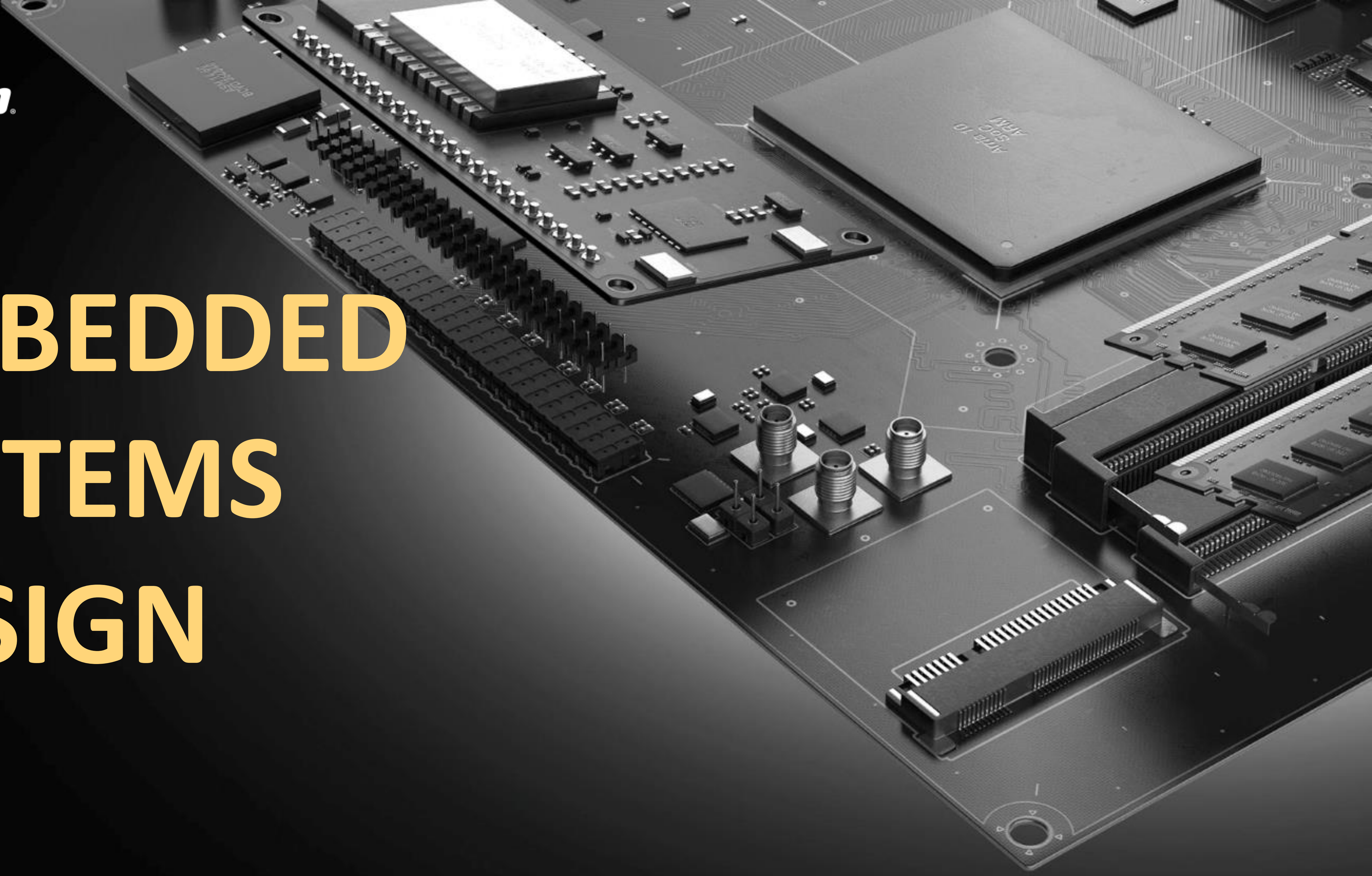
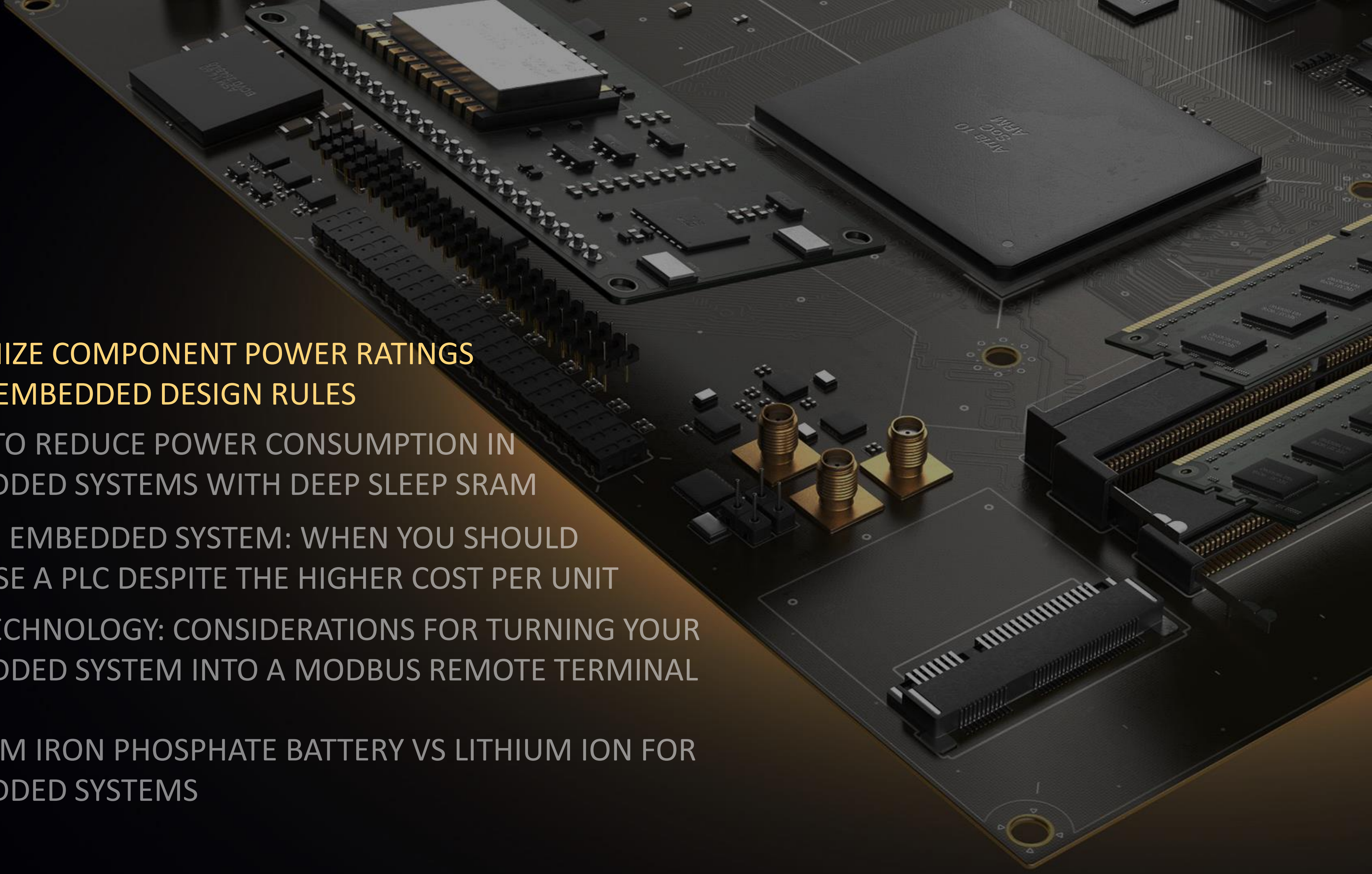
Altium



EMBEDDED SYSTEMS DESIGN

Altium



1. OPTIMIZE COMPONENT POWER RATINGS WITH EMBEDDED DESIGN RULES
2. HOW TO REDUCE POWER CONSUMPTION IN EMBEDDED SYSTEMS WITH DEEP SLEEP SRAM
3. PLC VS EMBEDDED SYSTEM: WHEN YOU SHOULD CHOOSE A PLC DESPITE THE HIGHER COST PER UNIT
4. PCB TECHNOLOGY: CONSIDERATIONS FOR TURNING YOUR EMBEDDED SYSTEM INTO A MODBUS REMOTE TERMINAL UNIT
5. LITHIUM IRON PHOSPHATE BATTERY VS LITHIUM ION FOR EMBEDDED SYSTEMS

1. OPTIMIZE COMPONENT POWER RATINGS WITH EMBEDDED DESIGN RULES

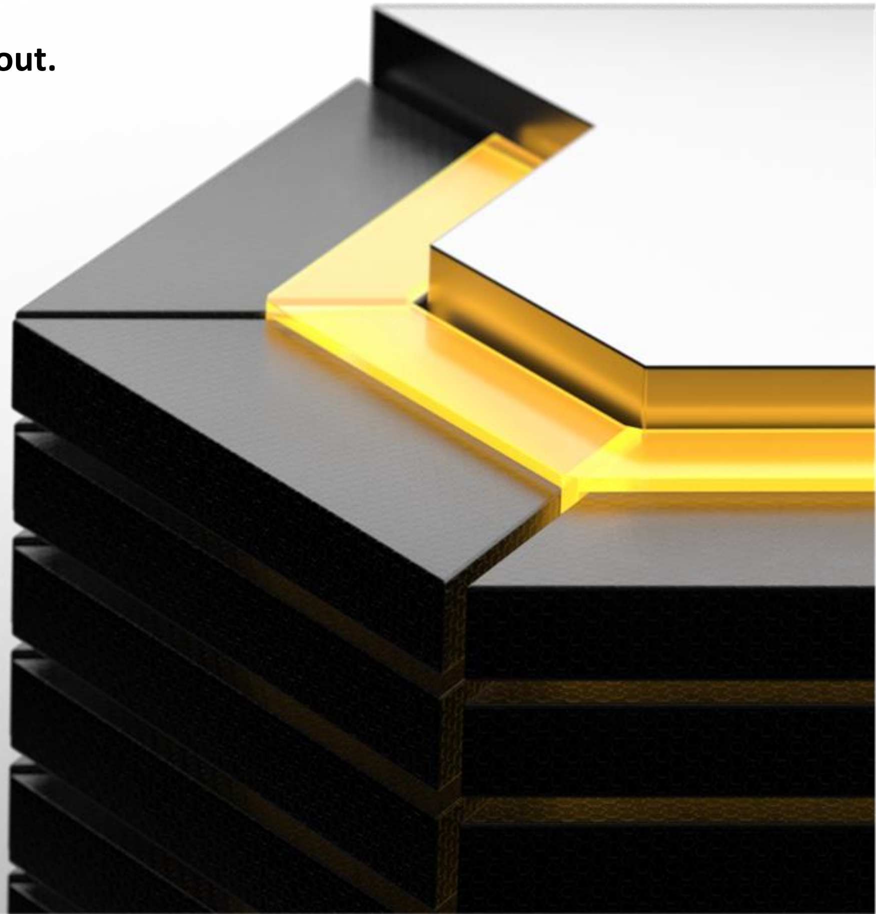
Altium

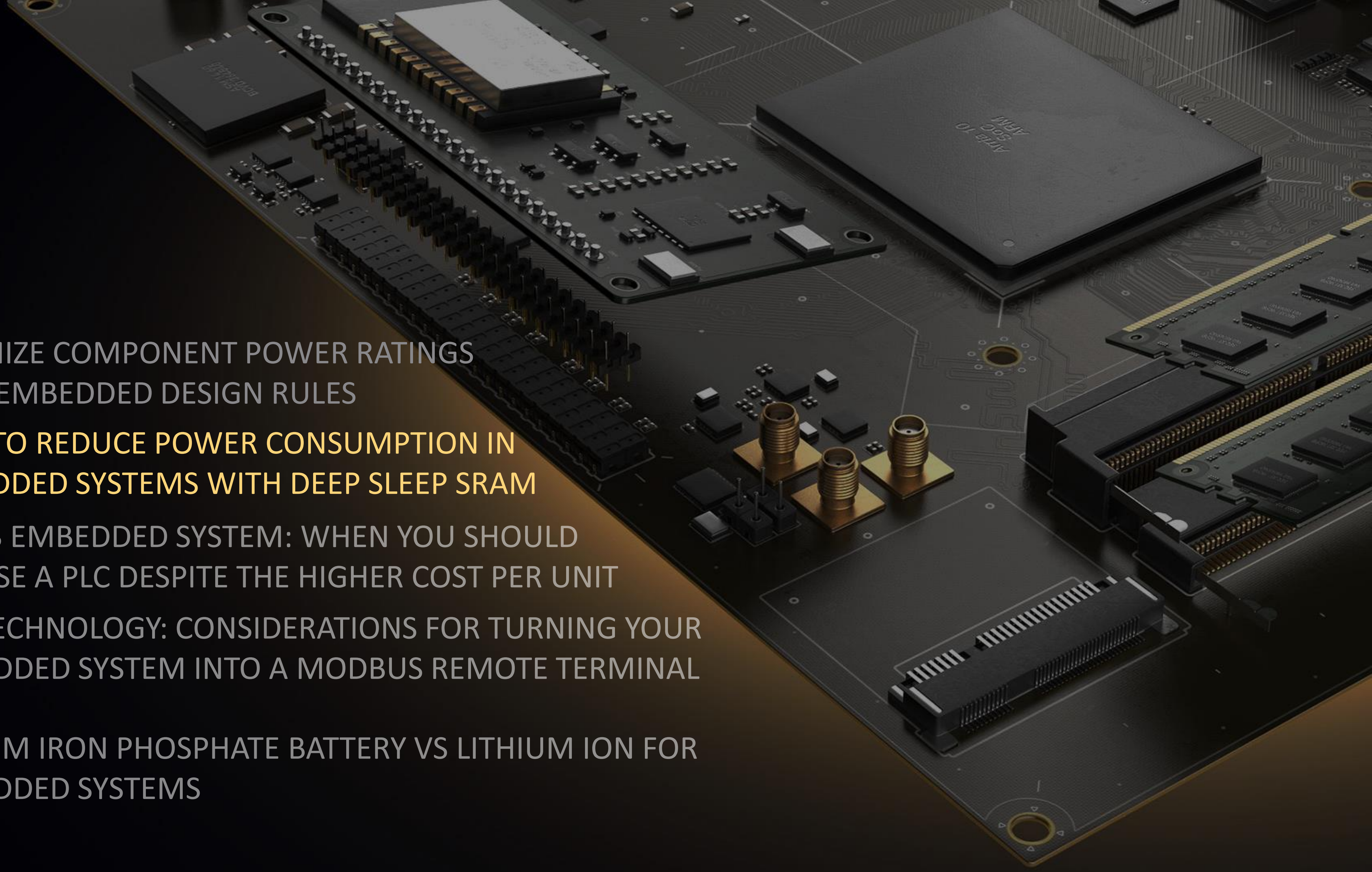
Embedded Design Rules

Ensuring your initial rules are properly applied the first time is critical to eliminating the need for redesign. By creating conditional limitations and implementing embedded design rules, you can circumvent the potential for error to avoid respins and ensure precise power ratings in your PCB layout.

- **Maintain a list of design rules to quantify critical components:** Embedding a few design rules will help to keep small changes from devastating a design and save time on team designs.
- **Create and embed design rules so that they last forever:** It is important that you define the constraints of your power dissipation before assigning design rules to manage them. The best way to do this it to make a datasheet. A well-written datasheet will typically have at least three classifications of power ratings that range from the best to worst case scenarios.
- **Use stitching vias:** Stitching vias are thermal conduits to the other side of the PCB. Often if components are too crowded on one side of the board, stitching vias can be used quite effectively to conduct the heat to the other side of the board.
- **External heat sources:** Consider the likelihood of heat conducting to your PCB from external sources.

See original content here





1. OPTIMIZE COMPONENT POWER RATINGS WITH EMBEDDED DESIGN RULES
2. HOW TO REDUCE POWER CONSUMPTION IN EMBEDDED SYSTEMS WITH DEEP SLEEP SRAM
3. PLC VS EMBEDDED SYSTEM: WHEN YOU SHOULD CHOOSE A PLC DESPITE THE HIGHER COST PER UNIT
4. PCB TECHNOLOGY: CONSIDERATIONS FOR TURNING YOUR EMBEDDED SYSTEM INTO A MODBUS REMOTE TERMINAL UNIT
5. LITHIUM IRON PHOSPHATE BATTERY VS LITHIUM ION FOR EMBEDDED SYSTEMS

2. HOW TO REDUCE POWER CONSUMPTION IN EMBEDDED SYSTEMS WITH DEEP SLEEP SRAM

SRAM in Embedded Systems

There are many different types of memory that are designed for embedded systems, including new entrants like CBRAM and Spin Wave devices. Some of the more traditional types are Flash, SRAM, and DRAM and each has its own place in the system.

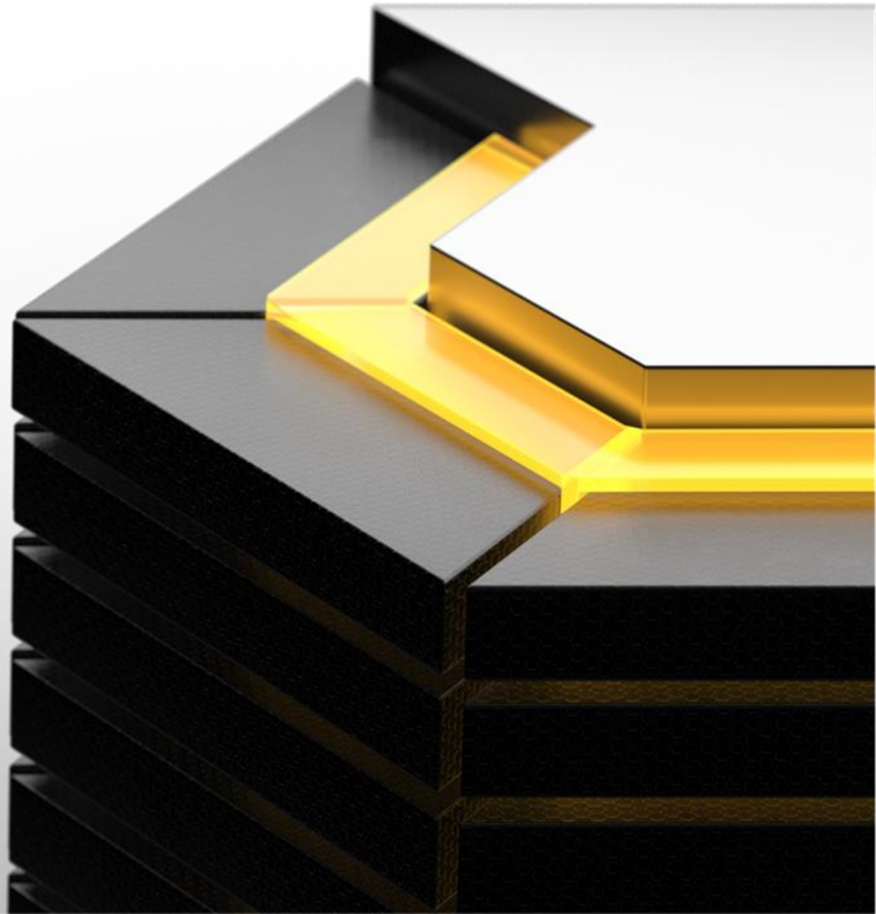
- DRAM has to be refreshed in order to retain data, making it use more energy than the other two types. It’s often used for general memory.
- Flash is often used to store code.
- SRAM is often used as a cache.

Types of SRAM

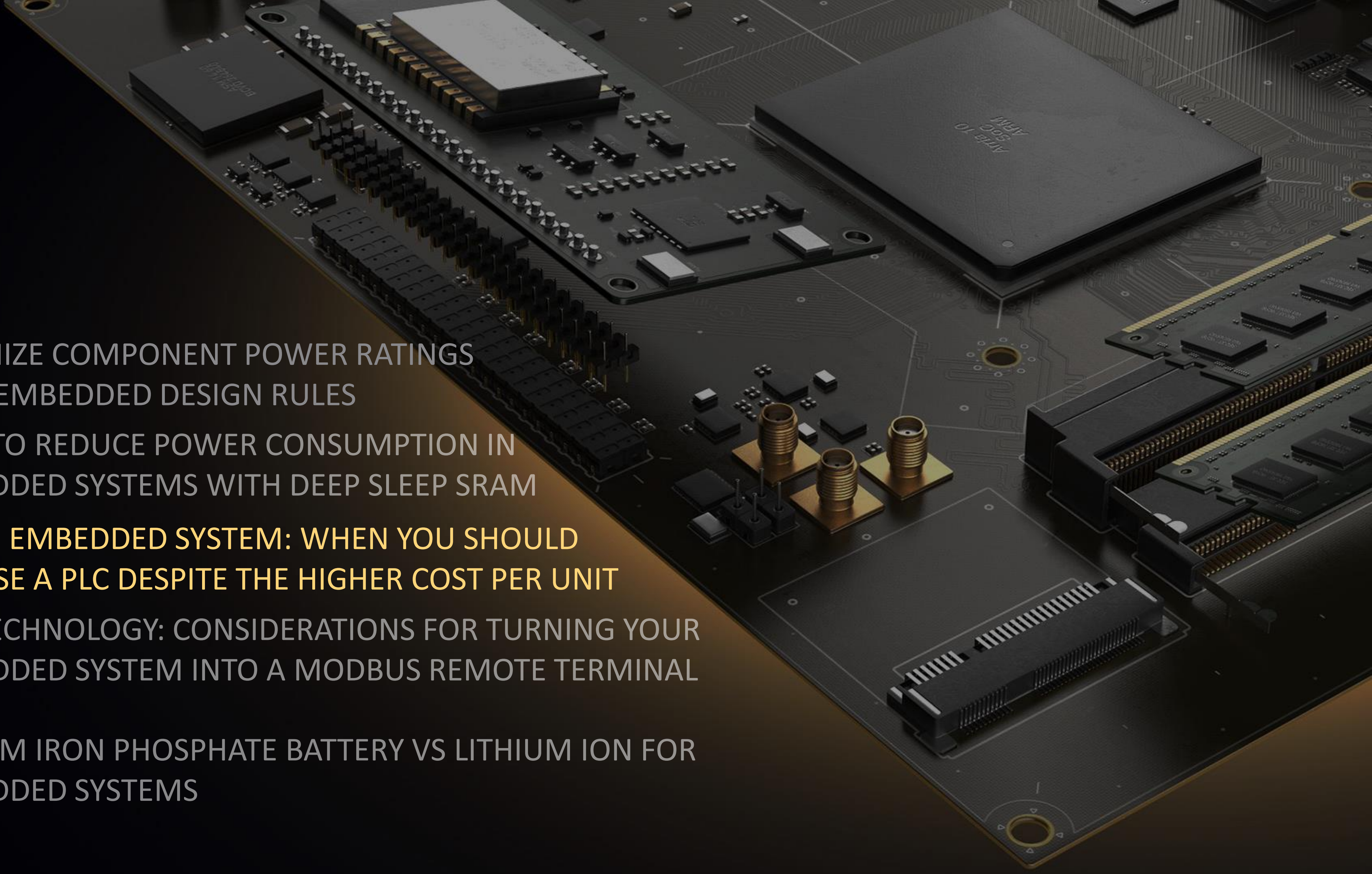
It’s important not only to know that you will use SRAM but exactly which type. Generally, SRAM with a deep sleep option can give you both the speed you need and save some energy.

- Before the explosion of portable electronic devices, SRAM mostly focused on speed. Since we started carrying around super-computers with power hungry processors, though, low power SRAM has found a niche.
- Deep sleep SRAM is a hybrid type that has a 10 ns access time.
- When used properly, deep sleep SRAM can give you the performance you need, while saving you nearly as much energy as a low power type.
- Before implementing deep sleep SRAM in your system make sure that you can get the most out of its deep sleep capabilities.

See original content here



Altium



1. OPTIMIZE COMPONENT POWER RATINGS WITH EMBEDDED DESIGN RULES
2. HOW TO REDUCE POWER CONSUMPTION IN EMBEDDED SYSTEMS WITH DEEP SLEEP SRAM
3. PLC VS EMBEDDED SYSTEM: WHEN YOU SHOULD CHOOSE A PLC DESPITE THE HIGHER COST PER UNIT
4. PCB TECHNOLOGY: CONSIDERATIONS FOR TURNING YOUR EMBEDDED SYSTEM INTO A MODBUS REMOTE TERMINAL UNIT
5. LITHIUM IRON PHOSPHATE BATTERY VS LITHIUM ION FOR EMBEDDED SYSTEMS

3. PLC VS EMBEDDED SYSTEM: WHEN YOU SHOULD CHOOSE A PLC DESPITE THE HIGHER COST PER UNIT

PLCs and Their Applications

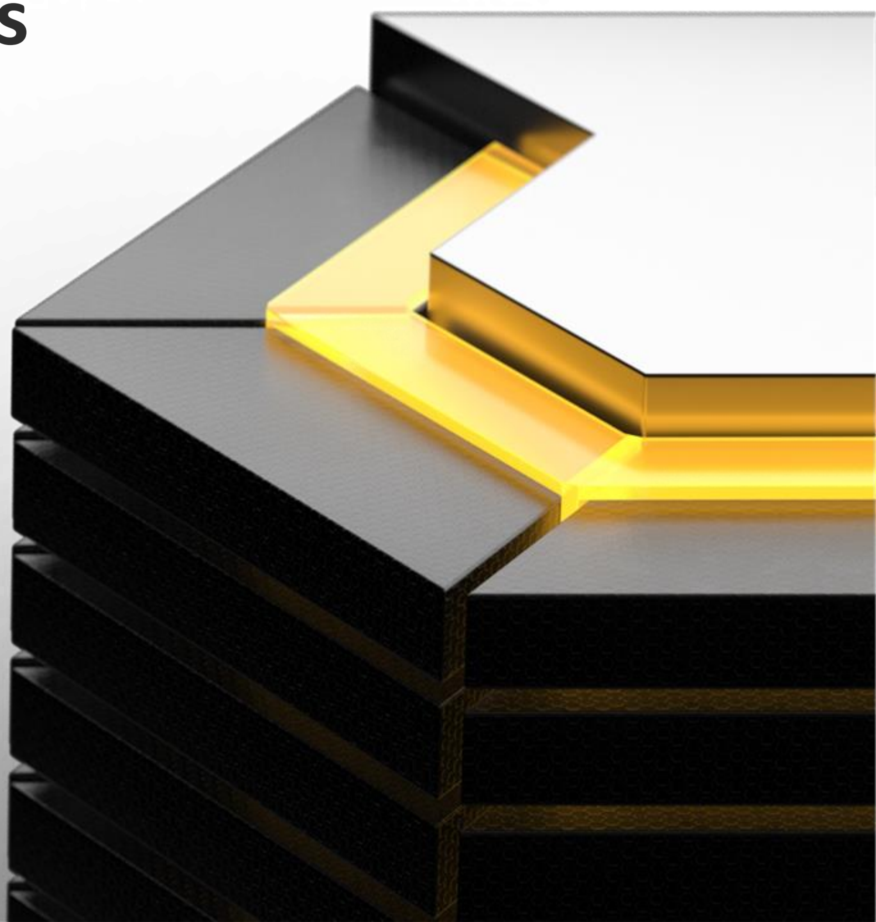
A programmable logic controller is a specialized industrial computer. It is custom programmed to monitor input signals (digital or analog), perform logical operations, and trigger specific output signals. PLCs are known to be rugged and are commonly used in extreme industrial environments or applications that have almost no room for failure.

- PLCs are popular because of their modular structure, which makes them easy to install in a plug-and-play manner.
- They consist of a central processing unit (CPU), a power supply and input/output (I/O) modules.
- Programming PLCs is less complicated than coding a microcontroller, because it revolves around ladder diagram, functional block diagram, and structured text on software provided by the manufacturers.
- PLCs are often a preferred option in automated control systems.

Localized Embedded Systems Replacing PLCs

In general, an embedded system is defined as a dedicated set of hardware and software working together to provide application-specific functionality. From an electronic designer's point of view, it consists of a microcontroller (MCU), memory chips, power management circuits, communication modules and input/output functionality. This might sound similar to a PLC, but there are distinct differences between them:

- **Configuration:** PLCs are known to be modular and easily replaceable if a specific module fails, while embedded systems are mostly designed as a single board.
- **Programming:** PLC programming is generally less time consuming since they come with easy to understand ladder diagrams, while embedded systems are coded in high-level programming languages, like C, and their firmware takes longer to write.
- **Cost:** PLCs are associated with having a high cost, while an embedded system, when localized, has a very low production cost per unit.



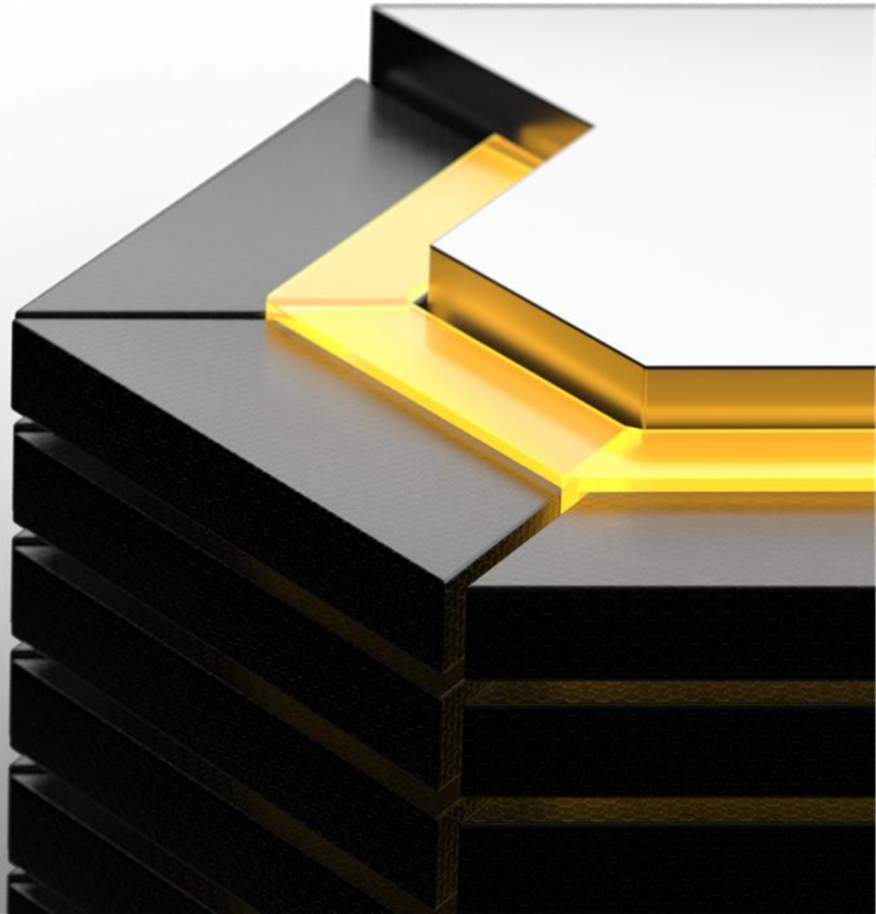
Altium

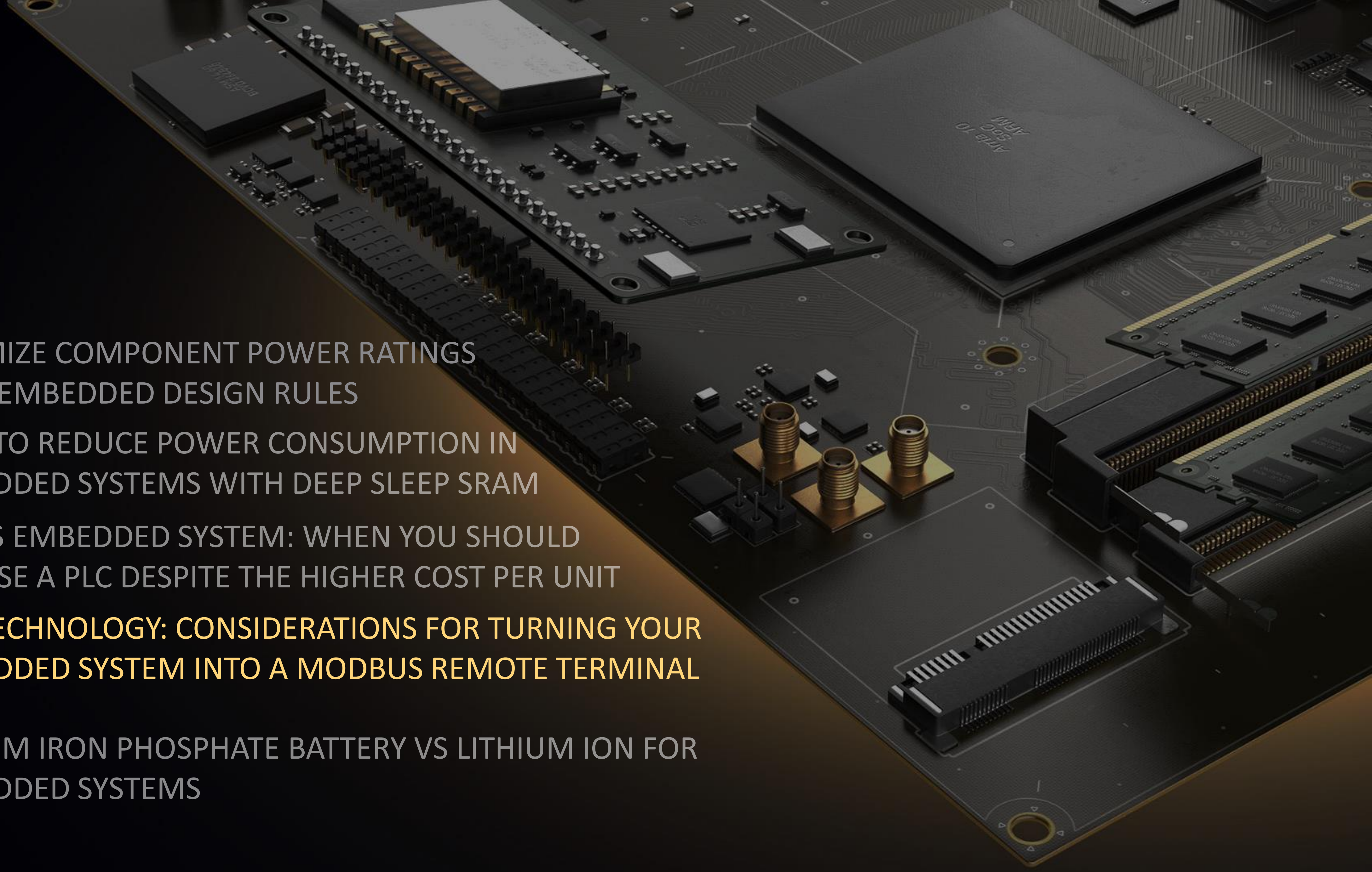
When to Use PLC Despite Higher Cost

Cost is always a consideration, but it is a mistake to automatically prefer a localized embedded system for cost alone. There are times when it is best to stick with using a PLC despite the fact that it can significantly increase your expenses. Here are some factors that make the cost of PLCs worthwhile:

- **Volume:** Unless you're working on a project that requires hundreds of PLCs with the same setup, it can be more cost effective to purchase a commercial off the shelf PLC. The costs in time of developing a localized embedded system and manufacturing its setup might not be worthwhile.
- **Certification:** If the cost of certifying your own embedded system outweighs the cost of the PLC, then it's better to stick with the latter option.
- **Support:** PLCs often come with support from a reputable company. If you're outsourcing the manufacturing of your localized embedded system to a contractor, make sure that you'll receive a decent amount of technical support.
- **Reliability:** If your intended application involves your control systems operating in harsh electrical and physical environments, it is often better to go with a PLC that is already proven to work in these environments.

See original content here





1. OPTIMIZE COMPONENT POWER RATINGS WITH EMBEDDED DESIGN RULES
2. HOW TO REDUCE POWER CONSUMPTION IN EMBEDDED SYSTEMS WITH DEEP SLEEP SRAM
3. PLC VS EMBEDDED SYSTEM: WHEN YOU SHOULD CHOOSE A PLC DESPITE THE HIGHER COST PER UNIT
4. PCB TECHNOLOGY: CONSIDERATIONS FOR TURNING YOUR EMBEDDED SYSTEM INTO A MODBUS REMOTE TERMINAL UNIT
5. LITHIUM IRON PHOSPHATE BATTERY VS LITHIUM ION FOR EMBEDDED SYSTEMS

4. PCB TECHNOLOGY: CONSIDERATIONS FOR TURNING YOUR EMBEDDED SYSTEM INTO A MODBUS REMOTE TERMINAL UNIT

What is a Modbus RTU?

Modbus is a communication protocol that is commonly used in industrial applications in a master-slave configuration. This protocol was developed in 1979 by Modicon and has been reliably used across automation, instrumentation and process control industries for more than three decades. The Modbus protocol involves a simple process of message request by a master followed by a response by the slave.

- In the context of industrial applications, an RTU is a device capable of executing processes and monitoring input data from sensors. They are advanced versions of Programmable Logic Controllers (PLC).
- RTUs are able to function on their own but they are often used as a standard communication interface that enables data transfer or remote monitoring back to a computer or touch screens devices.
- In that case, a Modbus RTU uses the Modbus protocol to transfer and receive information from other connected devices.

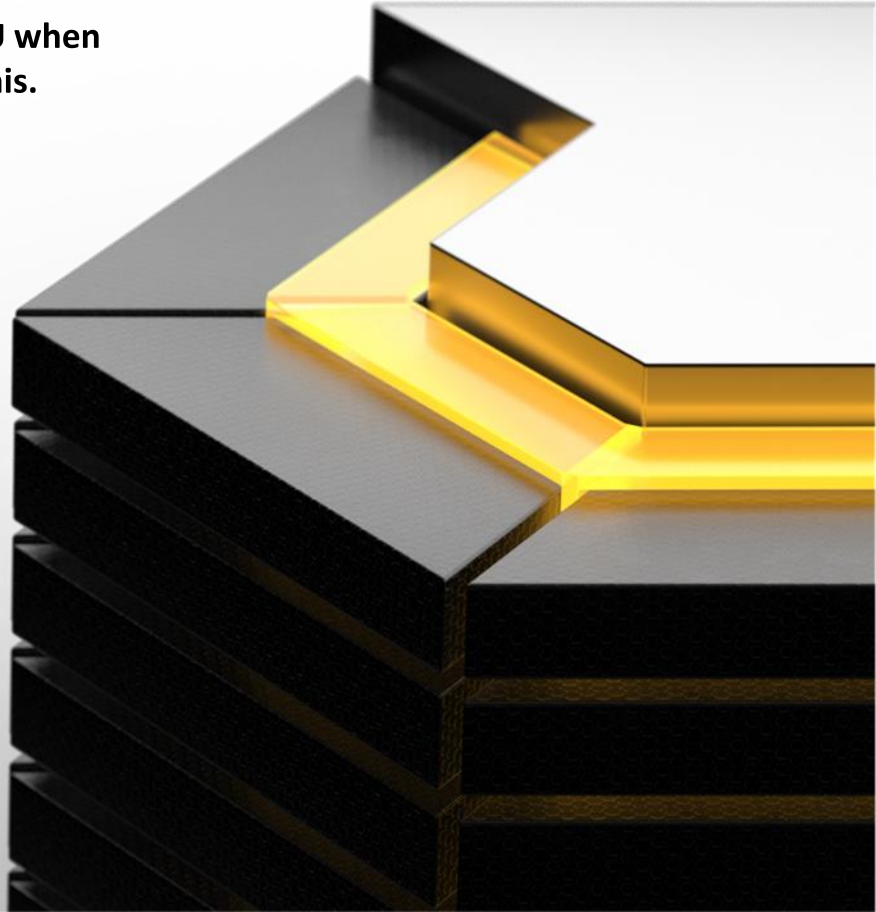
Altium.

Turning Your Embedded System into a Modbus RTU

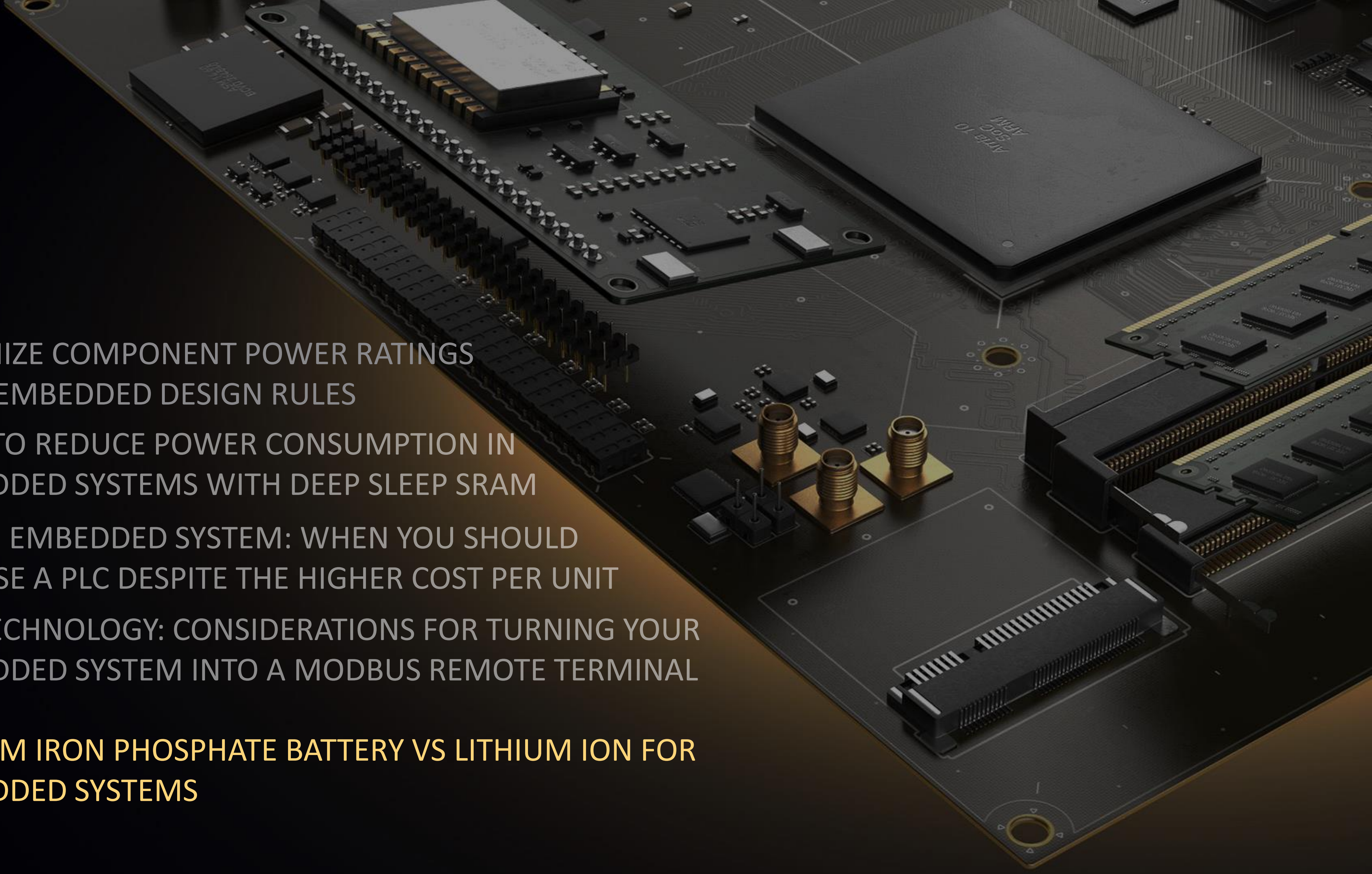
From a cost perspective, it's beneficial to turn your embedded system into a functional Modbus RTU when your application requires multiple Modbus RTUs. Here are some aspects to consider before doing this.

- **Communications topology:** You'll need to determine if your embedded system needs to adapt a Modbus master or slave setup.
- **Communications interface:** Serial communication is the popular interface for Modbus communication. An underlying interface like RS232, RS422, and RS485 are commonly used to connect Modbus devices. Modbus TCP/IP, on the other hand, operates on the TCP/IP protocol and uses IP address to identify the devices.
- **Modbus protocol:** There are two types of Modbus messaging structures, Modbus ASCII and Modbus RTU. Both share the same packet frame. However, Modbus ASCII uses 2 bytes of American Standard Code for Information Interchange (ASCII) characters in a field, and the Modbus RTU uses a single byte of binary value.
- **Memory allocation:** The Modbus protocol basically reads and writes data to register with addresses that follow a certain convention. A critical design process for the embedded system is mapping these registers to physical memory, which is likely to be different from the stated address in the Modbus protocol.

See original content here

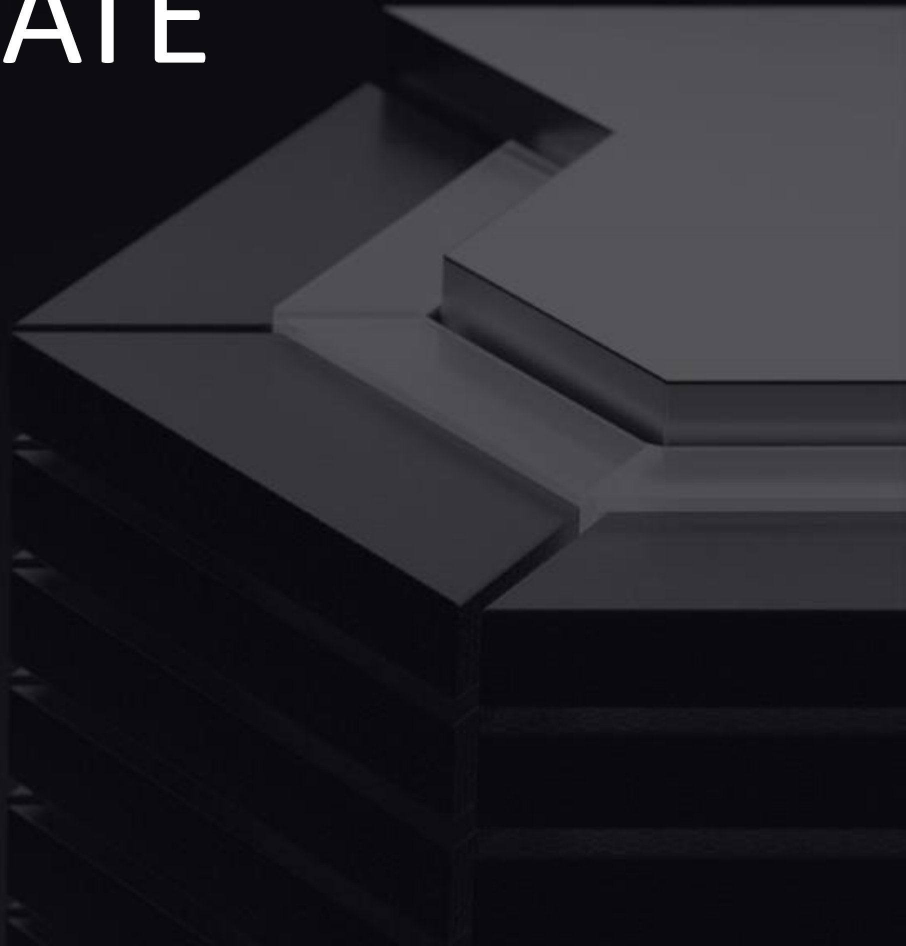


Altium.



1. OPTIMIZE COMPONENT POWER RATINGS WITH EMBEDDED DESIGN RULES
2. HOW TO REDUCE POWER CONSUMPTION IN EMBEDDED SYSTEMS WITH DEEP SLEEP SRAM
3. PLC VS EMBEDDED SYSTEM: WHEN YOU SHOULD CHOOSE A PLC DESPITE THE HIGHER COST PER UNIT
4. PCB TECHNOLOGY: CONSIDERATIONS FOR TURNING YOUR EMBEDDED SYSTEM INTO A MODBUS REMOTE TERMINAL UNIT
5. LITHIUM IRON PHOSPHATE BATTERY VS LITHIUM ION FOR EMBEDDED SYSTEMS

5. LITHIUM IRON PHOSPHATE BATTERY VS LITHIUM ION FOR EMBEDDED SYSTEMS



Li-Ion

- **Voltages:** 3.6 V nominal, ranging from 3.0 V- 4.2 V.
- **Specific Energy:** 150-200 Wh/kg
- **Charge Rate:** 0.7 C - 1 C, charging above 1 C will damage battery
- **Discharge Rate:** 1 C. You may be unfamiliar with the “C” rate. It means that if a battery is rated for 2400 mAh it can discharge with a maximum current of 2.4 A without being damaged.
- **Life Cycle:** 500-1,000 cycles. The life cycle depends heavily on operating temperature and depth of discharge (DoD).
- **Thermal Runaway:** 150°C, this is what causes Li-Ion batteries to explode.
- **Charging Temperature Range:** 0-40°C
- **Discharging Temperature Range:** -25-60°C

Altium.

Li-Phosphate

- **Voltages:** 3.2 or 3.3 V nominal, ranging from 2.5-3.65 V.
- **Specific Energy:** 90-120 Wh/kg
- **Charge Rate:** 1 C
- **Discharge Rate:** 1-25 C, possibly with 40 A pulses.
- **Life Cycle:** 1,000-10,000 cycles, highly temperature dependent.
- **Thermal Runaway:** 270°C
- **Charging Temperature Range:** 0-45°C
- **Discharging Temperature Range:** -25-60°C

Comparison

Here's how Li-Ion and Li-Phosphate compare on safety, discharge, and capacity:

- All in all, Li-phosphate batteries are much safer than Li-Ion batteries.
- Li-Ion's low discharge current rate of 1 C pales in comparison to Li-phosphates which can be up to 25 C.
- Li-Ion batteries can store up to twice as much energy per kg over Li-phosphate. That makes a big difference when you're trying to hit strict space targets.

See original content here

Altium®

Thanks for your attention!