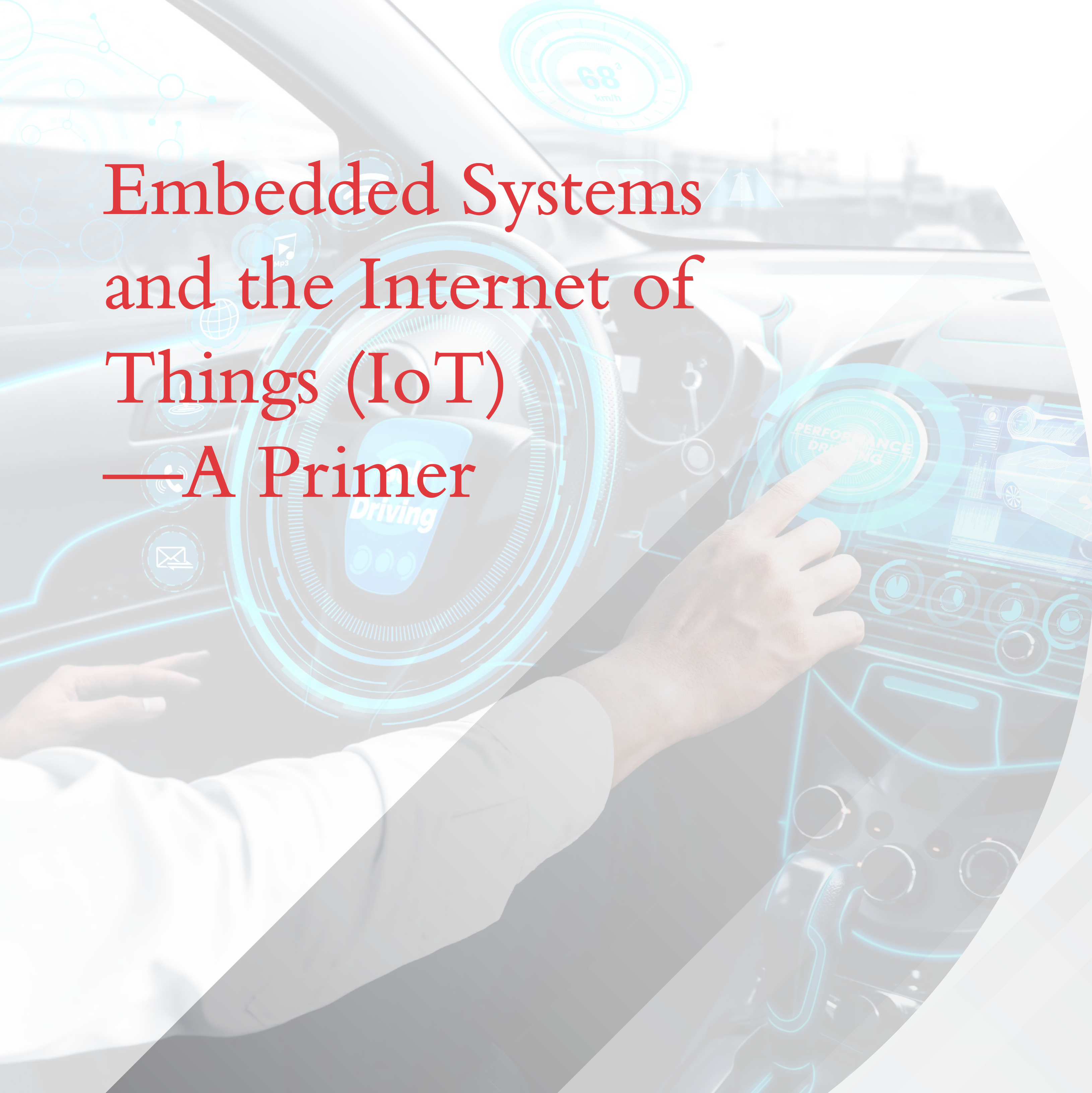
UNIVERSITY OF MARYLAND

Future-Proof Your Programming:

A Guide to Career Opportunities in Cyber-Physical Systems Engineering

Table of Contents



1

Embedded Systems and the Internet of Things (IoT)—A Primer

> Digital technology, pervasively, is getting embedded in every place: every thing, every person, every walk of life is being fundamentally shaped by digital technology — it is happening in our homes, our work, our places of entertainment. It's amazing to think of a world as a computer. I think that's the right metaphor for us as we go forward.
>
> – Satya Nadella, CEO of Microsoft

The world can be separated into two categories: those who tinker and those who don't.

At their core, tinkerers, dabblers, and DIYers are curious. It's not enough that things work; they have to know *how things work*.

Back before the advent of the internet, a tinkerer might have disassembled a toaster just to see the mechanics inside. These days, many who dabble find themselves engrossed in programming projects—using a Raspberry Pi or Arduino to control a robot, build a web server, set up a home automation system, create a time-lapse video, or even broadcast an FM radio station.

Sound like you? Then it sounds like you may have a bright future in Cyber-Physical Systems Engineering (CPSE)!

Many don't realize that this type of DIY is the perfect foundation for a career in CPSE. And while this is a relatively new field, it's a rapidly growing one. We need more embedded systems experts, and CPSE programs, like the one at the University of Maryland, are few and far between.

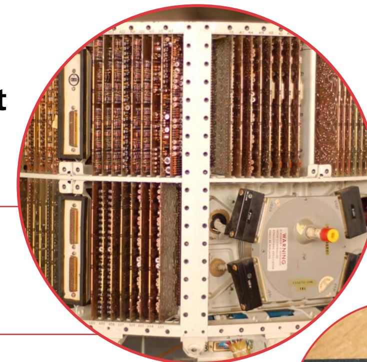
So, let's explore the embedded systems field and the career opportunities therein.

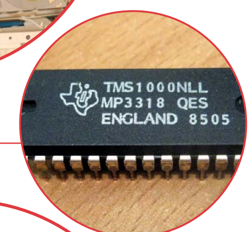
Being at the forefront of emerging technologies is as easy as scrolling to the next page.

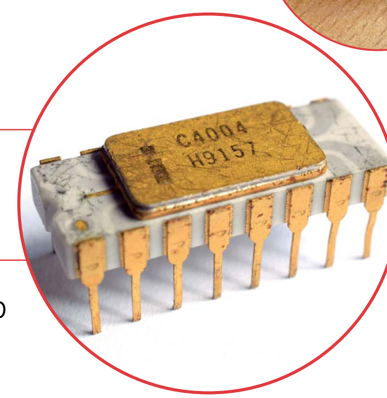
The Importance of Embedded Systems in Daily Life

Let's travel back in time—all the way back to 1961.

Scientist and engineer Charles Stark Draper developed the first embedded system: an integrated circuit designed to decrease the size and weight of the Apollo Guidance Computer. Yes—*that* Apollo. The guidance computer was installed on the Apollo Command Module and Lunar Module. And without Draper's embedded system, the moon landings might not have been possible.







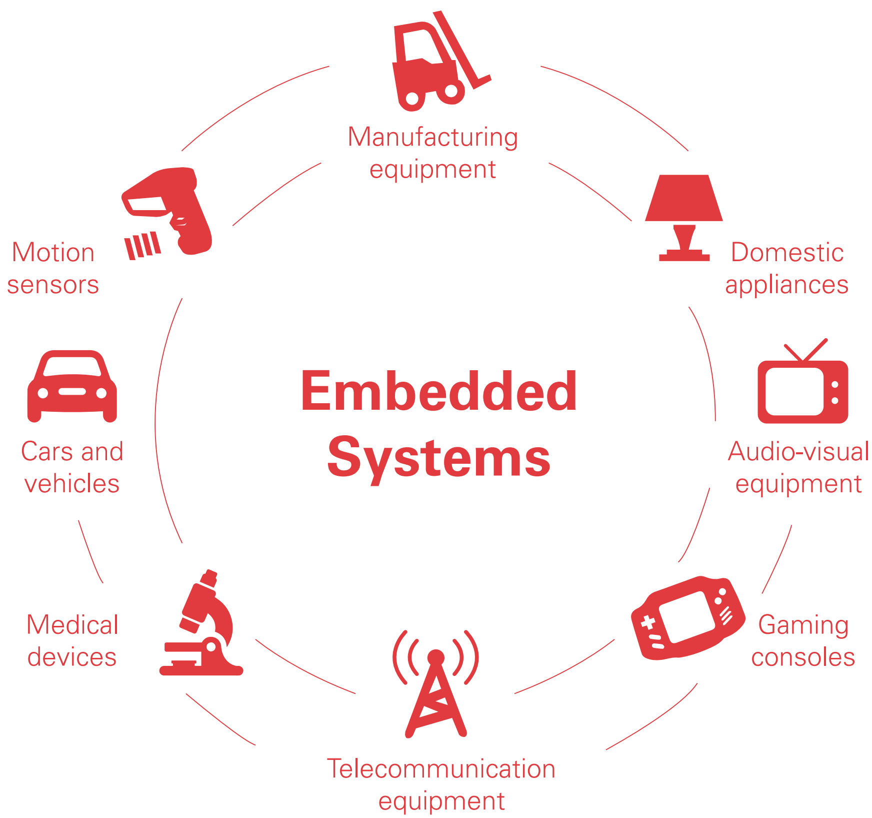
1965	Autonetics (now a part of Boeing) creates what is widely recognized as the **first mass-produced embedded system**: the D-17B. It's used in the Minuteman I missile guidance system.
1968	Volkswagen releases the **first embedded system for a vehicle**: a microprocessor that controls its electronic fuel injection system.
1971	Texas Instruments develops the **first microcontroller**, the TMS1000 series.
1971	Intel releases the **first commercially available microprocessor unit**, the Intel 4004. It was designed for use in calculators and small electronics, but the microprocessor still requires support chips and external memory.
1978	The National Engineering Manufacturers Association creates a new standard for programmable microcontrollers, improving the embedded system design.
Early 1980s	Memory, input, and output system components are integrated into the same chip as the processor, forming an entirely new kind of microcontroller.

These days, our microprocessors look more like this:

And our microcontrollers look more like this:

And both are crucial to the embedded systems that feature in nearly every aspect of our everyday lives (either today, or in the future!). To name just a few:

Vending machines	Remote controls	Self-driving vehicles
Cameras	Coffee makers	Robotic systems
Washing machines	Air bags	Telemedicine equipment
Drones	Super smart watches	Advanced manufacturing



So, What Are Embedded Systems?

Embedded systems are the electronic chips that are incorporated into smart devices to provide intelligence and awareness capabilities. Your smartphone is a collection of embedded devices, as are smart homes, smart cars, drones, and robots. Embedded devices have extraordinary functionality, like the ability to sense the environment around them, make simple decisions, and even communicate with one another.

The embedded systems functions are programmed into them, and this is all because hardware and software work together.

Common Features of Embedded Systems

- Small devices incorporated into larger gadgets
- Designed to perform specifically programmed, repeated functions on certain single-purpose devices
- Should perform their functions quickly, often between a prescribed time frame, and be synchronous with other devices in the system
- Based on microprocessors, microcontrollers, and field programmable gate arrays (FPGA's)
- Have a variety of communication systems, such as broadband
- Can use special operating systems (OS)—for example, real-time operating systems (RTOS)—but can also work without an OS

Microcontroller vs Microprocessor vs FPGA vs ASIC

A **microcontroller** acts as the heart of an embedded system. It's a single integrated circuit, typically used for a specific purpose and designed to implement specific tasks. As a controlling device, it gathers information, processes that information, and carries out one of several tasks based on the information received. Microcontrollers constitute the electronic brains of products and devices like automobile engine control systems, appliances, digital telephones, cameras, printers, and computers.

A **microprocessor**, on the other hand, is more like the heart of an entire computer system. A Raspberry Pi, for example, contains a microprocessor, and it's an electronic part that's used by a computer to do its work. A microprocessor contains a central processing unit, memory and other peripheral components in a single integrated circuit chip that contains millions of very small components, mostly transistors, that are wired together.

A microprocessor is the most complex electronic circuitry today, capable of running multiple tasks at the same time. You've probably played a video on YouTube while typing up a paper—that's due to microprocessors. All microprocessors are loaded by a program called the Operating System. You may be familiar with some of them like Windows, MacOS or Linux, but there are others, like real-time operating systems (RTOS) used in many embedded systems applications.

Microprocessors are used in regular computers, smart phones and other smart devices that can be programmed with many different functions. Some of these functions require the assistance of microcontrollers to take care of fast responses from the sensors. A camera on your smartphone, for example, has a dedicated microcontroller that adjusts the camera parameters based on light and movement to get clear images in the dark and avoid image blurring from camera jitter. The latter feature is often advertised as image stabilization. Without help from the embedded microcontrollers, the microprocessors would not have the capability, speed, and power to implement high performance features in modern devices.

An **FPGA** (Field Programmable Gate Array) chip is very similar to a microcontroller, but it's more flexible and easier to implement fast routines. It also has the advantage that it can be reconfigured readily. In its simplest form, an FPGA is an integrated circuit with an array of programmable logic gates or blocks which are the basic building blocks of digital circuits. Unlike microprocessors, which have predefined components and applications, the logic on an FPGA chip can be configured to perform both microcontroller and microprocessor functions based on a desired application or function. Modern devices are often developed first using FPGAs prior to using microprocessors to save on cost.

An **ASIC** (Application Specific Integrated Circuit) chip is a less flexible version of an FPGA chip in that it is designed and optimized to perform a specific task or application such as scaling and rotating images on a TV screen, handling cash deposits, withdrawals, and money transfers on an ATM machine, and more!

Despite the complexity and almost magical capabilities of modern devices, there are really only these few devices that are fundamental building blocks that comprise them.

The Internet of Things Explained

To make it simple, the Internet of Things is a giant interconnected network of things collecting and sharing data about the way they're used and about the environment around them. IoT is made possible by embedded systems.

> **The Internet of Things is not a concept; it is a network, the true technology-enabled network of all networks.**
>
> – Edewede Oriwoh, IoT researcher

These embedded devices and objects—from wearable fitness devices to smart microwaves—connect to an Internet of Things platform. From there, IoT platforms integrate data from all these different devices, analyze it, and share the most valuable information with applications built to address specific needs.

What does this look like in action?

Let's say an industrial manufacturer would like to prevent delays in production due to malfunctioning equipment. This manufacturer can use the Internet of Things and embedded systems technology to:

- Deploy sensors in all of its manufacturing equipment which monitor their performance.
- Analyze the data from the sensor readings to predict whether equipment failure is about to happen and perform preventative maintenance on the equipment.
- Use the sensor data to optimize the manufacturing process.

As embedded technology improves, we'll start seeing scenarios like this:

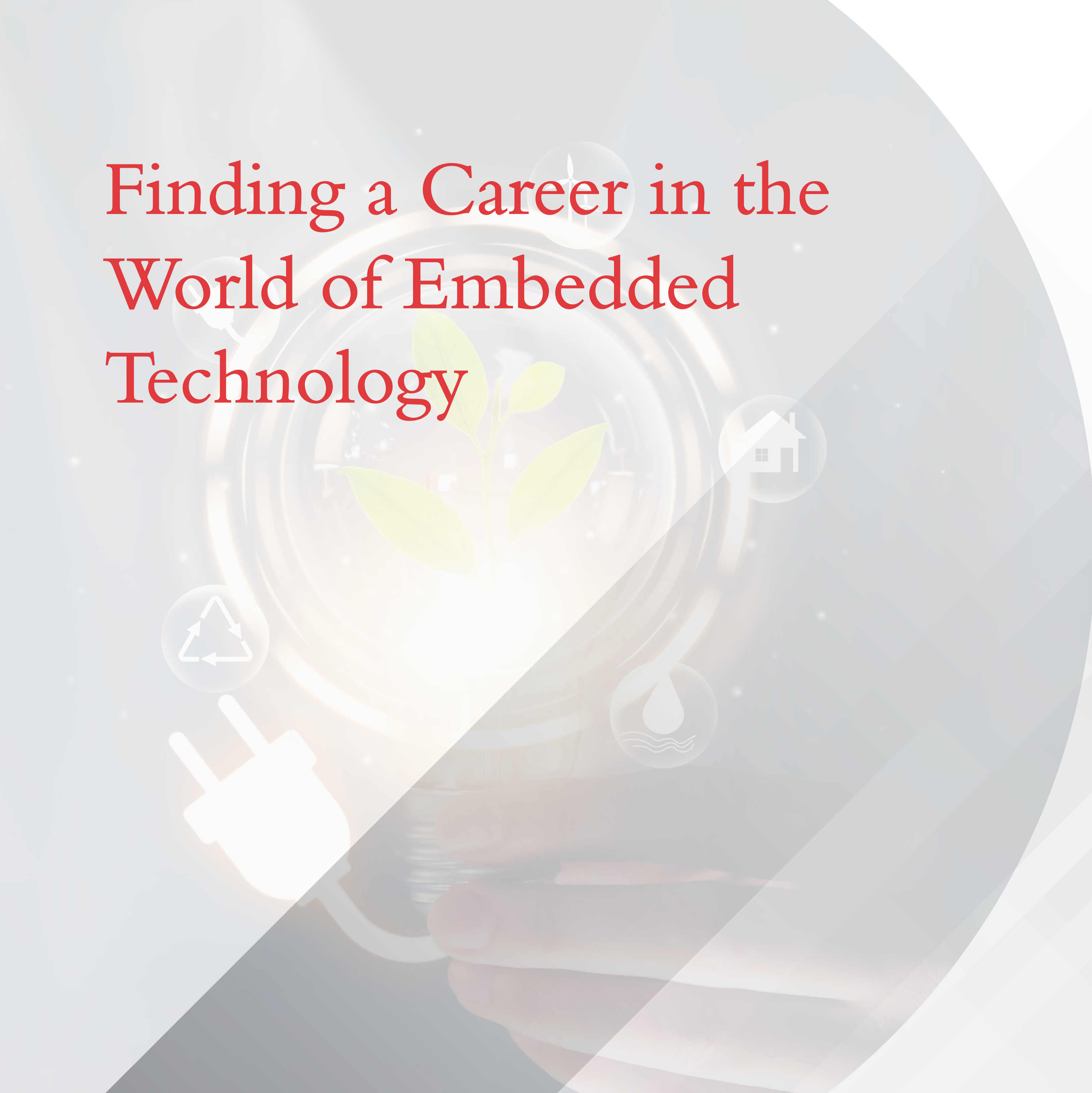
Every morning, your coffee is programmed to start brewing at 5:55 AM, so it's ready for you by the time your alarm goes off at 6.

On a normal day, you commute to work by train—the 7 AM express. But, of course, this is no normal day. The express is experiencing some uncharacteristic delays, according to the scheduling app on your phone, and in order for you to reach work on time, you'll need to catch the local train at 6:45 AM.

Your devices all speak to one another, download this information, and adapt. And so today, your alarm goes off fifteen minutes earlier on its own, and your coffee is still brewed, having corrected for the change in this morning's routine.

But the industry needs more people who can maintain, develop, and innovate in the embedded systems field. Let's take a look at what those careers might look like.

Finding a Career in the World of Embedded Technology



2

> There was a time when every household, town, farm or village had its own water well. Today, shared public utilities give us access to clean water by simply turning on the tap; cloud computing works in a similar fashion. Just like water from the tap in your kitchen, cloud computing services can be turned on or off quickly as needed. Like at the water company, there is a team of dedicated professionals making sure the service provided is safe, secure and available on a 24/7 basis. When the tap isn't on, not only are you saving water, but you aren't paying for resources you don't currently need.
>
> – Vivek Kundra, first CIO of the United States

Before we even jump into roles, titles, salaries, and all the other specifics of potential embedded systems jobs, let's talk about the focus areas. Because let's face it—you may not know exactly what you want to do with embedded systems, but you may know *where* you want to do it. These focus areas are just a handful of examples of the applications of how embedded systems and IoT are used in everyday life, and keep in mind that we are in the midst of a technological revolution. The 5G cellular network promises to significantly speed up the connection that embedded systems and IoT rely on, which will improve how these applications function.

IoT and Embedded Systems Applications

The Automobile Industry

Let's talk cars.

Jam-packed with embedded systems, cars—and their drivers—have been benefiting from this technology for years. Many of these systems use sensors and were designed to make the car movement itself safer, like automatic transmission, anti-lock braking systems, cruise control, and blindspot detection. But embedded systems can have ecological benefits, too: emission control technology that's designed to reduce air pollution.

Innovators are currently working on prototypes for even smarter cars—cars that are able to, after a crash, provide real-time location of the vehicle and alert the nearest police station or first responders. And in the autonomous vehicle world, embedded systems engineers are developing driverless cars that use sensors and communicate with other cars on the road. In the future, these cars could drastically reduce the number of accidents, injuries, and deaths by taking the human element out of driving.

Environmental Monitoring

What if bridges told you when they needed structural maintenance? Or, what if we could predict and monitor environmental parameters, allowing us to minimize—or even *eliminate*—environmental pollution?

Practical applications of environmental monitoring include:

- management of hazardous and radioactive waste
- protection of public water supplies
- weather forecasting
- air monitoring and identification and analysis of pollution sources that affect air quality
- mapping, protection, and management of natural resources
- allocation of resources for land planning and economic development
- soil monitoring
- protection of endangered species
- tracking and examination of global climate change

Exploring the ways embedded systems can help us keep our environment healthy is a worthy cause. And for those who are interested in encouraging resource efficiency and management in the use of embedded systems (the creation and use), there's work in that, too.

While using embedded systems has certainly improved the way we manage energy—making machinery and devices more efficient—there's still a long way to go.

Smart Agriculture and Farming

Farmers face a neverending onslaught of unique business challenges. These days, farmers can't repair their machinery the way they used to (i.e., themselves) due to software digital rights management and equipment design decisions—that tend to work in favor of the equipment manufacturers (and their pockets).

There are also strained tensions between suppliers and farmers. Some farmers are beholden to exclusive producers of seeds for some crops, yet cross-pollination with patented, genetically modified plants can be unavoidable.

How do embedded systems help?

Embedded systems alert farmers via SMS of changing conditions and needed interventions. Soil monitoring informs them of soil acidity and what—if any—additions are needed. Farmers can remotely manage accessible inputs, such as activating irrigation systems. And embedded systems allow them to collect data and push it to the cloud for additional processing, analysis, and data.

Smart Cities and Homes

Modern homes are already filled with consumer goods that double as embedded systems: smartphones, TVs, music systems, digital cameras, gaming consoles, and other entertainment-adjacent devices.

And while it may come as no surprise that a vacuum-cleaning robot is an embedded system, many don't realize fridges, coffee machines, and air conditioners are, too. All of these are great examples of embedded software in action.

But this technology is always progressing. As soon as devices became internet-enabled, the concept of a smart home emerged. There are security camera feeds that you can check remotely and program to update you whenever someone comes to your house. There are thermostats that adjust the temperature inside based on the weather. And those vacuum cleaning robots? Well, they can let you know when they get stuck on the stairs.

Outside the home, our cities have become smart, too. Embedded systems make up the foundation for traffic control systems, smart parking spots, surveillance systems, interactive kiosks, pollution monitoring solutions, and even some community services.

Commerce and Logistics

Technical innovations have changed the way we approach business and manufacturing. In the past decade, we've witnessed the rise of smart manufacturing, where factories and supply chains incorporate robotics, artificial intelligence (AI), Big Data, and IoT into their production processes.

For example, embedded systems have revolutionized fleet management and logistics. Control centers can manage more effectively by creating efficient driving routes, monitoring a vehicle's health status, and updating and scheduling in real-time. When it comes to quality control, factories now incorporate product-quality monitoring, enabling machines to identify defects the human eye might not see. Big Data helps manufacturers avoid bottlenecks in the supply chain with seamless inventory management tools that focus on having the right products available in the right quantity at the right time.

And finally—machine monitoring means that stockpiling replacement parts is a thing of the past. Predictive analytics is used to monitor the condition of all operating equipment in a factory. If the equipment is underperforming, an alert is sent to a plant manager. They can replace the part *before* the process breaks down.

Industry 4.0

Industry 4.0 signifies the fourth revolution that has occurred in manufacturing. The first industrial revolution was marked by water and steam power, and the second was by mass production, assembly lines, and electricity. But the fourth builds on the third—the adoption of computers and automation—and enhances it with smart and autonomous systems fueled by data and machine learning.

This digital transformation also means that we need more cybersecurity measures than ever before. Devices can be communicating personal, confidential information that, as we've seen in recent years, can fall into the wrong hands. But embedded systems are making their way into critical infrastructure, which could ultimately prove more disastrous—controlling and monitoring drinking water, for example. There are approximately 54,000 drinking water systems in the US, and many of them don't have someone watching IT operations 24/7. It's a situation ripe for exploitation.

Hardware and software have unique vulnerabilities that require specific approaches to cybersecurity. With technology progressing so quickly, security needs to follow suit. Embedded systems need to be protected against known and unknown—or yet-to-be-exploited—vulnerabilities.

Some embedded systems examples in Industry 4.0:

- **Robots** are becoming increasingly more affordable for warehouses, picking products and getting them ready to ship. Often, they're designed to perform high-precision tasks in work conditions too dangerous for humans. Their integrated embedded system connects to different subsystems, employing sensors, actuators, and software to perceive the environment around them and execute the intended output safely.
- **3D printing**, also known as additive manufacturing, used to be primarily for prototyping and was initially employing polymer plastics. Now, 3D printers are using a broader range of materials, including metals and cutting-edge nanomaterials like graphene. As the tech improves and the costs decline, 3-D printed

electronics could be integrated directly into the mechanical portions or enclosures of equipment. This industry is growing rapidly, with expectations of an aggregated market size that exceeds $35 billion in 2020.

- **Cognitive computing** uses a blend of artificial intelligence, machine learning, natural language processing, sentiment analysis, neural networks, and contextual awareness. It's an advanced system that learns at scale and mimics the human thought process with the goal being to help assist humans in decision-making. Taking the healthcare industry as an example, cognitive systems speed up the data analysis process; they help physicians collect and analyze data from various sources (medical reports, medical journals, diagnostic tools, etc.) to assist physicians in providing a data-backed treatment recommendation that benefits both physician and patient.

Healthcare

Hospitals and other healthcare facilities have used embedded systems in medical devices for years. What technology has done, however, is make it possible for these devices to frequently monitor at *home*. This can not only be seen in wearable devices and fitness trackers, but in devices like pacemakers, defibrillators, and ultrasound scanners, too. Doctors and patients can more easily monitor and manage conditions. And, as all this health data gets collected, stored, and analyzed, it has the added bonus of aiding population health efforts.

Embedded systems are used in diagnosis and imaging, as well. CT scanners, MRIs, and sonography systems all identify, evaluate, and track problems for patients. With embedded technology, these imaging options can process higher-quality images faster and with more accuracy—with lower radiation exposure.

Aerospace and Military

It comes as no surprise that embedded high-performance sensors and navigation systems are critical to the success of aviation, military, and space activities. In fact, embedded systems and IoT solutions are responsible for planes taking off and landing in the first place—for the satellites circling Earth, sending and receiving signals.

But how else are embedded systems used in the military? They're used in intelligence gathering operations, for one. Embedded systems control and secure the devices used to gather military intelligence (for example, broadcasts of radio; military, telegraph, and satellite traffic; television; and radar emissions) while also analyzing the massive volumes of data collected.

The military also uses embedded devices for unmanned aerial vehicles (UAVs). Whether they're being used for surveillance and reconnaissance missions or for transporting goods in war zones, they're able to execute their function while inspecting, mapping, and remote sensing along the way. One of the fastest-growing segments of the UAV market is that of extended visual-line-of-sight (EVLOS) military drones. They enable leaders to visualize areas of interest from a safe distance, minimizing the risk to human lives.

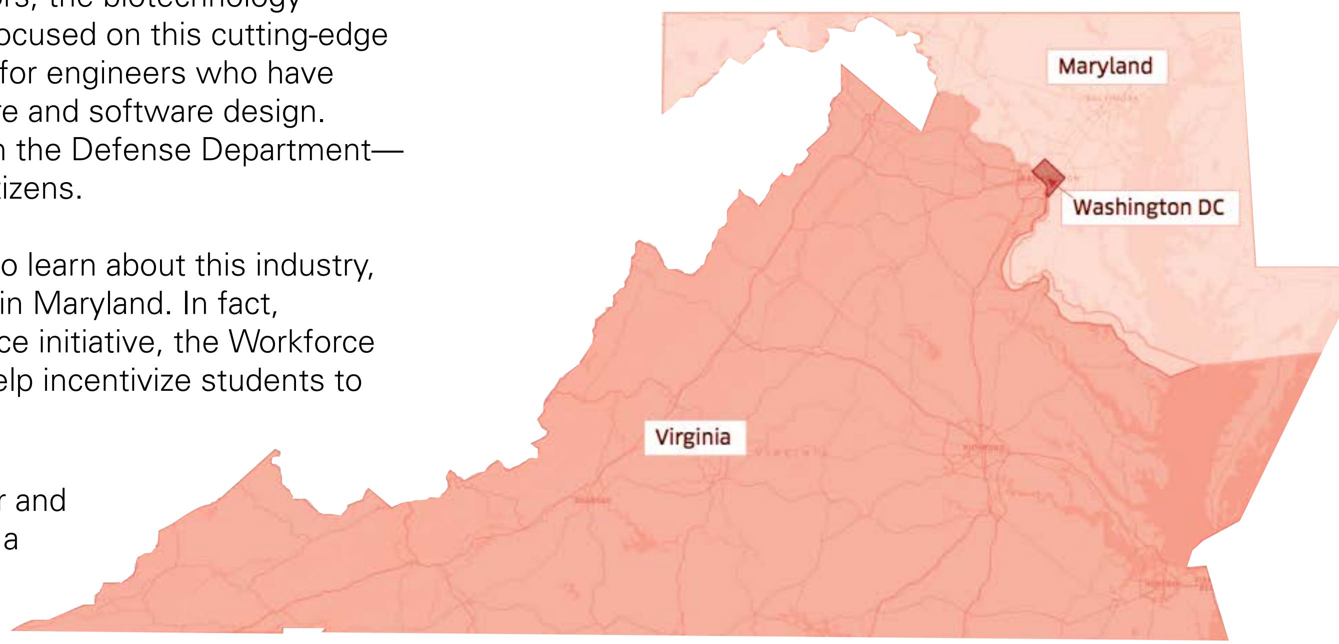
The Need for Embedded Systems Experts

Despite the widespread use of embedded systems, this technology is truly still in its infancy. In the years to come, industry leaders anticipate even more powerful computers, lower energy consumption, and faster and more complex communication systems. That means we need people who understand embedded systems programming—who can grow and evolve as the technology does.

The Washington D.C., Maryland, and Virginia (DMV) area is the central hub for defense contractors, the biotechnology industry, and many other industries focused on this cutting-edge technology. There is a growing need for engineers who have specialized in emerging tech hardware and software design. Much of this work—especially jobs in the Defense Department—can only be filled by United States citizens.

You won't find a more ideal location to learn about this industry, get a degree, and start a career than in Maryland. In fact, Maryland has even started a workforce initiative, the Workforce Innovation and Opportunity Act, to help incentivize students to graduate in STEM fields.

All this adds up to unparalleled career and internship opportunities. But is there a future in ESIOT?

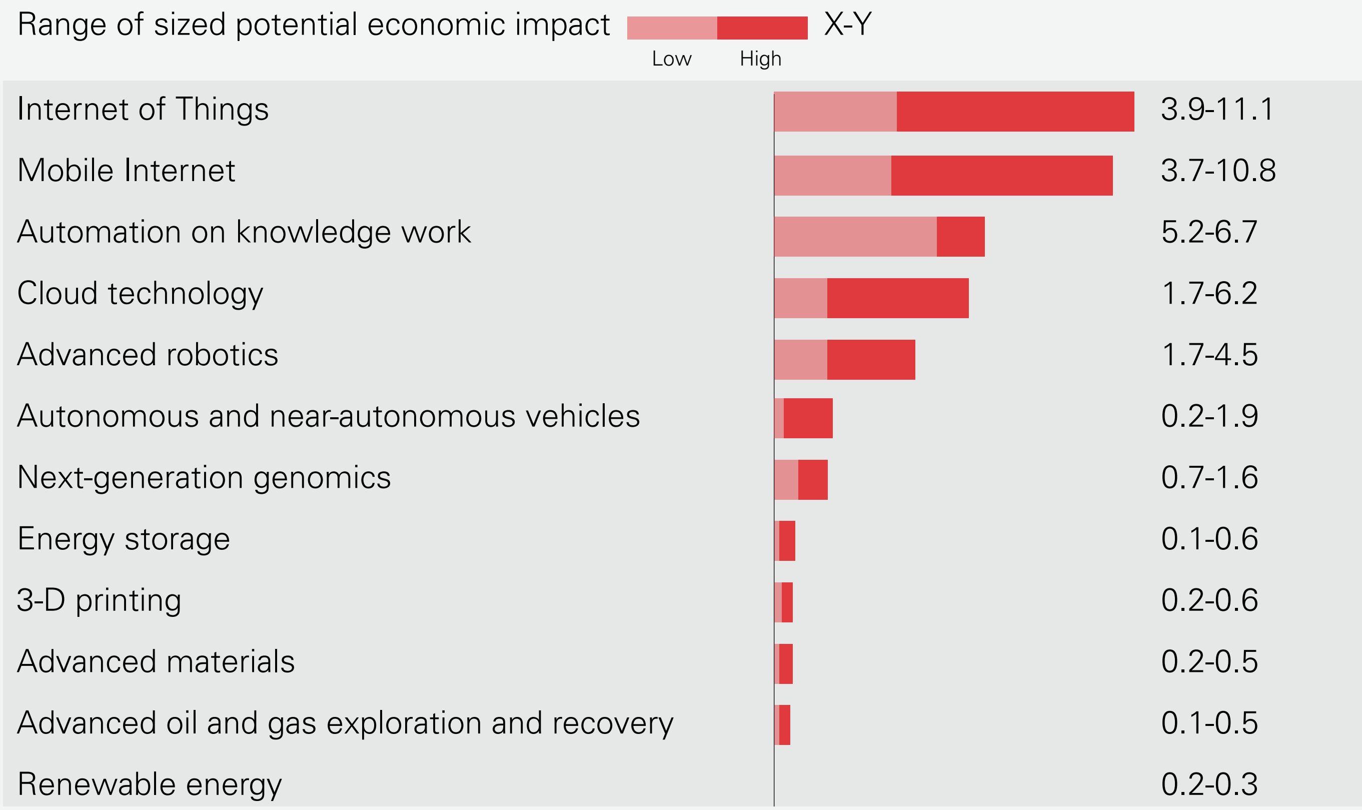


Fast Forward—What's the Future of Embedded Systems and IoT?

According to a 2018 QYResearch report, the global market for the embedded systems industry was valued at $68.9 billion in 2017 and is expected to rise to $105.7 billion by the end of 2025.

As for the Internet of Things industry—by 2025, it has a projected market value of 6 trillion US dollars worldwide, where it will be powered by 60 billion connected devices.

The Internet of Things will have substantial economic impact by 2025 among a list of disruptive technologies.



Needless to say, you'll have a bright future with a career in ESIOT.

Explore Some of the Many Career Opportunities in Embedded Systems

Check out some of the positions that a degree in ESIOT prepares you for, and their average starting salaries in the Washington, DC area.

Hardware Developer

Average starting salary: $90,639

In this role, you'll research, design, and manufacture systems and components to ensure they're functioning to their full capacity. Some of your duties may include designing new hardware, manufacturing equipment schematics, analyzing test results, and modifying the design of new hardware as needed.

Applications Developer

Average starting salary: $106,957

An application software developer is responsible for creating and designing software systems and applications according to business demands or client specifications. You'll operate various system tools and programming codes to customize programs, implement solutions, modify test codes, and update existing applications to boost efficiency and ensure optimal performance.

Software Developer

Average starting salary: $100,640

Software developers are responsible for designing application systems to support the user and business' needs. You'll test software codes, troubleshoot, interpret system data, and establish efficient parameters. You'll need an excellent knowledge of computer programs and programming languages, especially in order to detect defects and malfunctions.

Software Analyst

Average starting salary: $80,600

Responsible for creating and designing software programs and applications, a software analyst is also expected to modify existing ones for optimization according to business requirements. You'll work alongside the technical team to draw system codes, analyze programming languages, and ensure the stability and efficiency of software navigation by running multiple quality checks on the system. You'll consider the application's performance, configure servers, and improve software infrastructure according to quality findings.

IT Specialist

Average starting salary: $85,865

An Information Technology Specialist is responsible for the computer support of an establishment or individual. Because the tasks of an IT specialist are so diverse, you'll need extensive and proactive knowledge of IT: the aspects of network and systems administration, security and information, hardware and software management, troubleshooting, and more. Your tasks won't be limited to installation and activation; expect to monitor, manage, and analyze, as well.

IoT Engineer

Average salary: $145,928

IoT Engineers, or IoT Developers, need to have fluency in all of the IoT stacks: device-end, cloud-end, user-end, and connectivity. That means working knowledge of mechanical engineering, electrical engineering, firmware engineering, software engineering, data science, and UX design. The IoT engineer needs to understand how all these pieces fit together to form a whole.

IoT Infrastructure Architect

Average salary: $161,059

As an IoT Infrastructure Architect, you'll have to collaborate with a diverse group in order to communicate and spearhead the development of the IoT vision and technical strategy. Once that's finished you'll be the one designing the architecture from one end to the other. You'll need to work closely with IoT Solutions Designers to enable the design and construction of IoT solutions while outlining a process to build those solutions.

IoT Solution Designer

Average salary: $143,271

Also known as an IoT Solution Architect, the Solution Designer looks at the big picture and collaborates across teams to promote IoT solutions development. In some cases, Solution Designers need no programming experience, but you should be comfortable huddling with Infrastructure Architects or other engineers to figure out the best path forward for a particular implementation. The Solutions Designer needs to know the whole puzzle and is often helping communicate needs throughout the organization. You'll need to be comfortable speaking with executives to get buy-in, choosing and managing vendors, and wading through the trenches, building out systems.

Cyber Security Analyst

Average starting salary: $87,403

A Cyber Security Analyst is responsible for planning and carrying out security measures to protect a company's computer networks and systems. Your job is to constantly keep tabs on threats and monitor your organization's networks for security breaches. Day-to-day, expect to be installing computer programs or software, encrypting devices, reporting weak spots or security breaches, exploring new IT trends, and educating the company's information security team. Occasionally, you'll do your own version of a fire drill where you simulate cyber attacks to find possible network and system vulnerabilities.

Embedded Systems Engineer

Average salary: $95,355

An embedded systems engineer is responsible for monitoring and creating embedded network systems and servers to support business operations. EEs run multiple diagnostic tests to identify technical solutions that would increase the efficiency and stability of the systems. You'll be expected to update infrastructure to boost optimization and identify opportunities for enhancing engineering techniques. You'll need excellent technical skills for network resolution, as well as a strong command of programming languages and system codes to write scripts and generate support for data integration.

Cyber Security Engineer

Average salary: /$107,593

Cyber Security Engineers specialize in designing and establishing security measures to protect a company's network and data from cyberattacks or hackers. Responsibilities include planning and implementing cybersecurity systems, responding to problems or concerns, handling security breaches and emergencies, and troubleshooting to identify vulnerabilities. The job may require you to do clerical work as well, so expect to create instruction manuals for systems, manage schedules, produce progress reports, and coordinate with department managers.

Wireless Engineer

Average starting salary: $121,199

With the increase in mobile phone users at a record level, Wireless Engineers have never been more important. This individual provides support to Internet and wireless phone subscribers in a specific area by inspecting, monitoring, and maintaining all wireless equipment and facilities in that area. You'll make sure that wireless coverage in a zone remains active, linking all users with their subscription speed and signal strength.

Cloud Engineer

Average starting salary: $113,708

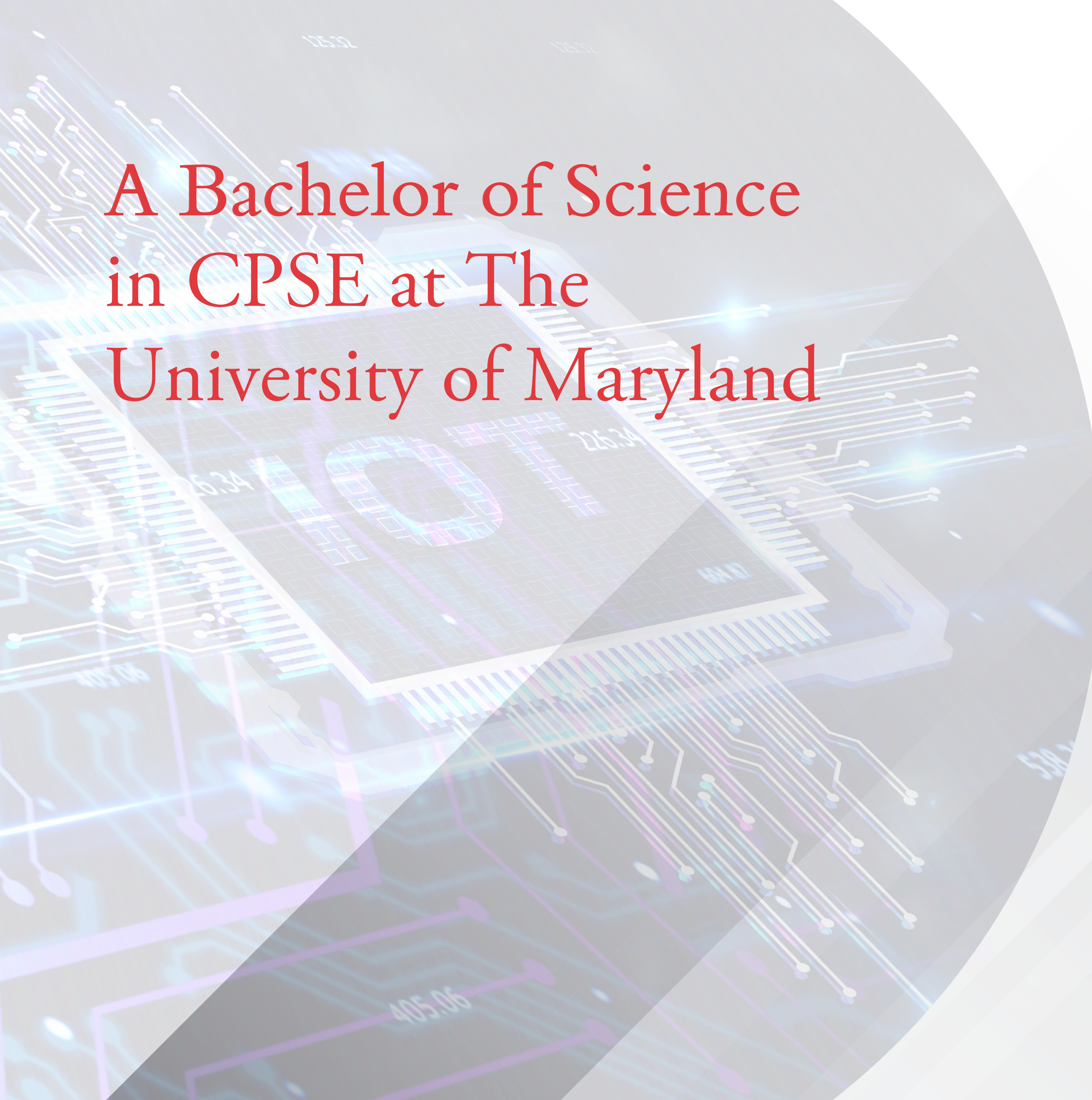
Cloud engineers assess the existing infrastructure of a business and research solutions for moving different functions (for example, database storage) to a cloud-based system. This person is responsible for migrating the function to the new system and then maintaining it. You'll need technical abilities, as well as the ability to negotiate with vendors, ensure the security of the data, and implement best practices throughout the process.

Cloud Developer

Average salary: $128,876

Cloud Developers create applications that are served on the cloud. They're essentially software engineers with a specialization in cloud computing. So, on top of development experience, Cloud Developers need to have a deep understanding of cloud systems: how they operate and how to deploy them securely, efficiently, and with little-to-no downtime. Day-to-day, expect to analyze customer needs, design systems and solutions, code, and debug.

A Bachelor of Science in CPSE at The University of Maryland



3

If you enjoy programming and working with embedded systems, there's a degree that could help turn your passion into a career: the Bachelor of Science in Cyber-Physical Systems Engineering (CPSE) at the University of Maryland.

Based at the Universities at Shady Grove campus, this engineering program is fast, efficient, and powerful. It combines elements of computer engineering, computer science, and information technology all into one extremely focused degree. In addition to engineering fundamentals and problem-solving skills, you'll focus on:

- field-programmable gate array (FPGA)
- microprocessors and microcontrollers
- hardware-descriptive languages (Python, C++, C, Linux, Javascript)
- hardware and software design with specializations in the areas of networks, cybersecurity, and machine learning
- and more!

As the only undergraduate program in the DMV area and the northeastern United States, we're proud to be offering it at the Department of Electrical & Computer Engineering at the University of Maryland, College Park. The Bachelor of Science in CPSE trains students to perform in the technology setting—even as the tech continues to advance.

This program provides strong engineering foundations and teaches you how to continue learning. And after you graduate, you'll be able to make an immediate contribution to private and public sector institutions.

Embedded Systems Course Highlights

The Bachelor of Science in CPSE program is extremely hands-on—and that's by design.

Dr. Mel Gomez, Program Director, believes that the only way to achieve ownership of knowledge—knowledge that's actually retained—is by working with your hands. Hands-on creates deep insights. Students need to understand the theories behind why things work. This program starts there, then takes it a step further by applying theory to real-life scenarios to improve existing technology.

Hands-on is more than just a lab component—though nearly all CPSE courses have one of those, too. At the Universities at Shady Grove, students work with instruments in the lab ***and take them home with them***—to work with them on their own time.

Working with embedded systems means these instruments are not only extremely expensive but are also hard to scale. Other programs have a hard time funding the hands-on portion that's needed for a program like this. But for the Bachelor of Science in CPSE offered at the Universities of Shady Grove, the hands-on aspect was non-negotiable. It's part of the training, part of creating codes and software; students need ample time and resources to learn by doing.

Check out some of the embedded systems courses we've highlighted below!

ENEB344: Digital Logic Design for Embedded Systems

Hands-on approach on foundations of digital logic for embedded systems applications; including input/output, logic gates, Karnaugh maps, latches, flip-flops, and state-machines. This course also covers design and analysis of synchronous sequential systems, multiplexers, decoders, encoders, binary arithmetic logic units such as adders, subtractors, multiplier, and divider, and implementation with PLA's. Hardware implementation to an FPGA board with OLED display.

ENEB341: Introduction to Internet of Things

An introduction to the foundations of Internet of Things (IoT), including IoT devices, communications, connection considerations, back-end services/applications, and business models. This course looks at the IoTs as the general theme of physical/real-world things becoming increasingly visible and actionable via Internet and Web technologies. It also covers networking protocols and gateways, security and privacy, data analytics, and cloud computing platforms.

ENEB340: Intermediate Programming Concepts & Applications for Embedded Systems

Introduction to programming for embedded systems development. Includes principles of software development in Unix, C, and other high-level languages, input/output, data types and variables, operators and expressions, program selection, repetition, functions, arrays, strings, introduction to algorithms, software projects, debugging, documentation. Includes hands-on applications in microprocessor environments.

Explore the Curriculum

Is the Bachelor's In CPSE Right For You?

Any STEM degree is tough, but this one is particularly rigorous. We're looking for people with grit—unique students who are focused, driven, and endlessly curious.

Engineering is the art of solving problems. And one of the best ways to approach problem-solving is with a diverse group of backgrounds and experiences. We have students who are starting their second career; we have single mothers. What they all have in common is their curiosity, grit, and investment in their future. We invest in you when you invest in us.

If you're a student finishing your associate's degree at a community college with a passion for CPSE, this program is perfect for you. Among other grants and scholarships, the Maryland Higher Education Commission has set up the 2+2 Transfer Scholarship, which is designed to assist and encourage transfer students from Maryland community colleges to attend a 4-year institution within the state. Earning an associate's degree from a Maryland community college will typically waive the UMD general education requirements.

In addition, we will be opening this program to first-year students beginning in Fall 2022. For more information on the admissions process for first-year students, please contact es-sg@umd.edu or visit our admissions page.

Meet Current CPSE Student, Ricardo

Ricardo Rabello, 44, holds a Pharmacy degree, but after a decade of practicing, he found the work repetitive. Wanting to learn new skills, he decided to get an Associate's degree in Mechanical Engineering at Montgomery College. There, he was introduced to the Bachelor's in Cyber-Physical Systems Engineering.

Through the Maryland Higher Education Commission 2+2 scholarship, he was able to transfer to the University of Maryland, waive any general education requirements, dive into his passions, and begin his journey to a new career.

What factors sold you on enrolling in the program?

I found it very attractive to tie a hobby of mine (tinkering with MCUs, sensors, etc.) to a professional path. But it was also the proximity to my house, small cohort, and being part of the prestigious James A. Clark School of Engineering.

What qualities do you think a student needs to be successful in this program?

You need a problem-solving-oriented mind. You need to be resourceful and not be intimidated by the unknown.

If you like bare-metal programming, you're really going to enjoy getting your bachelor's in CPSE because embedded systems are oriented towards efficiency. It can be easy to set up a whole smart house controller on a PC. But doing that in an MCU with 128kb of heap memory and running on battery? It becomes tricky.

What's your favorite class and why?

Digital Logic Design. It brings the applications down to gate level and makes you really think about what you are doing with every bit in your solution.

Outside of classes, how is USG helping to prepare you for the next step in your career (internships, career fairs, networking opportunities, etc.)?

Internships. In my internship, I'm getting a valuable lesson on how far I still have to walk before running. It's a very pleasant and very humbling experience.

What are your research interests and/or what parts of CPSE do you find compelling?

FPGA (Field Programmable Gate Arrays) / ASIC (Application-Specific Integrated Circuit) design.

What would you say to a student considering the program who's worried about finding a job afterward?

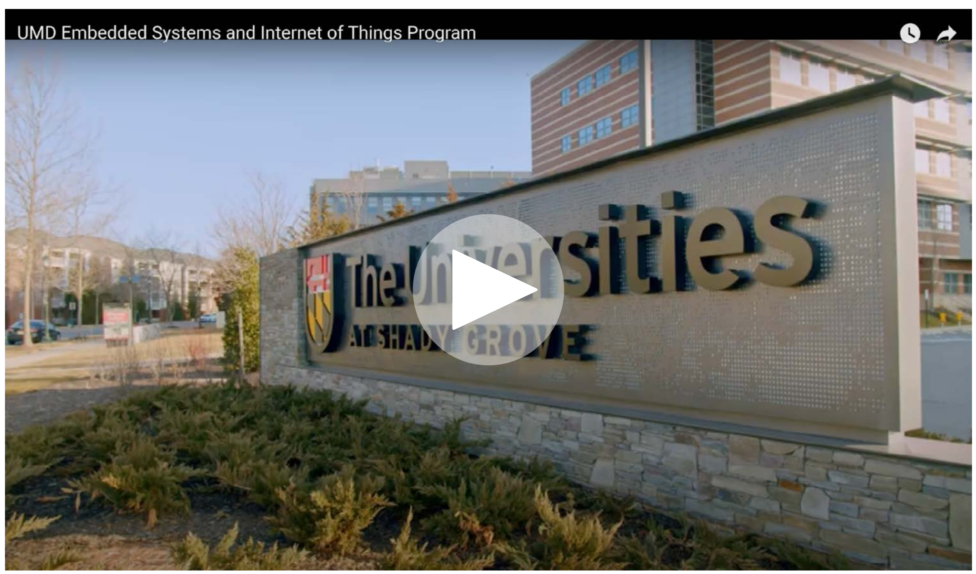
That's simple. Do a quick search for "Firmware/Embedded Systems Engineer" job opportunities, and that will answer your question.

What would you say to a student considering the program who's worried about the cost?

Six months of an entry salary as an Embedded Systems Engineer will pay for the cost of the program.

What are some of the most important skills you've learned so far? What have you learned that you couldn't have learned anywhere else?

I wouldn't say that what I learned could not be learned anywhere else. However, the way it was taught, with professors that are experts in their respective fields and lots of hands-on experience, is what makes the difference.



Applying to the Universities at Shady Grove

At this time, the **Cyber-Physical Systems Engineering** program at the Universities at Shady Grove only admits students for the Fall session. It's a two-year program.

Have admissions questions? Reach out to **CPSE** Program Coordinator, Ms. Glenn Kasten-Sportes, at es-sg@umd.edu.

Are There Any Requirements or Prerequisites to Learn IoT and Embedded Systems?

The **Cyber-Physical Systems Engineering** major is a two-year, cohort-based program with a pre-set class schedule starting in the fall semester. To be eligible for admission into the program, students must satisfactorily complete all lower-level general education requirements (likely by earning an associate's degree from a Maryland community college), and should have at least 60 transfer credits prior to starting the program.

Along with 60 transfer credits, you need to fulfill the Clark School of Engineering Limited Enrollment Program (LEP) requirements in order to be eligible for admission in the Bachelor of Science in **CPSE** program. That means you must have at least a 3.0 GPA and have completed the following courses:

- Calculus II
- General Physics I: Mechanics & Particle Dynamics
- General Chemistry for Engineers

There are additional requirements outside of the School of Engineering Requirements. The **CPSE** program also requires that you have completed these additional courses with at least a C-.

- Introduction to Engineering Design
- Calculus I
- General Physics II: Vibration, Waves, Heat, Electricity & Magnetism
- Introductory Programming (C, C++, Java or Python)
- One of these math courses:
  - Differential Equations
  - Calculus III
  - Linear Algebra

Compare Equivalent Courses

Scholarships and Financial Aid

There are plenty of scholarships available to students interested in the Bachelor of Science in CPSE. What's more—students are eligible for *both* University of Maryland merit scholarships (including Clark School of Engineering ones) and Universities at Shady Grove (USG) scholarships!

Make sure to apply to both, but there's likely a higher chance of funding at USG. In fact, 50 percent of applicants receive a Universities at Shady Grove scholarship, with over $1 million in funds awarded each academic year to students enrolled in a USG program.

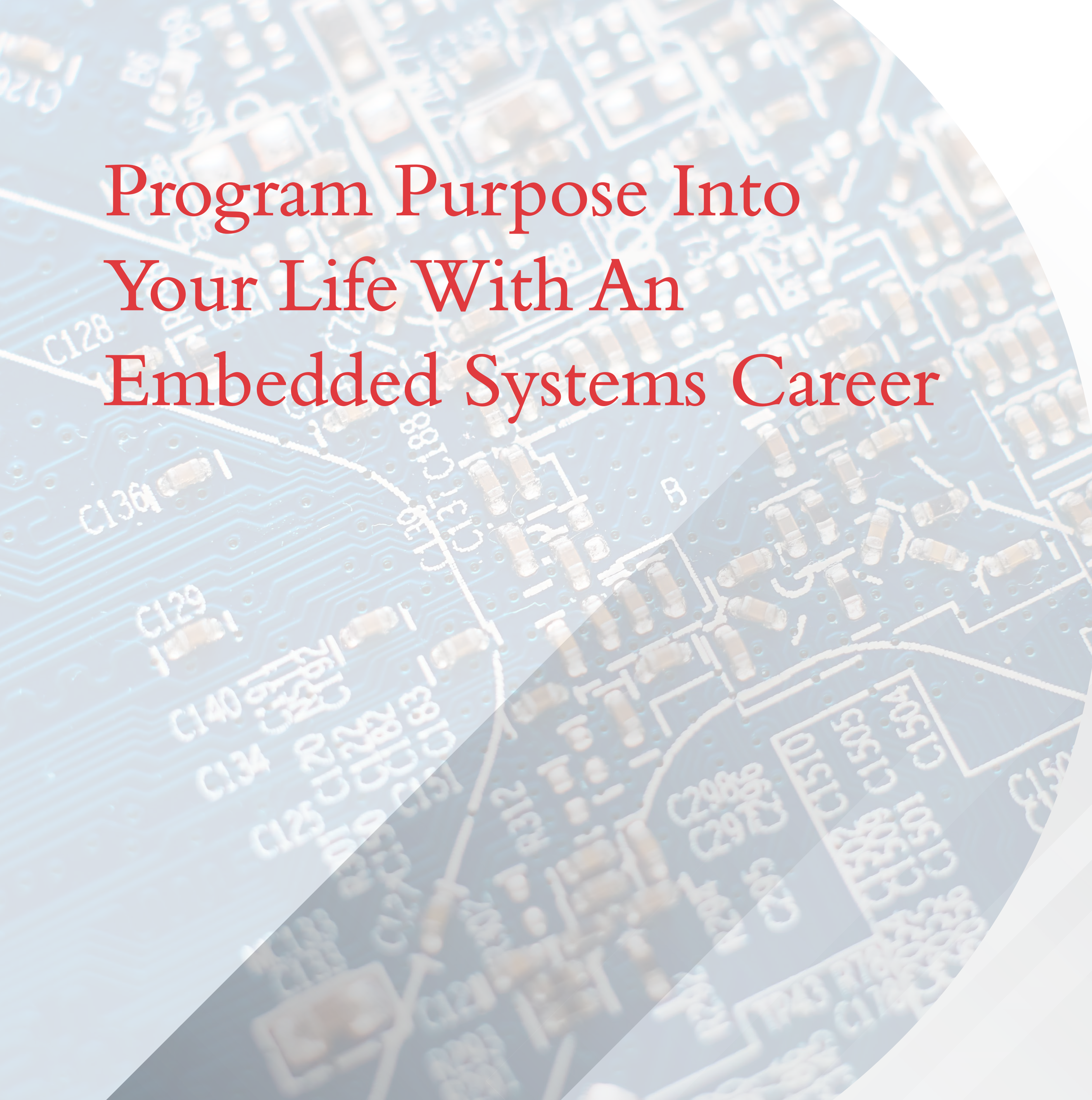
Frederick Douglass Scholarship	Formerly known as the *Transfer Academic Excellence Scholarship*, this prestigious scholarship is for academically talented transfer students from a Maryland community college. This exceptional scholarship honors the life and legacy of Frederick Douglass. Find out if you're eligible.
Maryland Transfer Scholarship	This scholarship provides partial tuition awards ranging from $5,000, distributed equally over two semesters, to $10,000, distributed equally over four consecutive semesters. Only in-state students transferring to the University of Maryland from a two- or four-year institution are eligible. Learn more.
President's Transfer Scholarship	Are you an out-of-state transfer student enrolling at the University of Maryland? This scholarship provides funding for a maximum of two years. The annual award is in the amount of $5,000. No separate application is required for consideration, as recipients are identified when their application materials are being evaluated.
Clark School Scholarships	Current and admitted students can complete this application annually to be considered for a variety of scholarships awarded by the Clark School and its academic units.

Housing and Food Insecurity Resources	Experiencing hardships related to housing or food expenses? Visit this page to learn more about the resources available to UMD students.
Scholarships at the Universities of Shady Grove	Visit the scholarship page for the full list. USG offers a wide range of scholarship funds covering everything from textbooks to full tuition in order to help students reach their academic goals.
2 + 2 Transfer Scholarship	Students who enroll in a science, teaching, engineering, computer science, mathematics, or nursing program will receive an annual award of $2,000. Read more about it.
Maryland Delegate & Senatorial Scholarships	These scholarships are awarded by Delegates and State Senators to students in their legislative districts.

Explore the Scholarships Database

Find Third-Party Scholarships

Program Purpose Into Your Life With An Embedded Systems Career



4

Learn More About the Cyber-Physical Systems Engineering Degree

> I believe that at the end of the century the use of words and general educated opinion will have altered so much that one will be able to speak of machines thinking without expecting to be contradicted.
>
> – Alan Turing, computer science pioneer, in 1947

Want a career that challenges you? How about a future you can count on?

We're calling all programmers, DIYers, tinkerers, microcontroller hobbyists, robot enthusiasts, and drone operators! Getting your Bachelor's in **Cyber-Physical Systems Engineering** is a great way to turn your passion into a career.

Get your education in the hub of the embedded systems industry by attending the University of Maryland, College Park at the Universities at Shady Grove campus.

Connect with us for additional resources about the Cyber-Physical Systems Engineering degree > **Request More Information**

Dig into your future subject matter: machine learning, artificial intelligence, cybersecurity, and more. > **Explore curriculum**