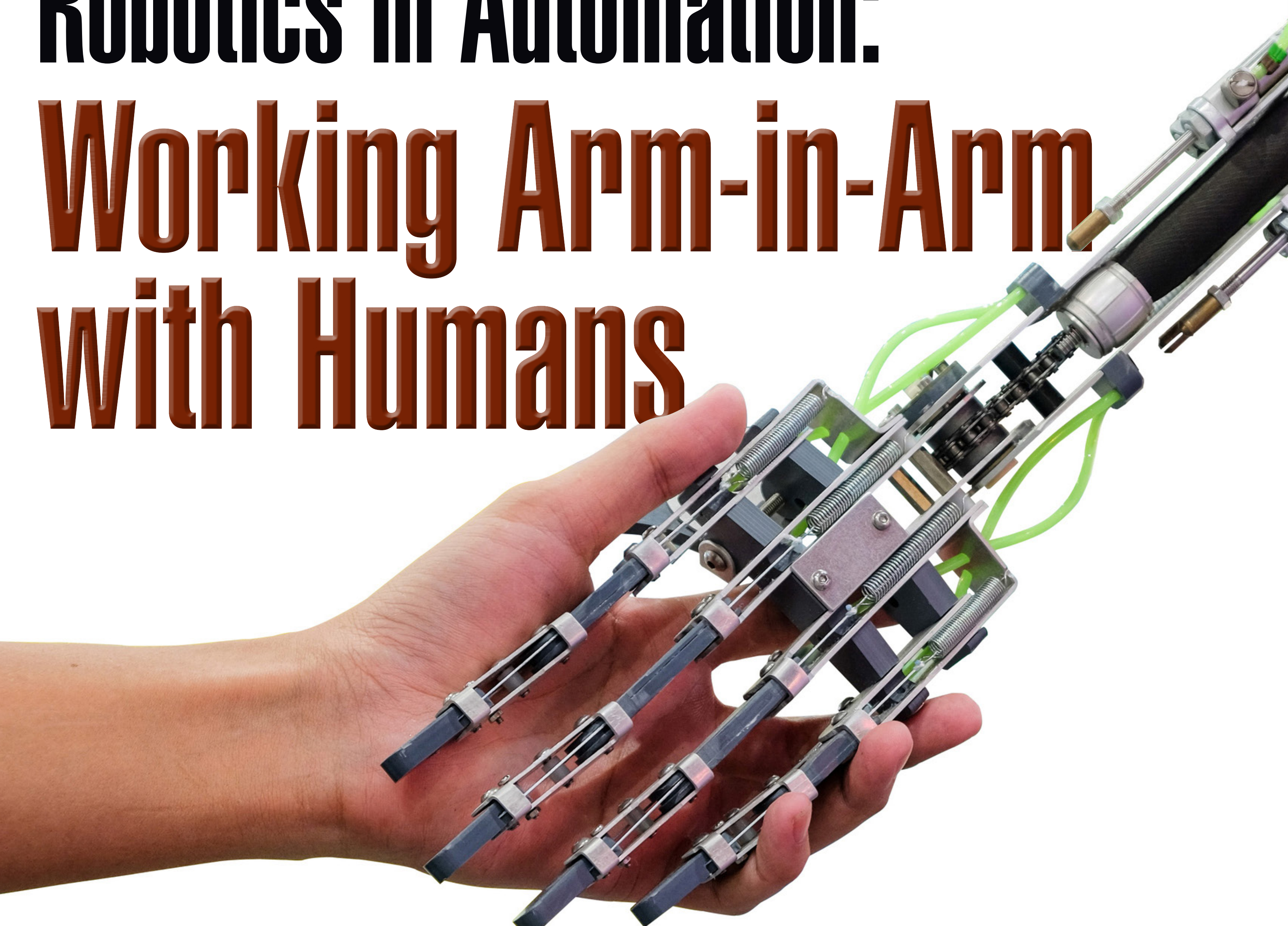
Machine Design® LIBRARY

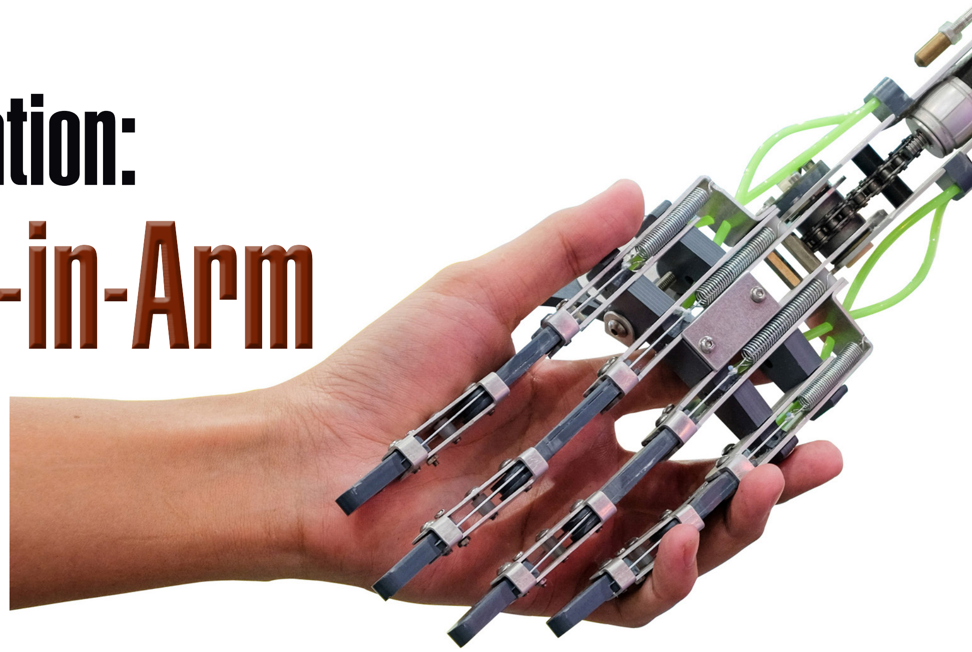
A compendium of technical articles from *Machine Design* and

PERFORMANCE MOTION DEVICES
MOTION CONTROL AT ITS CORE

Robotics in Automation: Working Arm-in-Arm with Humans

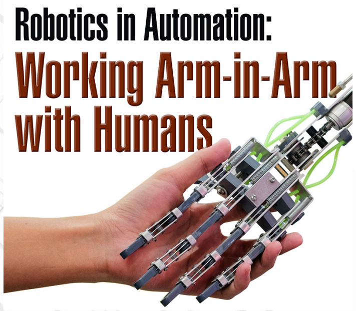


Robotics in Automation: Working Arm-in-Arm with Humans



Cover Credit: Chotchuang, Dreamstime

CONTENTS



Effects on Motors No Longer at Arm's Length

Bob Vavra,
Senior Content Director for Machine Design and Power & Motion

The relationship between humans and their robots continues to evolve. Where robotics was once a tentative toe-dip in the realm of robotic deployment, today we see a full integration of robotics into the operational infrastructure.

That change creates discussion about how best to deploy robotics effectively—both to improve productivity and safety metrics as well as to optimize the unique talents that humans still bring to the process. What we've learned is that the strategic use of both traditional robots and collaborative robots can enhance manufacturing's goals of increased throughput and effectiveness.

It does require a different way of thinking about robots and its human programmers. It requires a strategic initiative. Robots are no longer a one-off way to solve a singular problem. Robots have to be fully connected to the full enterprise—both to manage the process and to measure its value. Those who have done this, as our new e-book discusses, have found benefits that can be measured in dollars saved and production benefits.

The e-Book looks at the fundamentals of current and future trends for robot deployment, including how the data is being managed and funneled through the data pipeline as well as the size, flexibility, and ruggedizing of the robots.

We also look in particular at the medical device industry's use of robotics. The medical industry is among the most aggressive in its use of robotics for a wide variety of procedures—and it is also among the industries that are the most risk-adverse. The medical industry's continued example of how to design and deploy robotics points to their value as a solution and should provide confidence for other industries to follow suit.

Robots have surpassed their science fiction lore into smart, effective and efficient automation components. They are every bit as valuable as a PLC or a motion control device. They should be considered an essential piece of your automation strategy—with one important difference. Robots literally allow you to safely link arms with your device.

CHAPTER 1:

Five Key Trends for Tomorrow's Industrial Robots

TE CONNECTIVITY

As global robotics installations rise, the need for power, high-speed data transmission and connectivity intensifies.

According to the World Robotics 2021 Report by the International Federation of Robotics (*IFR*), the current growth for industrial robotics is anticipated to continue in 2022 and 2023, with more than 500,000 units installed per year worldwide by 2024. Here are five key robotics trends and technologies that are advancing the adoption of industrial robots and shaping industries around the globe.

Trend 1: High-Speed Data Transmission

The advances of Industry 4.0 are built on the ability to turn huge amounts of data into insights and intelligence. This requires manufacturers and robotics OEMs to identify the best ways to increase data transmission capabilities on the factory floor—including selecting between wired and wireless data transmission.

Wireless technology (Wi-Fi and 5G), however, may not deliver the durability and reliability needed in the factory, and connection can be impacted by electromagnetic and radio interference. Thus, manufacturers have relied on wired connections. Rugged RJ_{45} industrial connectors have been a mainstay for years, but new offerings in hybrid, single-pair ethernet (SPE) are making it possible to transmit high-speed data and more power via a single cable and connection.

Wired technology for data transmission in robotics offers the secure, real-time communication a robot needs to react immediately on the assembly line. In the future, there will be a place for both wireless and wired technology, including applications where both are required.

Trend 2: Miniaturization

Packing greater capabilities into ever-smaller components has driven incredible innovations and efficiencies in industrial manufacturing. The increase in mobile robots (AGVs

and AMRs) and cobots especially has made miniaturization an important element in the design of every element of the industrial robotics package—from relays and sensors to circuit boards and connectors.

Smaller and lighter relays and lower energy consumption are essential in the age of Industry 4.0. So too are compact, lightweight circuit boards. But smaller devices mean higher density, making heat dissipation critical.

In these early days of IIoT, compact interconnectivity between devices, components, control systems and the cloud requires a solution tailored to the application, ensuring the seamless real-time communication needed to drive machine learning and artificial intelligence.

Trend 3: Modular, Mobile Connectivity

In the smart manufacturing world of the not-so-distant future, production will run on a modularized system of workstations and production cells, rather than static assembly lines. To meet customization demands, manufacturers are turning to flexible and adaptable AGVs and AMRs. Working alongside cobots, they can enable a production line in which different tasks take place within different cells, or robots that pivot between workstations as needed.

Interconnectivity between devices makes it all possible but depends on a reliable stream of real-time communication. By definition, a modular design is flexible, therefore the connectivity solutions must be as transposable as the manufacturing system itself. Today's modular connectors are engineered for secure and reliable input/output connections. Many offer a unique locking system that prevents both mis-mating and accidental unplugging, whether from shock or vibration. The best offer compact, one-piece construction with preloaded contacts, delivering a space-saving, quick-to-install solution that gives engineers increased design flexibility.

Trend 4: Rugged, Robust Connectivity

In the factory, dozens of complex processes are happening every second. A single broken connection can bring a production line to a grinding halt. So rugged and robust connectivity solutions are a must.

Customers expect "rugged" connectors to be mechanically sturdy and tough, able to withstand the uncompromising vibration, shock, electromagnetic interference and dust of the industrial environment. Manufacturers don't want to even think about them, until it's time to decommission the line.

Reliable data connectivity is also critical. With the increasing quantity of data, as well as volume of data being transmitted between devices, sensors and components, manufacturers need connections that can consistently handle the constant stream of high-speed data transmission.

Trend 5: Single-Pair Ethernet (SPE)

In an industry that prides itself on efficiency, single-pair ethernet has clear appeal. SPE requires just one cable pair, instead of four pairs. By reducing the number of wires required from eight to two, SPE technology reduces costs, increases design flexibility and improves reliability. Plus, with the advent of hybrid SPE, manufacturers can transmit data and power through a single cable and connector, driving even more optimization while saving space.

Thanks to the SPE standard for hybrid interfaces, manufacturers can confidently invest in transitioning to SPE and more simply connect field-level devices to the network. Within the industrial network, SPE provides the infrastructure to transmit data seamlessly between all devices and sensors on the network to the cloud. And within the robots and cobots themselves, SPE makes it easier to pull increasing quantities of data and electricity from the base of the robot to the end of the arm and gripper without losing power, signal or data.

to view this article online, ☞ *click here*

☞ BACK TO TABLE OF CONTENTS

CHAPTER 2:

Robots Can Talk The Same Language

COLM GAVIN

By implementing universal robot libraries, automation engineers use a single software environment.

Driven by technological innovations, increasing applications, falling prices and greater adoption of factory automation, the use of robots is ballooning globally. From just 2013 to 2017, the number of annual industrial robot installations more than doubled in the combined American, Asian, Australian, and European markets.

The rise of robots is streamlining many corners of factory automation, ranging from automotive to pharmaceutical to electronics manufacturing, among many others. And like other innovations in manufacturing, the increasing utilization of robots comes with a set of challenges.

While robots often operate in concert with other machine automation equipment, proper configuration can be complicated utilizing available programmers in the best circumstances, and impossible without the assistance of specialists in the worst cases.

Modern robot libraries within integrated automation software—and artificial intelligence (AI)—improve on this situation, making cross-manufacturer robot integration more readily available to a broader base of automation engineers.

Compatibility Issues

In many factories where robots are used, complex moves are configured and programmed within dedicated robot controllers. These controllers use inputs and outputs (I/O) for data exchange with PLCs and other automation components. This separation of controllers presents challenges for synchronous operation among robots and other machine equipment, especially among a diverse group of manufacturers' robots, each with a different robot controller.

Each robot controller typically operates with unique software I/O structures, creating the requirement for a central PLC programmer to pair a different structure of code with each manufacturers' controller among otherwise similar robots. The programmer must also

Figure 1: In the classical robot configuration narrative, an automation programmer is not only responsible for commissioning plant PLC, HMI and machine equipment, but also the robot, its controller and HMI. All images courtesy Siemens

ensure correct code usage in each of the various robot controllers, according to the rules of manufacturer-specific software and hardware.

Additionally, there are often robot HMIs to configure. With so many configuration environments—each typically requiring specialized knowledge of its particular programming characteristics and language—the required engineering and commissioning efforts, along with propensity for errors, are high (**Fig. 1**).

Service and maintenance also are complicated, particularly with many components to troubleshoot when issues arise.

Using Universal Robot Libraries

A new narrative in robot configuration and programming is on the rise, more closely integrating multi-axis robot control with other machine automation PLC code, driven by robot libraries provided with integrated automation software suites. These libraries—which provide the means to drive robots entirely in a PLC program—are empowering engineers and tech experts to program robots on their own, while reducing reliance on expensive specialists.

By bringing robot control within PLC code, automation engineers can program robots without the need to understand the intricacies of robot control language. The robot controller remains, acting as an interpreter of PLC code blocks, but interpretation takes place intelligently behind the scenes of the PLC-commanded moves (**Fig. 2**).

This enables easy integration of robots with other machine automation components using the same and familiar PLC programming software to control multiple manufacturers' robots in a single subroutine, making time-sensitive robot interactions much easier to implement. In a typical deployment, there is no need to touch anything in the robot controllers.

While this functionality is especially useful for end-users, machine builders also benefit by increasing their ability to standardize automation programming. For end-users and machine builders alike, this functionality provides the ability to use the right robot for the desired job, with nearly identical programming and operational interfaces, regardless of the robot manufacturer.

These robot libraries also include standardized HMI faceplates, maintaining the same operational procedures, look and feel among multiple manufacturers. Within a robot faceplate on the primary machine HMI, programmed using the development software environment, engineers can place robots in jog mode to move them to defined positions. They can also teach robots positions and path points using built-in toolsets.

For new robots or paths, entire move trajectories are configurable in the graphical HMI

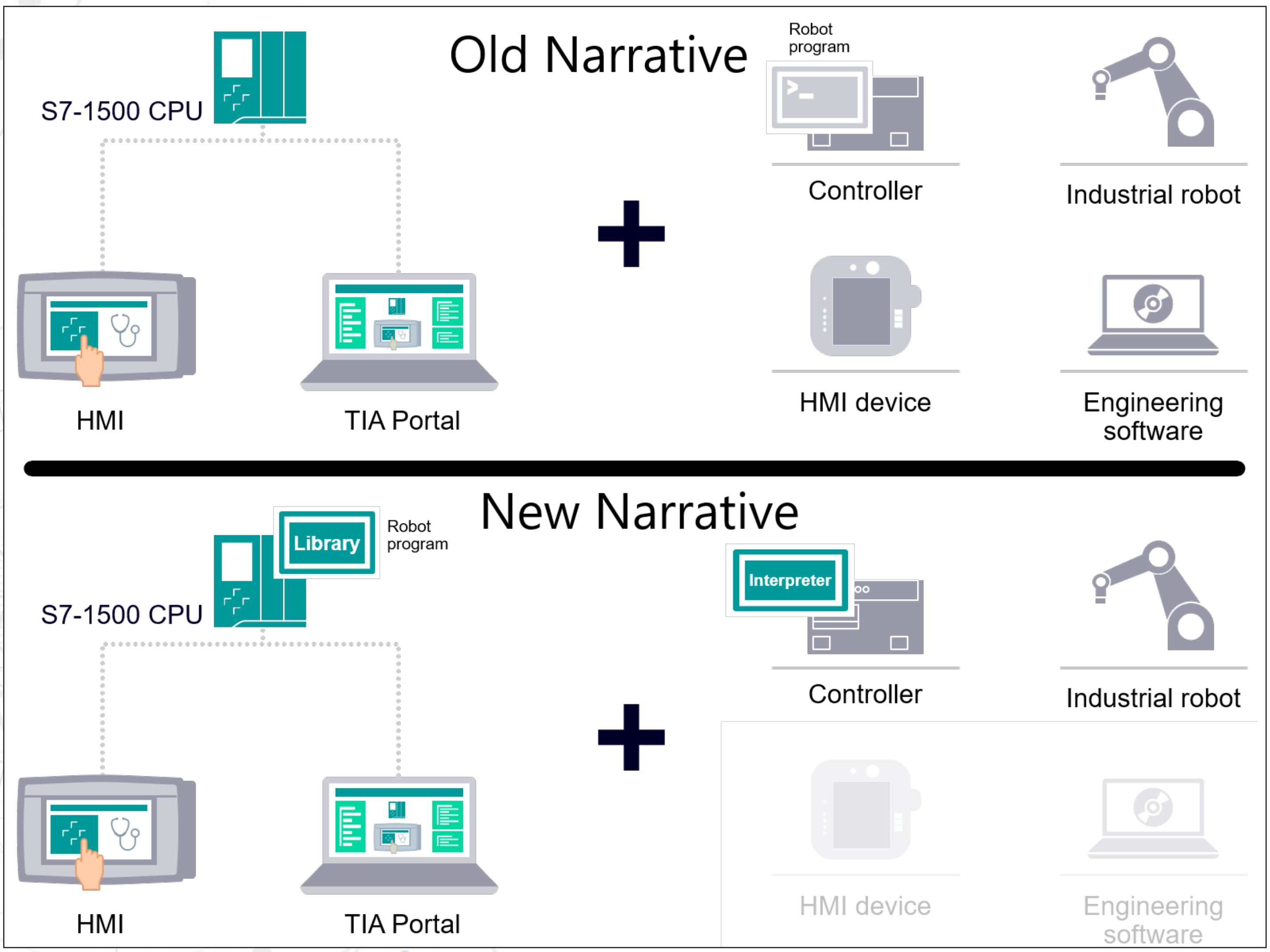


Figure 2: Siemens SIMATIC Robot Integrator software embeds robot control within a standardized PLC library, universal among multiple manufacturers' robots.

Figure 3: A picking robot improves its movement intelligence over time as it learns how to grasp and move products more efficiently.

faceplates, with the configured paths available for consumption and direct use in the PLC program. These universal HMI faceplates are ready-to-use in modern robot libraries, without any programming required—just relatively simple configuration.

The libraries serve as bridges to robotics, eliminating the need to change programming procedures or practices when switching among robot types or manufacturers. Some advanced systems even support simulation and AI-driven path planning, and the central location for all programming within a PLC makes edge computing with robots more achievable.

Robotic AI Enhancements

AI is further increasing the effectiveness of robots on the plant floor, with intelligent spatial assessment and smart object recognition automatically enabling robots to perform many tasks without the need for complex customized movement algorithms. This helps production lines scale up with limited programming effort required, and it empowers manufacturers to increase production while decreasing lead time.

Robots can work in tandem with visual analysis AI to verify quality standards are met, and to improve their own movement intelligence over time. Robot AI application examples include:

- Robotic arms with mounted cameras verifying products are free of defects.
- Autonomous guided vehicles in warehouses calculating and learning the most efficient routes for material handling.
- Self-driving cars identifying and avoiding road hazards and obstacles.
- Picking and packing robots optimizing product placement orientation (**Fig. 3**).

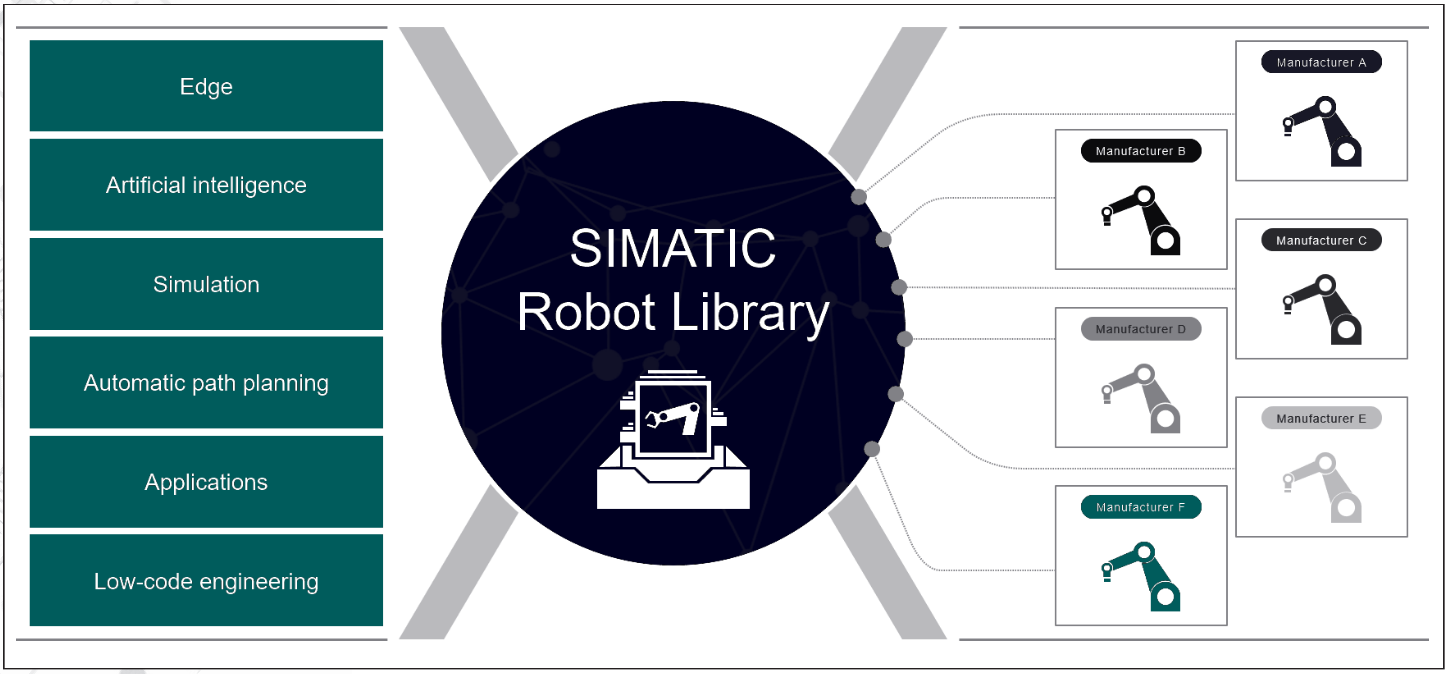


Figure 4: The Siemens SIMATIC Robot Library empowers automation programmers to control robots within a single PLC using identical programming blocks regardless of robot manufacturer.

Many robots still require operators or programmers to provide explicit instructions, but as these examples show, AI is becoming more prevalent. As this trend continues, robots will continue to carry out difficult, dangerous or demanding tasks otherwise executed by humans, freeing up skilled personnel to focus on innovative process improvements.

Identifying Results

A pharmaceutical manufacturer reduced programming and development time by about 30% by deploying the Siemens SIMATIC Robot Integrator software for use with its new and existing filling and sealing robots. Prior to the upgrade, engineers relayed messages for removing and placing hygienic labels between a robot controller and the PLC, which was driving other automated components on the line.

By deploying the Robot Integrator software, the manufacturer consolidated control to the central PLC, eliminating the need to write code in multiple robot controllers. This significantly reduced the programming learning curve because robot library code within the Siemens TIA Portal integrated automation suite is identical from robot manufacturer to manufacturer. Additionally, the robot HMI faceplates now provide the pharmaceutical manufacturer with the ability to quickly modify its recipes for pivoting between batches when required.

Save Time, Increase Efficiency

Robot integration libraries and advances in AI are increasing the value of robots on the plant floor by simplifying configuration, programming and maintenance. In some software environments, simulation and digital twins are available to help programmers tweak their code for more accurate robot moves prior to field deployment.

These capabilities provide automation engineers with more time to spend stringing together series of robot paths to get a job done—the time gained by eliminating the need to define the mechanics of each move, as well as data exchange between the PLC and robot controller. These capabilities also do away with custom robot software configuration and programming, making robot programming accessible to a wider base of engineers, regardless of the robot type or manufacturer (**Fig. 4**).

As a result, implementers are reducing project development time and costs, and benefitting from increased machine efficiency and manufacturing profitability.

COLM GAVIN is the portfolio development manager for Siemens Digital Industries Software, responsible for the promotion of digitalization topics for machine and line builders in the United States.

to view this article online, ☞ *click here*

☞ **BACK TO TABLE OF CONTENTS**

CHAPTER 3:

What You Should Know About Medical Robotic Systems

SEAN HÄGEN

As an extension of a surgeon, surgical robotic system design extends to inter-team communication dynamics.

With all the talk in the industry about medical robots, it is best to first define what that really means. For starters, there aren't actually autonomous robots that duplicate human decision making and fewer still that are semi-autonomous. However, there are medical bionic constructs in use today optimizing healthcare providers' performance through electronic or electromechanical devices.

The distinction worth noting here is that these medical constructs and devices facilitate and increase the abilities of the user, but do not replace them.

Unfortunately, it's likely too late to change the vernacular to something more fitting. Despite all the semantics, there is an important distinction that is key to design and development processes. In this article, we will explain two very important reasons why this distinction is critical.

The first reason involves regulatory strategy because it can significantly change depending on what kind of device is selected, especially when it comes to the level of autonomy, team usability validation and training. The second reason is in the understanding that these systems are an extension of a clinician's current ability and facilitates activities that they may be done manually. It can, therefore, make a substantial impact on the user interface design and development.

Strategic Regulatory Impact

Many robotic systems today are based on technology that could facilitate a procedure autonomously. Validating the safety and efficacy for FDA approval would be cost-prohibitive. The proof would involve having to demonstrate the robot safely managing contingencies, which can be unpredictable. Validation trials of a scale similar to pharmaceutical clinical trials would be required and involve hundreds, if not thousands of patients.

Unlike a medication—typically a high-volume consumable which is reimbursed by a

payee—a piece of capital equipment that is marketed to a healthcare facility does not offer the same ROI that could justify such a validation investment. Nevertheless, by having the clinician remain in control and the system consigned to an extension of their abilities, the human maintains responsibility for the result.

Maintaining the clinician as the decision-maker helps the validation process remain a function of the clinician's interaction with the robotic system's user interface. This means the user's clinical abilities are not to be validated—only the robotic system met expectations for efficacy, safety and performance. That said, both the design and usability validation can involve a lot more complexities than other medical devices.

Surgical Workflow Considerations

A surgical robotic system can be considered an extension of the surgeon's abilities and can impact the rest of the surgical workflow as well as all the other actors involved in the surgical procedure. With conventional surgery, whether it be open or laparoscopic, typically interactions between a team of clinicians are well orchestrated. The team usually comprises a lead surgeon, first assist, sterile nurse/tech, circulating nurse, anesthesiologist and possibly others (e.g., perfusionist), or duplicates of some of the actors listed. When it is a teaching hospital, interns and fellows may also be on the team.

The reason we need clear understanding of the roles of all the people involved, especially when introducing a new robot into the operating room, is to help achieve a favorable outcome for the patient. For instance, it can impact the manner in which the lead surgeon interacts with the team, especially if the robotic system's physical footprint requires some space in an already congested environment. Moreover, the team dynamics can be profoundly impacted if the team leader is isolated from the rest of the team and instead resides in a control console as part of trying a different team configuration.

Role of Communication in Different Scenarios

Even though the intention is to make the robot an optimized instrument for the surgeon, the surgeon's responsibilities and those of his extended surgical team can be greatly affected. For example, many surgical procedures require a sequence of different end effectors such as cameras, grippers, cutting forceps, surgical glue dispensers and blunt dissectors.

Even switching tools for robot-assisted minimally invasive surgery can require time-consuming removal and replacement. For such procedures, team members must interact with each other and the robot, while often the lead is interacting with them without face-to-face communication. Obviously, this can cause an added complexity because a substantial amount of the communication during many procedures is non-verbal. Granted, while the actors are wearing masks, they have learned to read body posture and facial expressions inclusive of the mask.

Such team dynamics requires consideration, and when it comes time to validate the user interface of the robotic system, the team should be included in the validation test. Moreover, depending on the type of healthcare facility, such as a teaching hospital or private institution, teams may or may not be cohesive with seasoned versus new members on the team. This is another important consideration for usability validation in which both team types should be involved in evaluating the design. Importantly, upstream usability engineering research must be conducted to inform the design and regulatory team of the requirements for future validation based on well understood use-related risk assessments (URRA).

Impact of Various Modes of Training

The URRA can impact training models for the robotic system, another regulatory path strategy which further complicates risk mitigation and the subsequent validation.

Traditional surgical devices that are usually manually operated typically do not require formal training. That is, training beyond orientation or an "in-service." There is a difference between orientation and formal training. Orientation is not considered a risk mitigation from a regulatory perspective. When a design control like training is a risk mitigation, robust documentation is required to demonstrate to the regulatory bodies the degree of control and repeatability that the manufacturer maintains.

Under the design control process, there is a protocol for how the trainer is trained, a record of who was trained, when they were trained and if and when subsequent training is required as well as required qualifications the training results in during the use of the system. The definition of "system" may include the robot proper, the control console, the robot drapes and the end effector's instrument attachments. Additionally, it often includes the cleaning and reusability of the instrument attachments. Obviously, formal training is a far greater and ongoing burden and responsibility on the manufacturer.

User Interface Design Impact

Team dynamics, along with the introduction of a robotic system into a surgical procedure, can have comparable influence on the regulatory impact regarding the design and development of all the system's user interfaces—both virtual and physical. The same user research insights that can inform the URRA and regulatory strategy apply to the design and development of the robotic system's user interface.

For instance, for the lead surgeon's user interface, it is important to first understand the likelihood of negative and positive transfer bias in the physical user interface. The same applies to the system's cognitive load requirements, especially if the system is expecting the surgeon to be responsible for what was previously a team effort. Then, there is the accommodation of team dynamics and communication to consider, as was explained earlier. These considerations also apply to the other user interfaces such as end effector access to the anatomy, instrument attachment and draping.

Positive/Negative Transfer Biases

Revisiting the surgeon's user interface example and the associated biases, the mode in which the surgeon executes a procedure manually, the variety of instruments available for use and their specific user interfaces, the instruments' capabilities, kinematics and feedback can result in either a positive or negative transfer bias depending on how the new control interface is designed.

Negative transfer bias can present a possibly dangerous condition and user error that could lead to harm. It can change the rote manner in which a task is performed relative to how it was learned and practiced previously, be it another robotic interface or conventional procedures.

Conversely, the user interface design can bring positive transfer bias by emulating or carefully transitioning from a normal behavior and interface to the new user interface and workflow experience. It is not a viable strategy to depend on training to convert the user's previously learned skills and behavior. A proactive and robust approach would be to do the due diligence in understanding the user's expectation and aspiration and achieve an in-depth understanding of the perceived attributes that afford the intended behavior.

Sean Hägen is founder and director of research and synthesis, BlackHägen.
PHOTOS: All courtesy of BlackHägen.

to view this article online, click here

BACK TO TABLE OF CONTENTS

CHAPTER 4:

Robotic Process Automation Sweeps Across Healthcare Industry

HRISHIKESH KADAM, technical contributor

As businesses step into a world where automation is fast becoming an industry standard, robotic process automation could well be the key to accelerating digital transformation goals.

The robotic-process-automation (RPA) industry is booming worldwide as more companies move toward automated technologies to unlock higher efficiency, accuracy and speed. Businesses will certainly look to leverage intelligent technologies such as RPA to keep employees efficient and engaged, as they enter the second year of the COVID-19 pandemic. While the technology experienced widespread adoption during 2020, it's expected to mature into a solution meant for every department of a business organization.

Gartner estimates that spending on RPA software solutions will grow by nearly $1.5 billion during 2021. The quick and easy nature of RPA in automating manual processes has been a key factor fueling its adoption worldwide. Large enterprises as well as small and medium-sized businesses can leverage the benefits of RPA via faster data processing and lower time spent on repetitive tasks by employees.

Considering the current rate of adoption, it can be estimated that the global robotic process automation market size will be worth over $23 billion by the end of 2026.

RPA Helps Scale COVID-19 Vaccine Distribution

Following the widespread disruption caused by the COVID-19 pandemic, healthcare organizations have started to recognize the value of automation in promoting business continuity. RPA, with its ability to speed up tedious, time-consuming tasks, can help streamline numerous healthcare operations, allowing providers to dedicate more time to patients. RPA solutions also offer increased support to healthcare organizations, enabling them to mitigate the risk of employee burnout as frontline workers get pushed to their limits.

Robotic process automation will account for around 37% of healthcare AI investments over the next few years, according to [KPMG's AI survey] that included about 100 respondents each from healthcare and life-science firms.

As the world continues to battle the pandemic, healthcare and life-science business leaders are banking on RPA's ability to monitor the spread of the virus and help with the development and distribution of vaccines. In March 2021, Notable Health, a leading provider of intelligent automation solutions for healthcare, partnered with the North Kansas City Hospital (NKCH) to accelerate the distribution of COVID-19 vaccines to Missouri based residents.

Notable Health and NKCH have formed a coalition called Operation Safe to scale their vaccination efforts and vaccinate up to 4,000 residents per day. The company is using RPA in combination with fast healthcare interoperability resources (FHIR) to train digital assistants or bots to scan electronic medical records (EMRs) for available appointment schedules for patients meeting eligibility requirements such as age or pre-existing health conditions.

NKCH is leveraging Notable Health's RPA tools to automate and digitize identification, scheduling, outreach and pre-visit intake workflows to vaccinate more patients faster.

Tech Giants at the Forefront of RPA Innovation

In a highly competitive environment, business organizations are exploring ways to be more productive using strategic and innovative thinking. This apparently has accelerated the deployment of RPA technologies among large enterprises as well as small and mid-size enterprises.

Leading tech firms are constantly innovating and adopting new strategies in anticipation of similar trends in the near future. In March, Microsoft India announced plans to roll out its Power Automate Desktop solution for Windows 10 users at no additional cost.

The Power Automate Desktop solution is a new RPA application that allows businesses to automate repetitive manual tasks and emphasize high-value processes to achieve higher efficiency and performance. It's an easy-to-use platform that would allow users to streamline workflows, reduce human errors, minimize maintenance cost and effort and boost scalability while ensuring security.

During the same month, Google Cloud formed a strategic, multi-year alliance with Automation Anywhere, a global RPA leader, to help meet the intelligent automation needs of businesses worldwide. As per terms of the deal, Automation Anywhere will move its cloud-native Automation 360 automation platform to Google Cloud as its primary cloud provider; the company is expected to be the latter's chosen RPA partner.

Together, the two companies aim to build automation-powered products and services tailored for industry-specific applications, primarily within retail, supply chain, financial services, telecommunication, healthcare and life-science sectors.

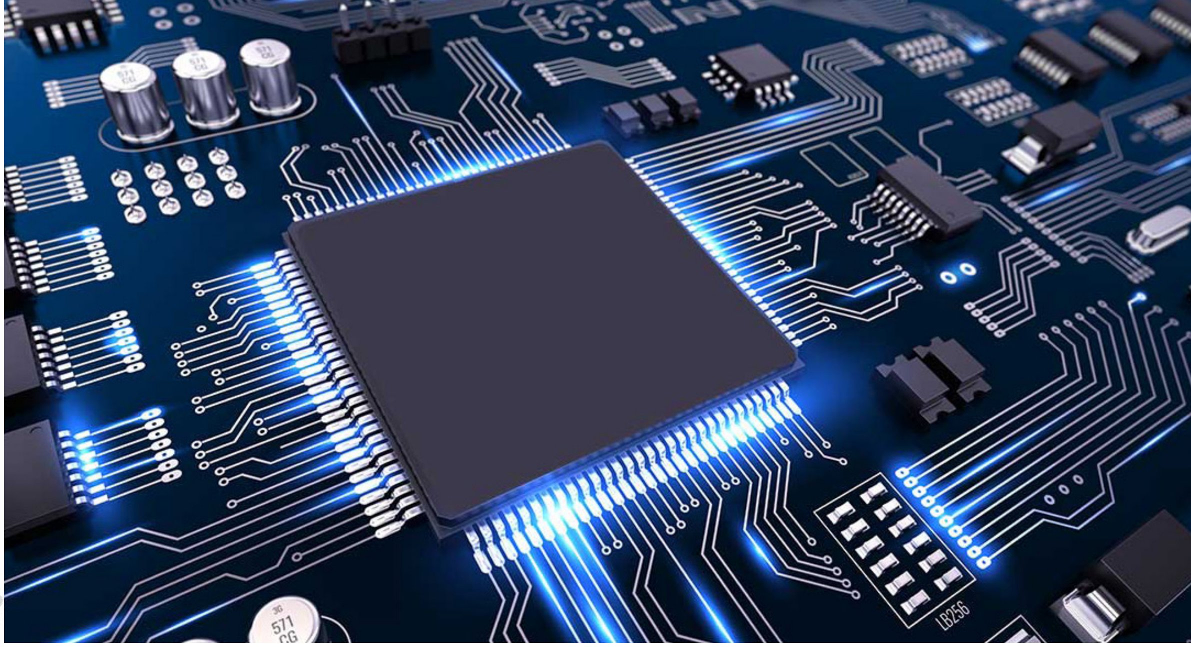
Evolution of RPAaaS and the RPA Industry's Future

Robotic process automation is expected to be one of the biggest, if not the most important, IT trends in the years to come. When deployed, the revolutionary technology can significantly enhance business efficiency and strategy making.

Robotic process automation as a service (RPAaaS) also is anticipated to go mainstream in the coming years. RPAaaS has the potential to help business cut development and deployment costs by utilizing the most reusable and impactful components.

to view this article online, ☞ *click here*

☞ BACK TO TABLE OF CONTENTS



CHAPTER 5:

Build vs Buy of A Three-Axis Motion Controller

PMD ENGINEERING

In this paper we will look at building versus buying a three-axis PCB-based (Printed Circuit Board) positioning motion controller. We will examine the design effort, types of resources needed, and per unit cost for each approach. In addition, we will look at a design alternative that some engineers believe represents the best of both worlds, a hybrid "blend" option that uses off-the-shelf modules to dramatically reduce the engineering needed to create a fully custom motion controller.

INTRODUCTION

In this paper we will look at three different approaches for building an embedded motion control board that can drive typical NEMA 23 or NEMA 34 sized Brushless DC motors or step motors. We will focus on a positioning motion controller used in end applications such as laboratory automation, mobile robotics, test equipment, packaging equipment, and robot arm control.

Broadly speaking, the motion control solution we are building will look something like what is shown in **Figure 1**. The board is fully self-contained and has all connectors and mounting hardware needed to function inside, or

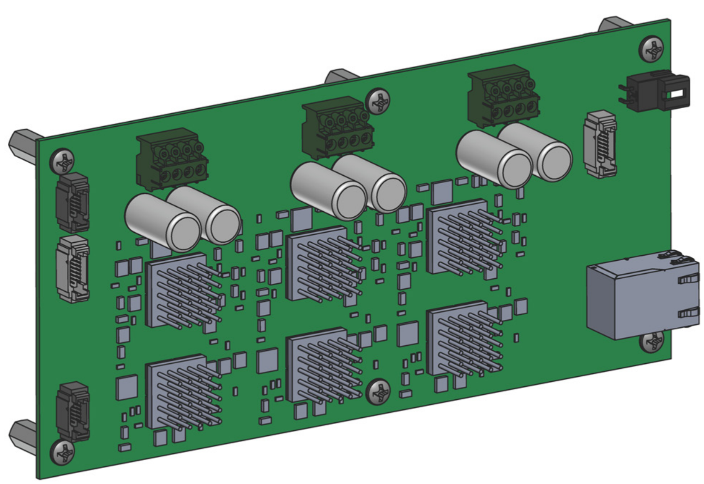


Figure 1: PCB-based positioning control board

on, the machine being controlled. It includes an onboard microprocessor so that user-written application code can execute directly from the control board for the purpose of controlling the machine's operation.

Why Embedded Control?

Figures 2A and **2B** show the two overall approaches that designers might use to construct a motion controller: a centralized controller located remotely from the controlled machine hardware, and an embedded controller located on or inside the machine.

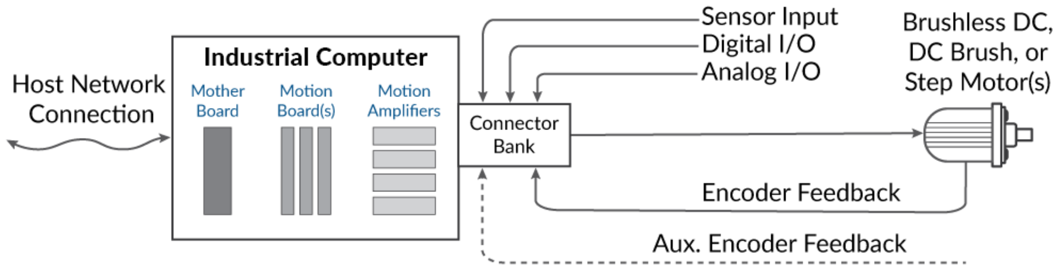


Figure 2A: Centralized motion controller architectures

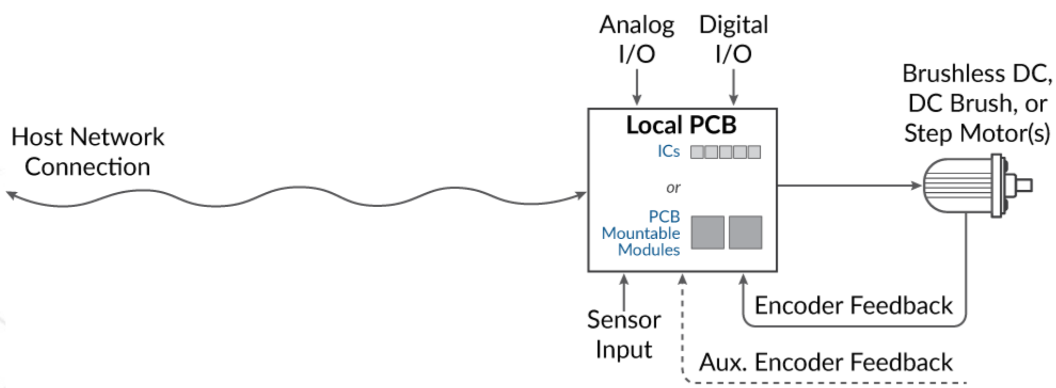


Figure 2B: Embedded motion controller architectures

Given these two different options it is reasonable to ask why build (or buy) an embedded PCB-based controller? The answer boils down to one simple principle: closer is better. Embedded controllers are closer to the sensors and motors that they will control. This means shorter cables which means less noise and faster response times. It is also less expensive because you eliminate the box/rack containing the remote controller and the thick bundle of cables to connect to the machine being controlled.

You may also be interested in: Motion Control Goes Small

Designing A PCB-Based Motion Controller...

To give our "build vs buy" comparison some specific performance metrics here are a few of the target design parameters of the desired PCB-based controller:

- Compact embedded PCB (Printed Circuit Board) designed to mount on the controlled machine hardware
- Three-axis with up to 350W onboard motor amplification per axis
- Controls a mix of Brushless DC and step motors
- Provides inputs and outputs for typical motion control functions such as encoders, home sensors, limit switches, and general purpose I/O
- Powered by a single DCV input (the motor drive voltage)
- Standalone operation with onboard user-written software code execution
- Ability to communicate with an Ethernet host interface

Figure 3 shows the desired control architecture. There are three-axis controllers in total but there is only one connection between the controller PCB and the Ethernet host network. The microprocessor on the controller board interfaces with the host network and parses received commands. It then coordinates the movement of the axes under its control according to this received command. Although we have labelled the axes X, Y, and Z this is arbitrary.

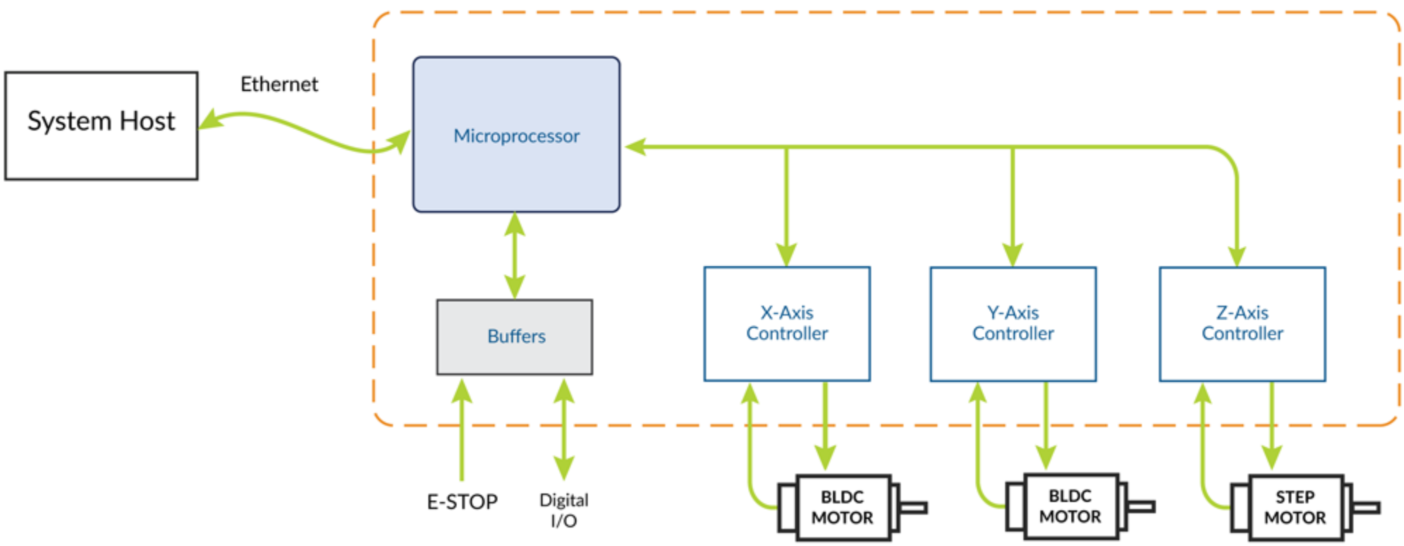


Figure 3: Overall architecture of three-axis controller

It's important to note that the commands sent on the system host are high level, such as "move to position XYZ" or "transfer the test tube from slot A1 to B7." The microprocessor contains user-written application code which knows how to convert these high-level commands into a series of low level motion commands for each axis to execute the sequence.

Alternatively, our embedded PCB-based controller can operate entirely stand alone, taking commands from a human operator using a touch screen or buttons, and using the host network only to report results.

...Is As Easy As 1, 2, 3, (4, 5, 6)

What is entailed in the engineering design of a three-axis motor controller? Every project is a bit different of course, but building our controller will broadly require the following tasks:

- **Select connectors and wiring scheme –** This step is undertaken by an Electrical

Engineer (EE) or system engineer and is a preliminary step to the step of designing the full PCB schematic.

- **Determine the motor amplifier scheme to use –** In the embedded motion control world, a big fork in the road is what to do about the motor amplifier. A fully designed-from-the-ground-up amplifier for Brushless DC or step motors that can drive 350 watts is a serious design project and will include considerations of EMI (Electro Magnetic Interference), current control, and the PCB layout. The alternative is off-the-shelf PCB-mounted amplifiers. These compact modules can drive 500W (or more) and provide high performance current control, amplification, and safety features to drive the motor's windings.
- **Determine whether to build, or buy, a motion control engine IC –** Another big fork in the road for our embedded motion controller task list is whether to use an off-the-shelf motion control IC or instead write motion engine code for a general purpose DSP, microprocessor, or FPGA (Field Programmable Gate Array). Off-the-shelf motion control ICs cost from $15 to $50 per axis and provide a wealth of profiling, servo control, and amplifier control functions. If this path is chosen, the associated software writing task below can be skipped.
- **Design & lay out the PCB schematic –** Utilizing decisions made in the previous two design tasks, this step is undertaken by one or more EEs and consists of researching and selecting IC components to achieve the various electrical functions needed and constructing the PCB's schematic.
- **Write the motion engine software code –** If a fully custom solution for the motion engine IC is to be used it will require software to be written for profile generation, servo loop closer (often PID Proportional Integral Derivative control) and other real time management functions such as E-stop, limit switches, and more.
- **Write the Microprocessor system software code –** This microprocessor communicates with the host network and coordinates the overall machine behavior including motion sequences, user interface, safety monitoring and other functions. This could be the same microprocessor as the motion engine IC, but using that approach adds complexity, requiring a high speed multi-tasking real time operating system. With the present day cost and size of microprocessors most three-axis motion controllers will instead separate the main PCB microprocessor from the motion control microprocessor(s).
- **Write the microprocessor application software code –** Having a functioning operating system and software drivers for connecting to the PCB's motion and peripheral ICs is not the same as having the final machine control solution. So, we are going to break the overall microprocessor function into three parts, of which the system software code detailed above is the first, and this task is the second.
- **Develop the motion control profiles & parameters –** This is the third and final piece of the software development process and consists of working with the actual machine hardware to optimize the motion profiles, servo settings, and other motion control settings. This effort typically involves trial and error and use of a trace facility so that trajectory overshoot and undershoot can be minimized among other optimizations.

OK, with the major project design steps detailed let's look at each of three options; Build, Buy, and Blend to see how they come out in an analysis of design cost, time to market, and per unit cost. We will start with Option 1 which is the Build option.

OPTION 1: BUILD

The table below gives very rough estimates, in engineering design man months, for each project development step listed above when choosing the build option.

	Design Effort In Man Months	Associated Additional Expenses
Connectors & wiring scheme	3	$1,000 or less for samples, prototyping cost
Determine motor amplifiers scheme	2	$250. May include the purchase of off-the-shelf amplifier modules or specialized ICs to benchmark performance
Determine whether to build or buy motion control engine IC	1	$250. May include the purchase of a motion control IC engine IC development kit to benchmark performance
PCB schematic development	12-18*	$50,000 for prototypes cost assuming three spins of the board
Motion engine IC software development	0-12**	$5,000 for developer kits & software tools cost
Host micro system software development	6	$5,000 for developer kits & software tools cost
Application software development	8-12	$0
Motion parameter development	2-4	$0
TOTAL	**34 - 58 months**	**$61,500**

* Higher if amplifier circuit must be developed, lower if off-the-shelf amplifier is used.
** If off-the-shelf motion engine IC is used development cost for this category will be zero.

The timeframes above are estimates and should not be taken too literally. Some teams will get it done faster, and some slower based on their experience and the project complexity.

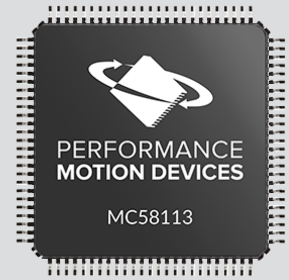
Build Option Summary

The design effort and associated costs for building a three-axis motion controller PCB are significant. You will need the right mix of engineering skills but even with this, a 2- or 3-person team will still typically take 18 months to complete the controller. Note that this can be reduced if off-the-shelf motion control ICs and amplifiers are used in the design.

The key benefits of the build approach are that the resultant controller is entirely tailored to the application in terms of connectors and form factor, and that it has the lowest cost on a per unit basis. Assuming a production run of 1,000 controllers per year, an achievable complete PCB cost is $225 to $450 which translates to a per axis cost of $75 to $150.

OFF-THE-SHELF COMPONENTS TO SPEED YOUR PCB DESIGN

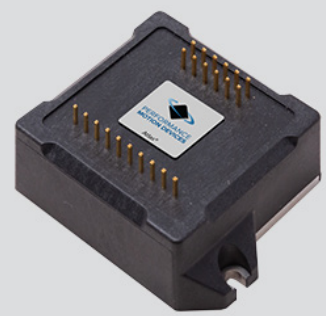
Motion Control ICs



Off-the-shelf motion control ICs such as the Magellan® MC58113 from PMD provide a wealth of motion control functions including profile generation, servo loop closure, and real time signal management. They provide direct input of quadrature encoder signals and direct output control signals for amplifiers. Many even provide advanced features like FOC (Field Oriented Control).

Motion IC Developer Kits, such as the MC58113 DK, allow software developers to start writing motion control sequences and spinning motors almost immediately. So, both the electrical and the software engineers can work in parallel - greatly lowering the project development time.

PCB-mountable Amplifiers



Off-the-shelf motor control amplifiers such as the Atlas® Digital Amplifier from PMD provide high power levels (up to 500W) and high performance amplification for step, Brushless DC, and DC Brush motors. These compact devices accept a digital torque command and provide all the functions needed to commutate the motor and precisely control the current through each motor winding.

Atlas Digital Amplifiers provide FOC (Field Oriented Control), performance tracing, and voltage, current, and temperature safety monitoring.

OPTION 2: BUY

For this approach we will select a ready-to-go PCB-based motion controller. There are a number of board-based products to choose from, but for the purposes of this analysis we'll use PMD's Prodigy/CME™ Machine Controller.

The configuration shown in the image above is a four-axis controller but can be purchased in a three-axis configuration as well. The Prodigy/CME™ Machine Controller is a representative motion controller board that meets all of our application requirements. In addition, the usual features such as profile generation, servo position control, and encoder input, this product supports on-board amplifiers, user downloadable code written with C language libraries and provides Ethernet as well as other host connection options.

*So how did the buy option affect the project timeline? (*The table on the next page shows this)

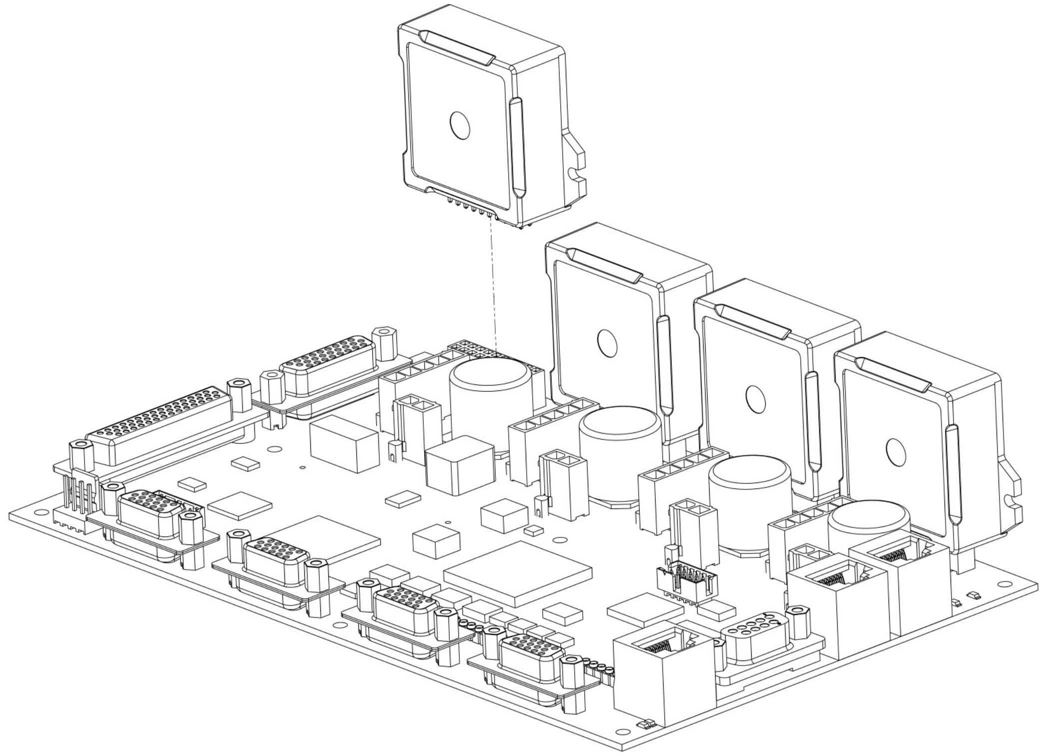


Figure 4: Prodigy®/CME Machine Controller from PMD

	Design Effort In Man Months	Associated Additional Expenses
Connectors & wiring scheme	0	$0
Determine motor amplifiers scheme	0	$0
Determine whether to build or buy motion control engine IC	0	$0
PCB schematic development	0	$0
Motion engine IC software development	0	$0
Host micro system software development	0	$0
Application software development	8-12	$0
Motion parameter development	2-4	$0
TOTAL	**10-16 man-months**	**$0**

Buy Option Summary

The appeal of the buy option centers around two factors. The first is that the engineering effort costs much less. The second is that the design effort takes less time which should mean the machine being controlled can get to market faster.

The tradeoffs are also twofold, however. The first is that the resultant controller costs more per axis. Again assuming 1,000 controller per year, the overall controller costs $900 - $1,200 and each controlled axis typically costs $300 - $400. That's two to three times the per axis cost of the board created using the build option. The second tradeoff is that the PCB is not tailored to the application. The form factor is pre-determined, as are the connector types and pin arrangements.

OPTION 3: BLEND

There is an interesting new approach toward building fully custom motion control boards that isn't quite a build option, nor a buy option. This hybrid "blend" approach uses miniaturized motion control modules that are PCB-mounted yet contain fully functional motion controllers. These modules provide the ability to download code so that they can execute user-written application software.

These products almost always come in a single axis configuration and can drive up to a kilowatt and sometimes more of power. **Figure 5** shows such a product, in this case the ION®/CME N-Series Digital Drive from PMD.

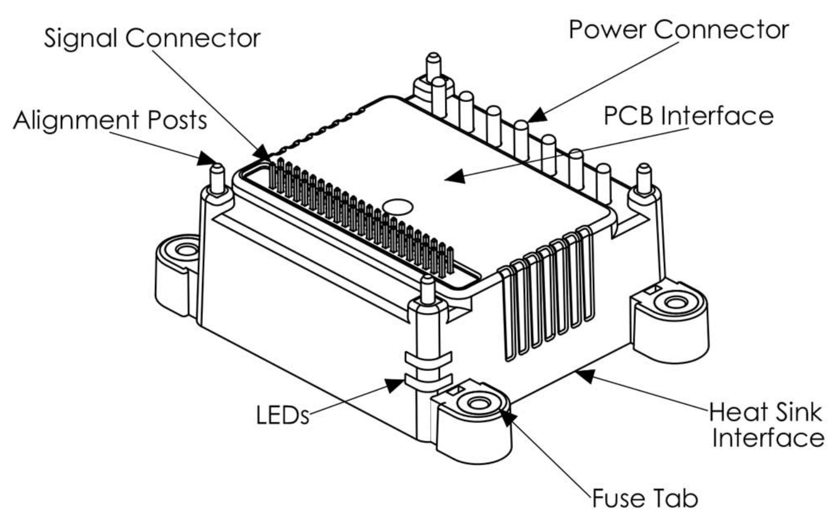


Figure 5: ION®/ CME N-Series Digital Drive

Just like the 100% buy option, these modules provide a wealth of ready-to-go motion control functions such as profiling, FOC, and current monitoring safety features. But unlike the 100% buy option they are mounted on a user designed, application-specific interconnect board. An example of such a board is shown in both top view and bottom view rendering in **Figure 6A and 6B**. Note that in addition to the modules themselves there are just connectors and a few capacitors on the PCB.

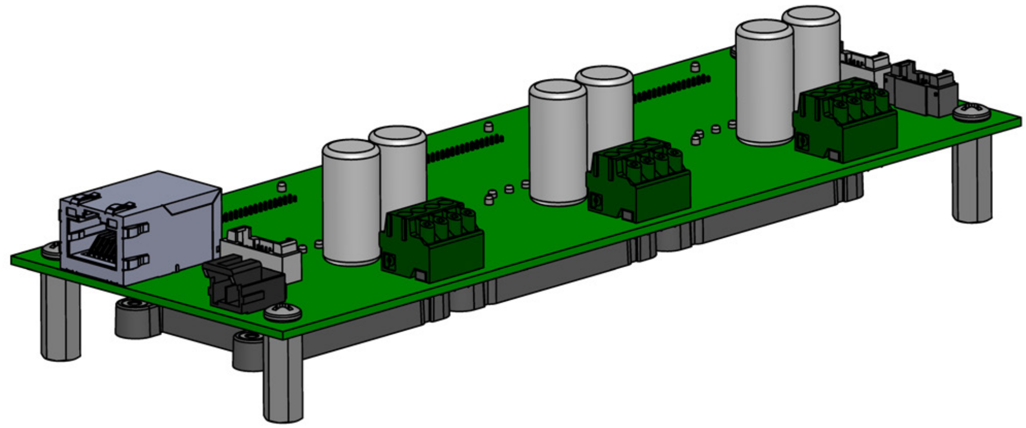


Figure 6A: ION/CME N-Series Digital Drive top view

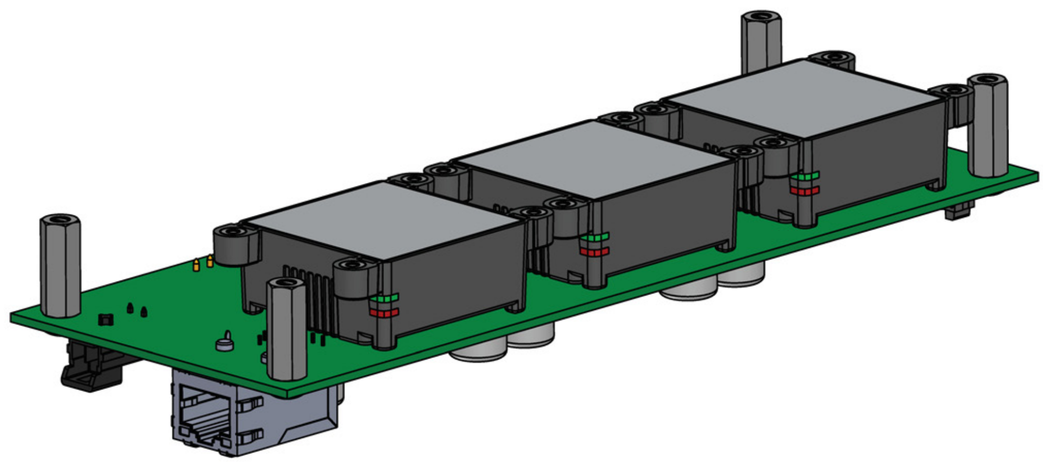


Figure 6B: ION/CME N-Series Digital Drive bottom view

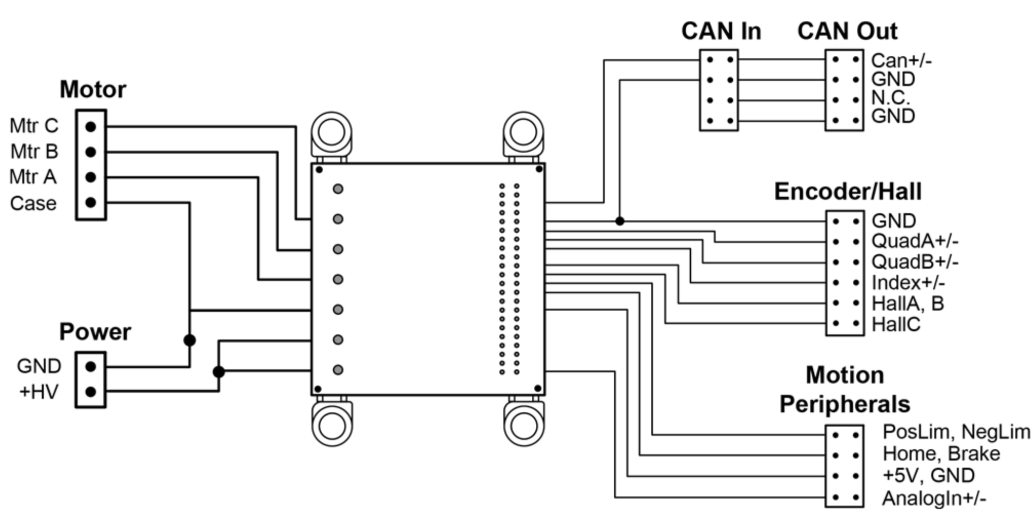


Figure 7: Simplified wiring diagram for a “blend” approach motion controller

The effort involved with creating a board using motion control modules is not so much creating a schematic, it's more like creating a wiring diagram. A simplified version of such a wiring diagram is shown in **Figure 7**. Also important is that the resulting board is not 10, 12, or 14 layers (as would be typical for the full “build” option board), but usually just 4 layers. The engineering skillset level required is therefore significantly lower because all the heavy lifting such as motor amplification and signal pro-

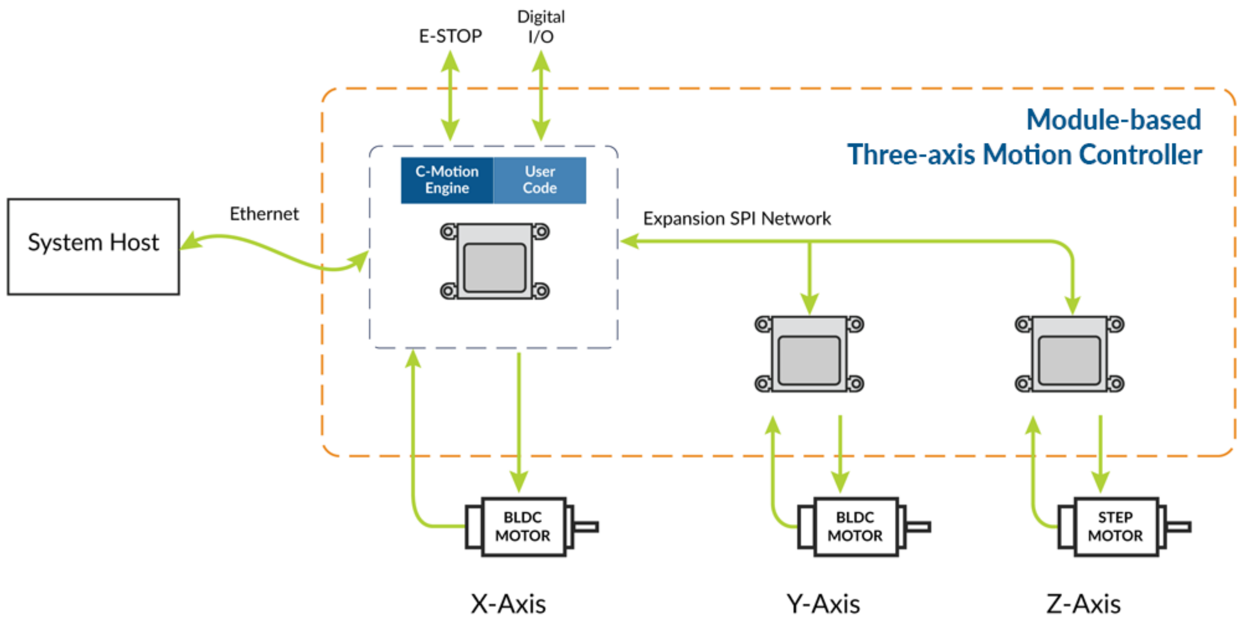


Figure 8: Control architecture for three-axis module-based controller PCB

cessing is done inside the modules.

Can these modules support the desired motion controller architecture seen at the beginning of the article? Yes, they can. Our application specification required the PCB to receive commands and provide reporting through an Ethernet connection. And we wanted to write PCB-resident application software that would interpret these commands and drive the three motor axes accordingly.

This can be accomplished using PCB-mountable modules as shown in **Figure 8**. One of the modules holds the user-application code as well as controlling one axis and then in turn commands the other two axes.

So how does this new option affect the project timeline? The table below shows this:

	Design Effort In Man Months	Associated Additional Expenses
Connectors & wiring scheme	3	$1,000 or less for samples, prototyping cost
Determine motor amplifiers scheme	0	$0
Determine whether to build or buy motion control engine IC	0	$0
PCB schematic development	3	$5,000 for prototypes cost assuming two spins of the board

(table continued on next page)

Motion engine IC software development	0	$0
Host micro system software development	0	$0
Application software development	8-12	$0
Motion parameter development	2-4	$0
TOTAL	**16 - 22 man-months**	**$6,000**

With the blend option there are still some EE tasks, but they are much, much simpler than building a full motion control board. After that, the focus is on the application software development and the motion parameter development, which is unchanged from the build or buy options.

Blend Option Summary

The appeal of the blend option is that it lets you create a fully custom PCB-based controller with form factor and connectors of your own choosing, yet with only a fraction of the engineering effort needed compared to the traditional build option. This is achieved by the use of fully functional motion modules that reduce the PCB-design task to one of creating a wiring interconnect board.

From a cost perspective, the blend option has the same to slightly lower per unit cost than the buy option. So assuming a production rate of 1,000 three-axis controllers per year, this is $250 - $350 per controlled axis.

Summary

The table below summarizes the engineering effort, design out of pocket expenses, and per axis cost of our three-axis motion controller PCB using the build, buy, and blend options.

Option	Engineering Time in Man Months	Additional Expenses	Per Axis Cost
Build	39-57 man-months	$61,500	$75-150
Buy	10-16 man-months	$0	$300-400
Blend	16-22 man-months	$6,000	$250 - $350

The above figures are estimates for the purpose of comparison only. Actual results will depend on many factors including the experience of the design team, whether some previous design work can be leveraged for the new design, and the complexity of the machine being controlled.

We hope you found this comparison of engineering effort and per unit cost for a three-axis PCB-based controller helpful. Understanding the tradeoffs between the build, buy, and blend options will help you make the right choice in your next motion controller design.

PMD Products Featured in This Article

PMD has been producing ICs that provide advanced motion control of Brushless DC, DC Brush and step motors for more than thirty years. Since that time, we have also embedded these ICs into plug and play modules and boards. While different in packaging, all PMD products are controlled by C-Motion, our easy to use motion control language and are ideal for use in medical, laboratory, semiconductor, robotic, and industrial motion control applications.

MC58113 Motion Control IC

Performance Motion Devices' **MC58113 Motion Control IC** is ideal for controlling Brushless DC, DC Brush, and step motors, providing high-performance motion control features including field oriented control and closed loop stepper operation. In addition, the MC58113 IC provides profile generation, servo-loop closure, commutation, current control, and direct PWM (Pulse Width Modulation) motor output command to drive MOSFET switching amplifiers. It is ideally suited for a wide range of applications including scientific, mobile, medical, robotic, and automation applications. **>>LEARN MORE**

ION/CME N-Series Digital Drives

N-Series ION Digital Drives combine a single axis Magellan IC and a high performance digital amplifier into an ultra-compact PCB-mountable package. In addition to advanced servo and step motor control, N-Series IONs provide S-curve point to point profiling, field oriented control, downloadable user code, general purpose digital and analog I/O, and much more. With these all-in-one devices building a custom controller board is a snap, requiring you to create just a simple 2 or 4-layer interconnect board. **>>LEARN MORE**

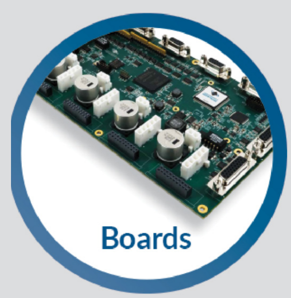
Atlas Digital Amplifiers

Atlas Digital Amplifiers are compact single-axis amplifiers that provide high-performance torque control for Brushless DC, DC Brush, and Step motors. Atlas amplifiers come in both a vertical and horizontal mounting configuration and are available in three power ranges: 75W, 250W, and 500W. They are used for direct control of motor torque or in conjunction with higher-level motion controllers providing position or velocity control functions.

>>LEARN MORE

Prodigy/CME Machine Controller

PMD's Prodigy®/CME Machine-Controller boards provide high-performance motion control for medical, scientific, automation, industrial, and robotic applications. These versatile all-in-one controllers support quadrature, Sin/Cos, and SSI encoder formats, and are available in 1, 2, 3, and 4-axis configurations, with support for DC Brush, Brushless DC, and step motors. Prodigy/CME Machine-Controllers have on-board Atlas amplifiers that eliminate the need for external amplifiers. **>>LEARN MORE**



Visit us at www.pmdcorp.com or call +1-978-266-1210 to learn more.

BACK TO TABLE OF CONTENTS