies the part's critical dimensions. Your process must be verified, so choose a verification method that suits your application. Metrology can be done afterward, but don't wait too long to verify your process because the situation can change quickly. If, for example, a tool is broken in the machine magazine, you don't want to have to redo the entire production run.

Several techniques can be used to measure features on a part. You may want to use the tool inside the CNC machine to make sure the part has been finished correctly, or add an external device or robot tool. For instance (and as shown in the video below), a force-torque sensor can measure the external diameter of a part.



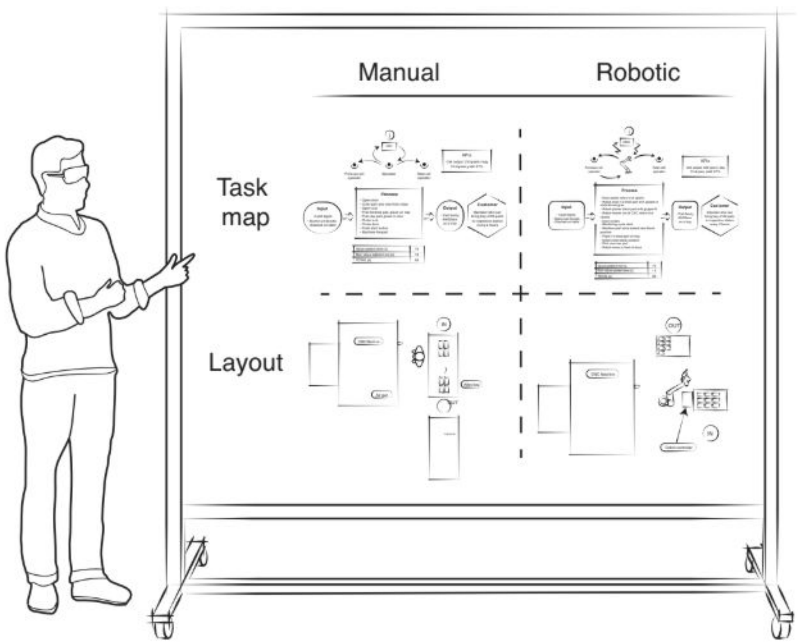
Video: [Measuring a part's external diameter with a force-torque sensor.]()

Robots can perform a variety of sub-assembly tasks, such as fastening parts together or inserting sealant. Be aware of what your robot arm might be capable of doing to save time in your production process.

Manual-robotic comparaison

The comparison step is the final part of the design phase. This step is like superposing two transparent slides to compare the differences. Putting the two layouts and processes next to one another will help you figure out the potential gains (or losses), how the robot will affect your production process, and what needs to be adapted from the manual to the robotic cell.

Keep reading to find out what you need to discuss during your manual-robotic comparison.



Layout

This requires close collaboration with the team that currently handles the manual process. When comparing the before-and-after layout, you should identify:

- Differences between the cells.
- What needs to be fabricated/produced or purchased for the robotic cell to work correctly.
- The impact on the previous and/or next cell in the process.

These things must be sorted out during the design phase so you don't have to think about them once you are installing your robot. The more prepared you are before getting the robot, the sooner you'll be in production.

Identify customer

CNC machine "customers" (that is, the next step after the robot cell in your production process) are usually another CNC machine or quality control station. This typically doesn't change when you integrate a robot in front of the CNC machine.

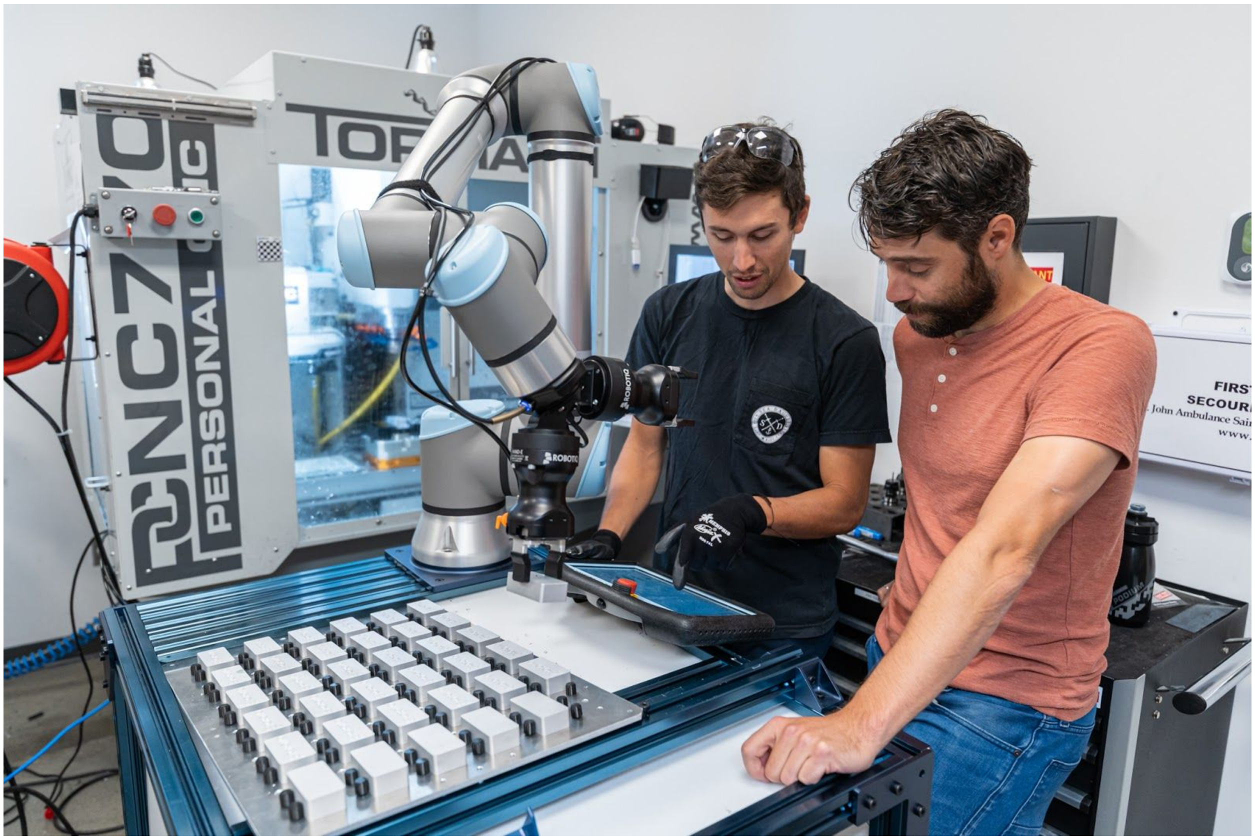
Define output

The CNC machine's output should be the same with or without a robot. If there is a difference, take note of it and be sure to make changes to the following cell so the production process can continue normally.

Define input

A CNC machine's input is usually a raw piece of material or a part in its first setup.

Remember that parts have to be slightly separated one from another so that the gripper can grasp them. The standard bin you had in place before might need to be replaced with an ordering device.



Describe process

Obviously, the loading and unloading of the machine will be different. The operator will be repurposed to another task. This means that all the steps that were previously done by the operator need to be covered by the robot. Think about all of them: cleaning parts, opening doors, etc.

Your cycle time will also be different (and probably more consistent)—how will this impact your process?

Document flow of information

Information can take all forms. In fact, in this specific situation, the robot will need to exchange information with the machine. They will need to exchange I/O signals and start/stop instructions to mimic the operator's actions.

The size of the blank is another piece of information that can be entered in the robot for the program to adapt accordingly; see if this is something you can make use of.

Finally, quality control is \ a piece of crucial information in the process. Make sure to have this part of the process ready to go before installing your robot.

Measure KPIs

Depending on your critical KPIs you will want to adjust them with your new production process. For example, if your main KPI is production capacity, there will be a difference (likely an increase) and you need to be ready to deal with that fluctuation. If your schedule was set with the human operator operating the machine, you may need to recalculate your output as a result of the robot.

Performing these comparison steps will highlight what you have nailed down and what you may have missed. It is important to have this discussion with your team before going forward. Getting the operators, engineers, and everyone else involved together is critical for a smooth integration.

INTEGRATE a robotic machine-tending cell

Off-the-line cell preparation

Assembling cell

Programming machine tending

Operation steps

Although there are many things to include when programming your robot, the following steps will form the pillar of your program.

Robot arm movement: All collaborative robots can be programmed by hand guiding. This is an easy way to program a robot since it doesn't require much experience in robotics or programming.

Gripper movement: This depends on which gripper you're using. A Robotiq Gripper, for instance, can easily be programmed using a special embedded feature in the Universal Robots teach pendant.

Raw part delivery and grasping: First determine how the parts will be picked, then set up an ordering method so the raw parts are easy to grasp.

Raw part placement: This can be challenging because parts usually have to be placed quite precisely in the CNC machine to ensure that the CNC program will run normally. To ensure repeatable part positioning, you should either add a way to mechanically lock the part in place or add force sensing to the robot gripper or tooling so it knows where to place the part.

Door closing and opening: The CNC machine must be kept closed for safety reasons. If you already have an automatic door, set up the robot-CNC machine interface. If not, program the robot to open and close the door by itself.

Finished part grasping and delivery: After the robot grasps the finished part and removes it from the machine, it should place the part in a convenient location for transport. Make sure the finished parts won't damage each other, or the quality of your final product will be affected.

Other stuff: After that, you can make the robot do whatever crazy stuff you want. Clean the table, push some buttons, whatever!

Read more:

- Machine Tending Playbook
- How to do Machine Tending Using Collaborative Robots

Programming good practices

Here are a few programming best practices to help you program efficiently.

ackups

Losing all your work is frustrating. Instead of programming for hours and risking losing everything, take no chances and back up all your work on Insights. This lets you keep your robot program and installation backed up on a cloud platform that is totally independent of your robot.

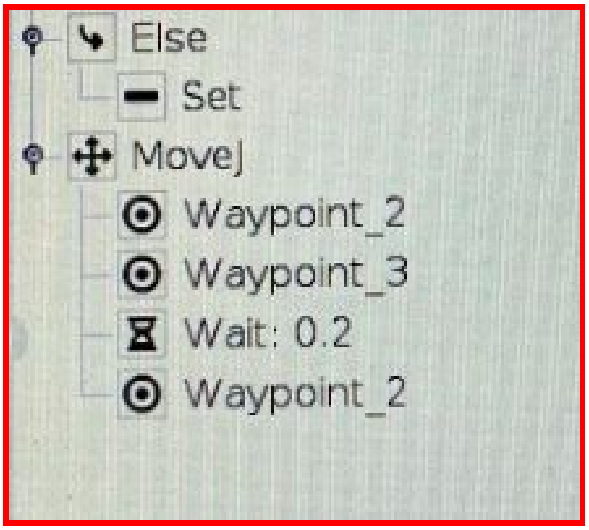
You can also do backups on a USB key or your own server. Don't forget to back up regularly to make sure you don't lose a day of work.

Relevant program names

You will need a program for each part you're machining, so you may as well name your robot programs after your parts (or at least something relevant to each part). If you're the one programming the robot and a night shift operator needs to start production in advance, you want them to be able to restart the robot correctly.

Variable names

One of the biggest mistakes you can make when programming is keeping waypoints and variables anonymous. This bad habit will slow you down in the debugging sequence, and it will be impossible for your teammates to debug or otherwise work on your program.



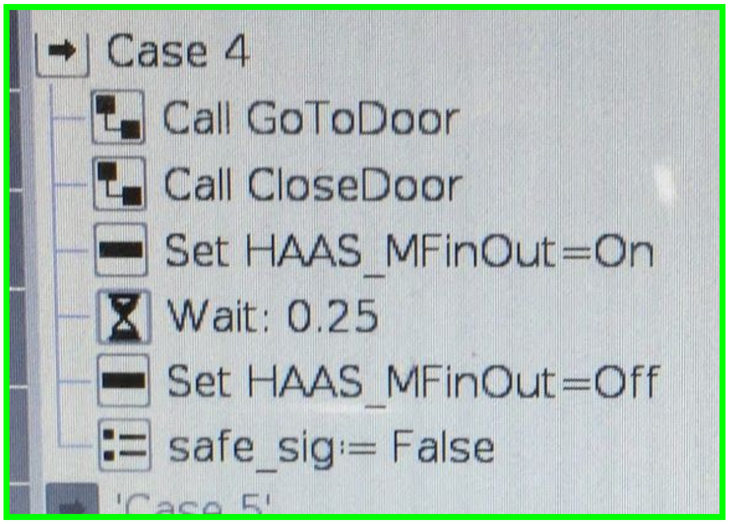
Bad variable and waypoint names.

Good variable and waypoint names. (WP stands for waypoint.)

Also, adding folders with specific names lets you lighten the reading of your program and more easily track where you're at.

Subprogram

Don't hesitate to create sub-programs, especially if you will be using a certain routine in several programs. In fact, instead of reprogramming the "Go to Door" and "Close Door" sequence, you can simply call the subprogram in your main program. This is a lot more work up front, but it will all be worth it when you're writing your next program.



Here, the ''Call'' sequence means the user has called the subprograms ''GoToDoor'' and ''CloseDoor''.

Structured program

A logically structured program will improve readability, which pays off when training, debugging, and making future improvements.

*''Having a general structure rather than a very specific one will make the program easier to adjust, modify and improve. Also, the more specific the program is, the more difficult it is for someone else to work with you on the coding. If this is your first robotic cell deployment, chances are it will not be your last. A general code will help you code the upcoming ones and may be usable in a future robotic cell that has a similar application.'' —**Annick Mottard, Robotiq Product Specialist***

Program comments

In complex programs, it might be a good idea to add a couple of comments here and there, especially if the name of a variable or waypoint does not mean much. In that case, insert a small comment to make things clearer for you and the future robot users.

> ***Quick tip**: Comments and folder names can have space, but not variables or waypoints. You can record status, next steps, and any other useful info for the next programmer. Sometimes, that will be you a few years later, re-diving into your code.*

Preserve the safety of people and equipment

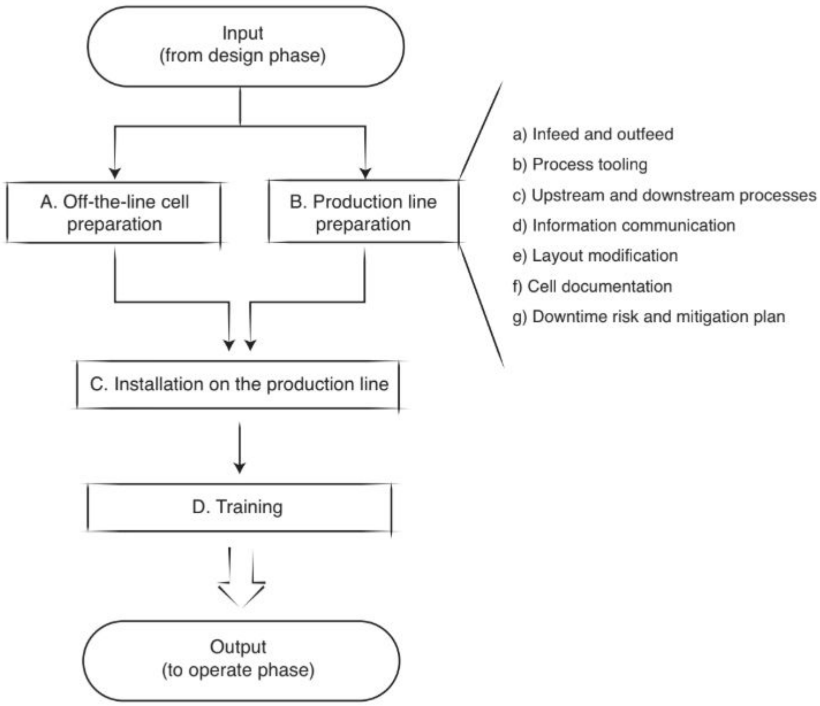
Programming a robot has to be done with caution. Otherwise you might program a point that is inside the CNC machine by mistake. Make sure to **test your program at a low speed** and with a hand on the e-stop.

Take the same cautionary approach with people: prevent people from getting close when the robot is moving, and be sure to avoid close calls. Let the robot run far away from people in the first run and see how it acts before bringing people too close.

With all these tips and tricks you should have a better experience programming your robot. Always remember that you might not be the person using the robot, so try to make it easy on them by programming strategically.

Production line preparation

The integration phase is not just about programming; in fact, it's mainly about testing. Building your program will take a few hours to a few days, but the testing and bulletproofing of your cell will be a much bigger time consumer.



Overview of production line preparation steps.

Infeed and outfeed

If you have followed the lean robotics methodology, the infeed and outfeed design should be done and ready to use. However, you need to test your robot and gripper to make sure it is working correctly.

One important tip when testing is to try out all possible situations. Let's say that the part is off a little bit or there is no part present: what should the robot do?

Also, you will need to manage the beginning and the end of your pallet. If you empty your pallet, what happens? Are you stopping your robot? Would you like someone to be notified? All these questions must be answered and addressed.

Concerning the outfeed, remember to manage quality control as well. Would you like to place a part aside to measure it? Would it be useful to have an extra operation to clean your part and try it out before storing it?

Process tooling

In machine-tending applications, the processing tool is the CNC itself. The interface between the robot and the machine is the workholding. Since you have chosen a workholding that can be activated by the robot (or by the machine) in the integration phase, you will need to make sure that the part is always placed at the right spot when starting the machining process. Using force control can be a good way to place the part correctly in your workholding. Avoid placing the part using only position instructions, as those can result in misplacement.

Speaking of workholding, you will want to have vise jaws that will follow the recurrent jobs. In fact, if you know a job will come back in your production, build a package of jaws, fingertips, and other accessories that can be reused for the next production run. By doing so, you can bring back this job without having to re-do the entire setup.

The next step of the integration phase is setting up information processes. How will the information about the different jobs be fed to the robot, and how can the robot help you manage your production?

Communication processes

With robot integrations, we tend to focus on the robot. But what we should be focusing on is the communication between the machine and the robot!Upstream and downstream processes

Manufacturing design can be an important aspect of CNC machine tending. Let's say you need to add or remove a feature on the part to be able to grasp it with the gripper.

You also need to adapt the next cell if there is a change in the manufacturing process because of the robot.

Information Communication

You might need to digitize the way some information is transmitted to the cell before you install the robot. Otherwise, not receiving the right information might result in a malfunction of the robot cell.

These steps can also mean that you will need to adapt to what the other machine will receive as information.

Read more:

- [Quebec-Based Company APN Increases CNC Machine Uptime]
- [Increase CNC Machine Uptime with Robotiq's 2F-85 and Wrist Camera]

Documentation & procedures

In the integration phase, you need to put in the work! The layout must be changed, and you need to document your cell and decide who will be responsible for any problems in the cell. Again, you need to prepare these steps in advance since you don't want to be bogged down once you go into production.

Layout modification

It's time to make the switch! Go back to your planning and layout comparison, and move the cell accordingly. Try not to improvise too much. If your team has done the cell design work carefully, it might impact production if you suddenly set the cell up differently than what was planned.

If you predict that there will be movement in the cell and that the robot will change positions over time, there may be a way to compensate for those movements. Machine Tending Copilot software will allow you to run your program even though the robot or vise has moved slightly.

Cell documentation

Documentation... who needs it? Well, once you go into production you want your robot to be running non-stop. If something goes wrong, you want to know who to contact and what to buy in order to restart production quickly. Having a BOM of your cell and the list of your service providers is crucial. Don't wait for a problem to make you realize you need it!

Downtime risk

When unexpected downtime occurs, who will you contact? In an ideal scenario, the robot will notify the right person when a problem occurs. The best way to do that is to install a monitoring device such as Insights. The robot will send you an SMS or email notification when it is down or if there are no more parts in your tray. You can also add custom events to be notified about other conditions in your program. Finally, using this software you can also monitor your cell efficiency and part output.

Stable processes

Automating a process means that nobody will be watching the machines for a potentially long time. If the process is unstable, there will be many opportunities for problems to arise and all your production efforts could be wasted. So to avoid a disappointing robot integration, make sure to bulletproof your process before you let it run unattended.

Treat your robot the way you would a new employee. Start by giving the robot a simple task. After a few hours of operation, you'll inevitably find parts of the program that need changing. Make the corrections. Only after the program is solidly up and running should you strive for a new application or increase the operations in the same cell. Robots can't do everything, so let them handle small tasks and leave the human operators to concentrate on complex ones.

Read more:

- How to Do Process Analysis for Machine Tending
- What You Need to Look for in Machine Tending Automation
- Machine Tending Playbook
- De-Risking a Concept Spreadsheet

Murphy analysis (or process failure mode effects analysis, PFMEA)

The Murphy analysis may as well be called "how not to crash my robot assessment." There are many variables outside your control; a Murphy analysis helps you identify these potential problems and find solutions to them.

If you've noticed a potential problem but are convinced it will never happen, trust us: it will happen. Make sure to do a full Murphy analysis, also known as a PFMEA. This technique will allow you to go through all the machine and robot steps and prevent potential errors.

Read more:

- 10 Steps to Conduct a PFMEA
- De-Risking Concept Spreadsheet

Setups

A robot cannot perform machine setups, but your employees can. Since they don't have to tend machines anymore, they can reduce downtime by preparing machines for the next production run. This will increase spindle time and allow you to output more parts.

Read more:

- Machine Tending Playbook
- What You Need to Look for in Machine Tending Automation

Training

Managers have to define the project's raison d'être—the reason why robots will benefit everyone in the company—and ensure everyone understands it. In addition, it is important not to underestimate the importance of everyone's role. It's a good idea to list all the roles people will play in a robotic cell deployment project in order to keep track of who's responsible for what. A template is available on leanrobotics.org.

Training the team who will manage the robotic cell is perfectly aligned with the first lean robotics principle, *people before robots*. It will not only make cell operation and maintenance safer but will also inspire your team to see the robot as a tool to be mastered, rather than a job-threatening machine.

Training plays a role on all three fronts:

1. Proper training will allow the operators to autonomously run, maintain, and troubleshoot the cell.
2. Training will increase the operators' mastery of robotics technology.
3. Training can clarify how the robotics effort fits into the bigger picture, and how the operators are contributing to this grander vision.

Well trained operators will be able to maintain the cell properly, avoiding downtime. And when unplanned downtime does inevitably arise, they'll react quickly and efficiently.

Thus, your investment in training will yield a great return by making everyone more productive: the robotic cell itself, the cell operators, and the engineering team that works on future deployments. Plus, it will boost your team's morale.

When done well, robotics training can take your business to new heights. But, how do you make sure it's effective?

When it comes to collaborative robotics, there are huge benefits to providing training systematically from the top down. One of the most compelling reasons to choose in-house expertise is that it gives workers a feeling of ownership of the robot. It makes sure everyone is on the same page, which leads to better innovation in products and robot applications.

Robotics training is well worth the investment, but you have to plan it in the right way.

The solution for effective training

To ensure your training program will be effective, start from your business needs and work up. This means following a series of structured steps:

1. Assess the big-picture goals of your business.
2. Align these goals with the benefits of in-house robotics expertise.
3. Quantify your existing resources.
4. Assess which resources are most valuable for your move to in-house expertise.
5. Get your stakeholders on board—both the technical team and other affected groups of employees.
6. Identify key applications for robotics.
7. Use these key applications to determine which new skills are necessary.
8. Finally, you are ready to look at the types of training available and implement a training plan.

Select your training type

There are many training options to choose from, with varying levels of price, time, and attention required.

Below are some of the most popular methods for robot training.

- Less costly:
  - On-the-job coaching and mentoring
  - Job shadowing
  - Self-directed study
  - Video presentations
  - E-learning
- More costly:
  - One-on-one tutorials
  - Seminars
  - College courses
  - External group workshops
  - In-house consultants

A good training program will use a mix of different types.

You will probably need an external supplier for your initial robot training. However, as the level of in-house expertise increases in your business, members of the team can take over some of the training burden. This is why it's so important to involve your stakeholders from early on; otherwise, they may resist what they see as extra work.

The simple process for creating an effective training program

We've created a series of eBooks to help you out. Arranged into 10 modules, together they guide you through the whole process of developing a robotics team, from assessing your business needs to implementing your training program and measuring the effectiveness of the program.

Read more:

- Module 1: Why You Need In-House Collaborative Robot Expertise
- Module 2: Efficient Meetings and Project Management
- Module 3: What Robotics Expertise Do You Need?
- Module 4: Who Should Have Robotics Expertise?
- Module 5: Which Applications Require Robotics Expertise?
- Module 6: Build a Robotics Learning Program
- Module 7: Build an Individual Learning Plan for Employees
- Module 8: Implement Metrics for In-House Expertise

- Module 9: Get Feedback From the Training Sessions

OPERATE Your Robotic Machine Tending Cell

Monitor

Once your robot cell is up and running, it is important to monitor its performance. The first few days might be hectic, but once production is stable you should be able to notice an increase in your production. At this point you should go back and look at your manual cell KPIs to see the magnitude of your production increase. You can monitor your robot with nothing but plain old-fashioned observation, or you can use robot-monitoring software as well.

Connected robots

Connectivity can give you a better idea of how well the robot is working. You might be able to use a simple device or tablet that will tell you everything you need to know about your robot, including working time, the number of parts produced, and a whole lot more. This is not only useful for preventive maintenance, but also for getting alerted when the robot has a problem or if it has finished its task.

Robotiq launched the first software for optimizing Universal Robots performance. Insights is a web application that takes robot cell deployment to another level, letting users find out how productive their robot was in any given period of time.

Insights is software designed to help you integrate your robot faster and maximize production. It sends you a text message when the robots requires your attention and provides real-time data to monitor, troubleshoot and improve your production. With Insights, you can now get access to the robot and fully control it remotely. Use Insights to get support from your favorite robot experts, wherever they are. It is currently available in North America, Latin America, and Europe.

Read more:

- Robotiq's Insights Now Provides Remote Access to the Robot Cell
- How to Take Full Advantage of Insights

Return on investment

How much does a machine-tending cell cost?

The “do-it-yourself” kit

You can build your own custom cell with a collaborative robot and an end effector. This is probably the cheapest option, with robots available for as little as US$25,000 and inexpensive grippers at around $5000. Programming can be done by hand-guiding, so this setup is easy to integrate and quick to earn an ROI.

The “I don't want to spend time integrating” kit

Some robot manufacturers and robot integrators offer pre-engineered machine-tending cells. Based on either a collaborative robot or an industrial robot, these kits come with everything you need for your machine-tending application. Once you have the kit, here’s one good sign of a productive machining process: chips flying off!

Robotiq uses its best off-the-shelf components to bring you an optimized kit for CNC machine-tending applications. Here’s what you will find in the kit:

- Dual Hand-E grippers with an angled bracket (90°)
- Fingertips starter kit, including a fingertip extender for larger parts
- Machine Tending Copilot software, including Force Copilot and Contact Offset
- Robotiq Wrist Camera (optional), including the new Visual Offset function
- eLearning course on how to best use your kit

The “all-in-one” kit

Now we’re talking about a major investment. If you want a setup that handles many steps for your CNC machine or carries huge parts, you’ll have to pay more. You’ll need an industrial robot, safety fencing, a dedicated space for the machine-tending cell, and someone to program and synchronize everything.

Some companies offer built-in programming kits that are easier to work with, but these still require a huge investment of time in terms of training workers to use the robot. These cells start at $100,000, and there's basically no upper limit. You can introduce all the fancy accessories you could think of. This kind of cell must be set up by a qualified integrator.

Comparison

Here's an overview of the costs you can expect from adding a robot compared to. other options.

Option	Implementation cost	Incremental cost	Suitability
Adding a CNC machine	About $250,000.	Labor to tend machine.	Best if existing machines are nearing end of life.
Outsourcing	None.	Per part, per contract.	Best if cash flow is an issue.
Adding a robot	About $50,000, including end effector and integration.	Labor to manage robot.	Best for increasing per-machine productivity.

Read more:

- Robotic ROI calculator: How long to pay back your robot investment?
- Increasing CNC Productivity: More Machines, Outsourcing, or Robots?
- What the Heck is Robotic Machine Tending Anyway?
- Machine Tending Playbook

Payback period

One reason to use a robot is to raise production. Use a time-based ROI calculation to make this benefit easy for everyone to understand. It's easier to understand earning back the investment if you describe it as "within six months" rather than "after 1,273 parts."

Use the following data to calculate the payback period:

- Average spindle time before introducing the robot
- Production rate before introducing the robot
- Profit on the finished parts
- Hourly machine cost, including employee cost and machine cost
- Robotic cell cost
- Average spindle time after introducing the robot
- Production rate after introducing the robot

To calculate the payback period, find the difference between spindle time before you added the robot, and after:

$$Payback\ period\ =\ \Delta\ spindle\ time\ =\ (robotic\ cell\ cost\ /\ machine\ rate).$$

For example, if your robotic cell cost $50,000 and your operating costs are $100/hour, the payback period will be as follows:

$$\Delta\ spindle\ time\ =\ (\$50,000\ /\ \$100\ per\ hour)\ =\ 500\ hours.$$

This means you will earn back your investment after your machine has performed 500 more hours than it would have normally. That is, after your robotic cell has performed 500 more hours of machine tending than it would have if you had kept it as a manual cell.

Now, let's say you will run the machine autonomously during the night shift, which represents 40 hours per week:

$$Night\ shift\ payback\ period\ =\ \Delta\ spindle\ time\ /\ hours\ worked\ per\ week\ =\ 500\ hours\ /\ (40\ hours/week)\ =\ 12.5\ weeks\ or\ 3.12\ months.$$

You can also calculate the additional profit you will earn from producing extra parts (as a result of the increased spindle time):

$$New\ machine\ rate\ =\ old\ machine\ rate - (\Delta\ production\ rate \times profit\ per\ part).$$

For example, if your production rate goes from 10 to 12 parts per hour, and you earn a $5 profit per part, your new machine rate is as follows:

$$New\ machine\ rate\ =\ \$100/hour - (2\ parts\ per\ hour \times \$5\ per\ part)\ =\ \$90/hour.$$

Calculating the new machine rate makes your ROI even more obvious, although the payback period alone is usually enough to convince management to add a robot to the shop floor.

Your next robot

Because the return on investment is not about how much you spend, but how fast and accurately you can produce your parts, you need to keep these data accurate and easy to use. If you present all this info to your manager, you might find that you are to have a couple of extra production boxes that will provide enough of an ROI to allow you to order another robotic cell.

Here are some rules of thumb when it comes to payback periods:

- If the project can be reimbursed within three months, it's a no-brainer.
- Within six months, it's generally a "go."
- If it's a year, you may need to push a little harder and provide further explanation.
- At two or more years, it might be hard to convince your manager (but not impossible).

Of course, if your first robot project succeeds with a quick ROI, you'll have a great argument for why management should consider another robot for your shop floor.

Read more:

- Machine Tending Playbook
- Getting Started With Collaborative Robots (2018 Edition)

Let's start production faster

Manufacturers work with Robotiq to take control of their robotics projects and improve productivity and quality. They choose our flexible Plug + Play components because they can be deployed easily across many stations. Our community of experts empowers manufacturing engineers to quickly deploy their robotic cells and build their automation skills.

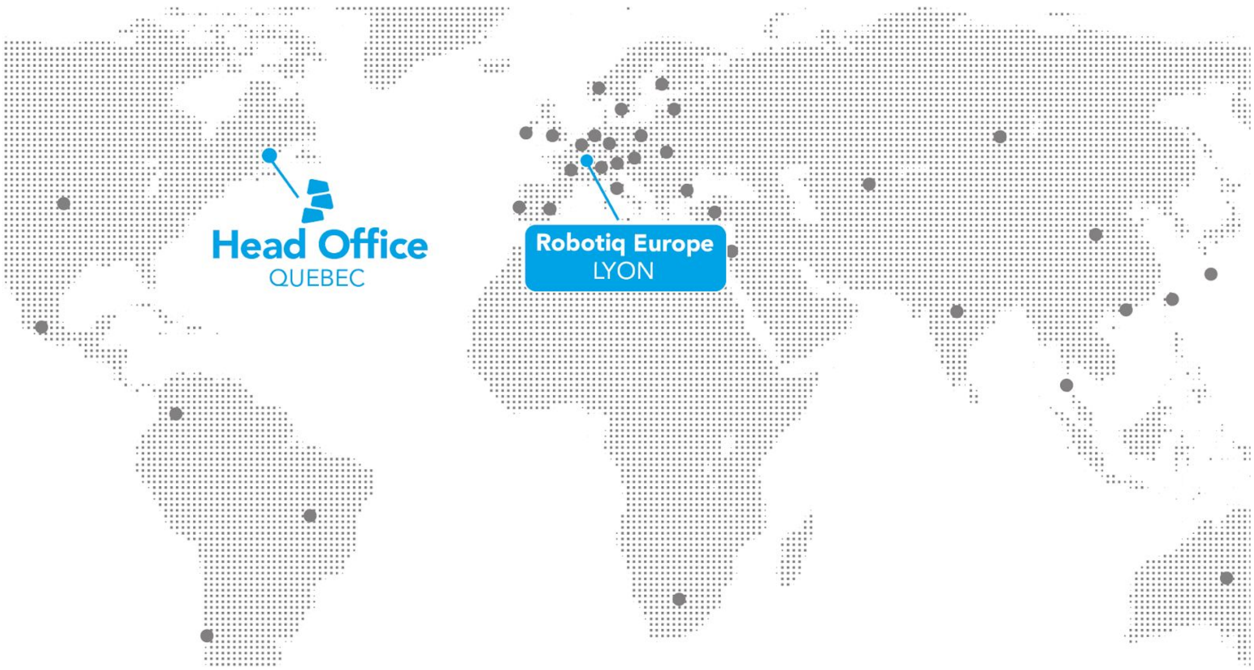
Robotiq's tools and know-how simplify collaborative robot applications, so factories can start production faster.

About Robotiq

Robotiq's lean robotics methodology and products enable manufacturers to deploy productive robot cells across their factory.

They leverage the lean robotics methodology for faster time-to-production and increased productivity from their robots. Production engineers standardize on Robotiq's Plug + Play Components for their ease of programming, built-in integration, and adaptability to many processes. They rely on Flow's software suite to accelerate robot projects and optimize robot performance once in production.

Robotiq is the humans behind the robots: an employee-owned business with a passionate team and an international partner network.



Let’s keep in touch

For any questions concerning robotic and automated handling or if you want to learn more about the advantages of using flexible electric handling tools, contact us.

Join us on social media:

Workfloor: Robotiq’s Blog

Twitter

Linkedin

Facebook

Youtube

Robotiq’s community where industrial **automation Pros** share their **know-how** and **get answers**

Ask Your Question

Robotiq.com | leanrobotics.org