*Application Note*

# *Industrial Functional Safety PLC Architecture Implementation based on TI MCU/MPU*

*Zekun Bai, Linjun Meng, Donna Xu*

**ABSTRACT**

This application note introduces how to design functional safety Programmable Logic Controller (PLC) systems compliant with IEC 61508 and ISO 13849-1 standards using Texas Instruments (TI) Sitara microcontrollers (MCU) and microprocessors (MPU). The document covers the basic concepts of functional safety, system safety goal decomposition methods, TI chip safety architecture characteristics, and recommended PLC system design approaches. Through modular design principles and flexible redundant architectures, design engineers can achieve SIL 3/CAT 3 level functional safety requirements with reduced cost and complexity. Target Audience are Industrial control system design engineers, functional safety architects, system integrators

## Table of Contents



## Trademarks

# 1 Functional Safety Fundamentals

## 1.1 What is Functional Safety

Functional safety refers to the ability to prevent or mitigate hazardous events through the correct operation of electrical, electronic, or programmable electronic safety-related systems. In industrial applications, equipment and systems may generate hazards under specific fault conditions. The objective of functional safety is to reduce the probability of such dangerous occurrences to an acceptable level.

## 1.2 Causality Chain from Fault to Harm

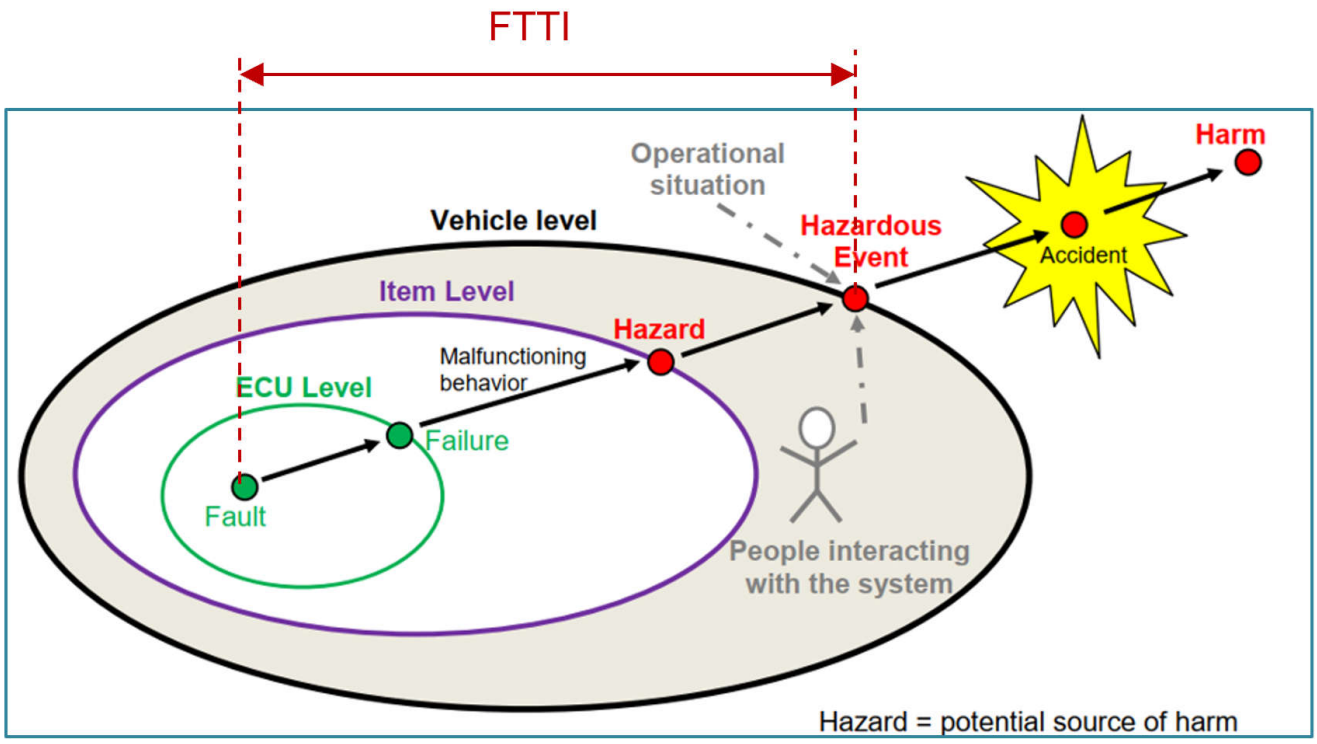


**Figure 1-1. FTTI Visualization**

- Level 1: Hardware/software fault (Fault) → CPU computation error, memory corruption, timing fault, etc.
- Level 2: Fault leading to failure (Failure) → abnormal system function, incorrect output, unresponsiveness
- Level 3: Failure triggering hazardous event (Hazardous Event) → unexpected equipment action, control failure, sensor blindness
- Level 4: Hazardous event causing personal injury (Harm) → personnel injury, equipment damage, production stoppage

According to the IEC 61508 standard definition, the complete process from fault to personal injury includes:

**Fault** → Latent Fault Metric

- System imperfections, such as hardware defects, manufacturing process deviations, software defects
- Random failures: caused by operational stresses (temperature, voltage) and cumulative usage time
- Systematic failures: caused by inadequate design or improper process

**Failure** → Dangerous Failure vs Safe Failure

- When a fault is not detected or detected too late, it leads to loss of system function
- Dangerous Failure: a failure that may lead to loss of safety function
- Safe Failure: a failure that leads the system to enter a safe state

**Hazardous Event** → Foreseeable but unavoidable vs Foreseeable and avoidable

- Failure ultimately transforms into an event that may harm personnel or equipment
- Example: unexpected robot movement, elevator brake failure, unexpected industrial equipment startup

**Personal Injury (Harm)** → Classification of injury severity

- If a hazardous event is not prevented in time, actual injury will result

## 1.3 Core Problems Addressed by Functional Safety

The objective of functional safety design is to establish multiple defense lines in this causality chain:

- **Fault Detection**: detect the presence of faults as quickly as possible
- **Fault Isolation**: isolate fault effects and prevent propagation to critical functions
- **Safety Response**: trigger safety functions when faults lead to failures
- **Redundant Design**: single faults do not lead to loss of safety function
- **Diagnosticability**: faults are detected promptly and can be identified by the system

# 2 Safety Standards and Grade Classification

## 2.1 Major Safety Standard Systems

Industrial and automotive fields employ different but interrelated safety.

**Industrial Application Base Standards**:

- **IEC 61508** - General safety standard applicable to all electrical/electronic/programmable electronic safety-related systems
- **IEC 62061** - Safety of machinery - safety-related electrical/electronic/programmable control systems
- **ISO 13849-1** - Safety of machinery - safety-related parts of control systems (using performance level scheme)

**Domain-Specific Application Standards**:

- **IEC 61800-5-2** - Adjustable speed electrical power drive systems - Part 5-2: Functional safety requirements
- **ISO 10218** - Robots and robotic devices - Safety requirements for industrial robots
- **IEC 61496** - Safety of machinery - Electro-sensitive protective equipment

**Industrial Communication Safety Standards**:

- **IEC 61784-3** - Industrial communication networks - Functional safety
  - Defines FSoE (Functional Safety over Ethernet) protocol
  - Specifies safety extensions for EtherCAT, PROFINET, EtherNet/IP and other protocols
  - Defines communication layer fault detection and recovery mechanisms
  - Communication fault PFH budget typically does not exceed 1% of total system PFH

**Automotive Applications**:

- **ISO 26262** - Road vehicles - Functional safety of electrical/electronic/programmable electronic safety-related systems

## 2.2 Functional Safety in Industrial Communication (FSoE)

The widespread adoption of Industrial Ethernet brings new challenges: the need to achieve determinism and safety on standard networks.

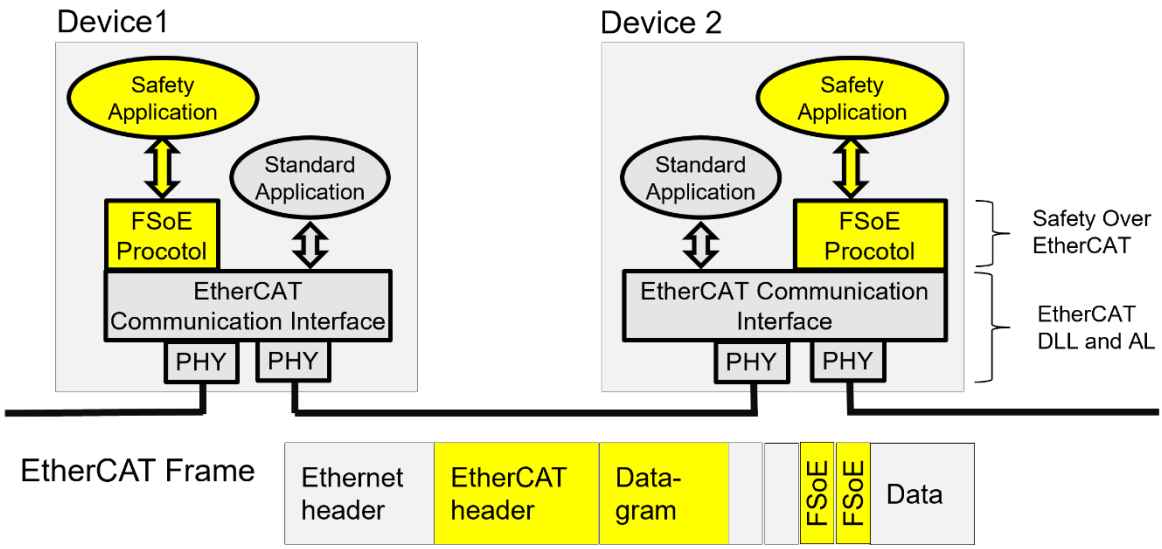


**Figure 2-1. FSoE protocol hierarchy structure**

- Application layer: safety-related data (position, velocity, fault codes, etc.)
- FSoE Protocol layer:
  - Sequence number detection (detects duplication, loss, out-of-order)
  - CRC verification (detects data corruption)

  - Connection ID (detects incorrect forwarding)
  - Watchdog timer (detects timeout delays)
- Physical layer: standard Ethernet MAC/PHY

FSoE does not modify the underlying physical layer, it enhances through application layer protocol.

*Figure 3 Differences between FSoE and Standard PLC Communication*

| Characteristic | Standard Ethernet/EtherCAT | FSoE Enhancement |
|---|---|---|
| Base protocol | EtherCAT standard | Same |
| Safety data | No special processing | With CRC and sequence number |
| Fault detection | Packet loss detection | Complete FSoE diagnostics |
| Delay detection | None | Watchdog monitoring |
| Reliability | Medium (network dependent) | High (automatic fault transfer) |
| Application complexity | Low | Medium (protocol stack processing) |

## 2.3 Safety Grade Indicator System

### *2.3.1 IEC 61508 - Safety Integrity Level (SIL)*

IEC 61508 defines 4 safety integrity levels based on dangerous failure probability indicators.

*Figure 4 IEC 61508 key indicator*

| IEC61508 | | | | | | |
|---|---|---|---|---|---|---|
| Hardware Fault Tolerance | | | | | | SFF |
| **0** | 1 | 2 | **0** | 1 | 2 | |
| - | SIL 1 | SIL 2 | SIL 1 | SIL 2 | SIL 3 | **< 60%** |
| SIL 1 | SIL 2 | SIL 3 | SIL 2 | SIL 3 | SIL 4 | 60% to < 90% |
| SIL 2 | SIL 3 | SIL 4 | SIL 3 | SIL 4 | SIL 4 | 90% to < 99% |
| SIL 3 | SIL 4 | SIL 4 | SIL 4 | SIL 4 | SIL 4 | ≤ 99% |
| Type A | | | Type B | | | |

IEC 61508 does not demand a certain HFT, systems can be 1oo1, 1oo2

- Type A subsystems: the failure modes well defined, where the behavior under fault conditions is determined and there is enough failure data to claim that the failure rates are met.
- Type B subsystems: more complex where the failure modes are not fully well defined, the fault conditions cannot be completely determined and there is no enough data to support that the failure rates are met.

**Key Indicator Explanations**:

- **PFH** (Probability of Failure per Hour): number of failures in 10^9 hours of operation (FIT unit)
- **SFF** (Safe Failure Fraction): proportion of non-dangerous failures
- **HFT** (Hardware Fault Tolerance): hardware fault tolerance capability

**Communication PFH Budget**: According to IEC 61784-3, the communication channel PFH should not exceed 1% of the total system PFH. For example:

- If system target is SIL 3 (PFH ≤ 10 FIT)
- Communication is allowed PFH ≤ 0.1 FIT (1% of 10 FIT)
- This imposes very high requirements on communication reliability

### 2.3.2 ISO 13849-1 - Performance Level (PL) and Category (CAT)

ISO 13849-1 uses a different classification approach emphasizing the fault resistance of the architecture.

*Figure 5 Performance levels range from a (lowest) to e (highest)*

| **ISO13849** | | | | |
|---|---|---|---|---|
| DC | Category | | | |
| | **1** | **2** | **3** | **4** |
| None | | | | |
| Low | c | c | d | |
| Medium | | d | e | |
| High | | | | e |

ISO13849 category specifies resistance to faults, similar to a HFT(hardware fault tolerance).

- Cat. 1: Single channel + well-tried components
- Cat. 2: Single channel + test equipment
- Cat. 3: Dual channel + diagnostics, no accumulation of faults
- Cat. 4: Dual channel + diagnostics, accumulation of faults, higher demand for diagnostics

Performance levels range from a (lowest) to e (highest). Key mappings:

- **PL e ≈ SIL 3**: requires HFT=1 with high diagnostic coverage
- **PL d ≈ SIL 2**: requires HFT=1 with medium diagnostic coverage

# 3 System Safety Goal Decomposition

## 3.1 HARA Process and Safety Goal Definition

System safety goal decomposition starts from HARA (Hazard Analysis and Risk Assessment)

**Step 1: Identify all potential hazards**

- Identify all undesired events that the system may cause
- Example (industrial robot): unexpected robot movement, brake failure, overspeed
- Example (PLC control system): unexpected output activation, control failure due to communication interruption, sensor misreading

**Step 2: Risk assessment and classification**

For industrial applications (IEC 61508), risk assessment is based on:

- **Consequence (Consequence)**: CA (minor) → CB → CC → CD (fatal)
- **Frequency and exposure time (Frequency)**: FA (rare) → FB (frequent)
- **Avoidability (Possibility of avoiding)**: PA (avoidable) → PB (difficult to avoid)

**Step 3: Define safety goals**

Safety goals should clearly state:

- The hazardous event to be prevented or mitigated
- Corresponding safety integrity level (SIL or ASIL)
- Acceptable failure rate (PFH)
- Diagnostic coverage and fault tolerance requirements

**Example**:

Safety Goal 1: "Prevent unexpected PLC control signal activation"

- SIL level: SIL 2

- PFH target: ≤100 FIT

- Diagnostic coverage: ≥90%

- Fault tolerance requirement: HFT=1 (single fault does not cause activation)

Safety Goal 2: "Ensure communication channel reliability"

- SIL level: SIL 2

- PFH target: ≤1 FIT (communication budget)

- Diagnostic coverage: ≥99% (FSoE mechanism)

- Fault tolerance requirement: DLR ring network redundancy

## 3.2 Safety Goal Decomposition and ASIL/SIL Assignment

Top level: System safety goal "PLC control system safety" - SIL 3

Second level decomposed into:

- Sub-goal A: "I/O module fault does not cause dangerous output" - SIL 3
- Sub-goal B: "Communication fault is detected and handled" - SIL 2
- Sub-goal C: "Processor fault is isolated" - SIL 2

Third level continues decomposition into specific requirements.

**ASIL Decomposition Principles (ISO 26262)**

ASIL decomposition allows distributing a high-level safety goal into multiple lower-level sub-goals. If independence and diversity requirements are met, the system can still achieve the original ASIL level.

*Figure 6 Decomposition Rules:*

| Original ASIL | Allowed Decomposition Combinations |
|---|---|
| ASIL-D | D(D) + QM(D) or C(D) + A(D) or B(D) + B(D) |
| ASIL-C | C(C) + QM(C) or B(C) + A(C) |
| ASIL-B | B(B) + QM(B) or A(B) + A(B) |
| ASIL-A | A(A) + QM(A) |

**Key Requirements**:

1. Independence (Independence)
   - Decomposed sub-goals should be mutually independent
   - Failure of one sub-goal should not cause failure of another
   - Example: using different hardware, different power supplies, different clocks
2. Diversity (Diversity)
   - Adopt different technologies or implementation methods
   - Example: using different processor vendors, different programming languages, different algorithms
3. Traceability (Traceability)
   - Clear documentation of the decomposition process and basis
   - Explain how each sub-goal combines to satisfy the original goal
   - Maintain complete requirements traceability matrix
4. Justification (Justification)
   - Need to prove that the decomposed architecture actually achieves the original ASIL
   - Usually proven through Fault Tree Analysis (FTA) or other quantitative methods

**SIL Synthesis Principles (IEC 61508)**

Corresponding to ASIL decomposition, IEC 61508 defines SIL synthesis methods. SIL synthesis essentially allows the synthesis (or combining) of two redundant elements with a systematic capability of N to have a systematic capability of N + 1, as long as N is less than or equal to SIL 3. The rules that govern SIL synthesis according to IEC 61508 are:

- SIL 2 + SIL 2 → SIL 3
- SIL 1 + SIL 1 → SIL 2

IEC 61508 does not allow recursive (multi-level) synthesis, but ISO 26262 does. For example, SIL synthesis (according to IEC 61508) does not permit either of the following:

- SIL 2 (in support of SIL 3) + [SIL 1 (in support of SIL 2) + SIL 1 (in support of SIL 2)] → SIL 3.
- SIL 2 (in support of SIL 3) + SIL 1 (in support of SIL 3) → SIL 3.

In contrast, ISO 26262 does allow multilevel decomposition.

IEC 61508 also mandates a two-channel implementation for SIL 4 systems (the hardware fault tolerance has to be >0 for a SIL 4 function). Otherwise, ASIL decomposition and SIL synthesis are equivalent constructs in the ISO 26262 and IEC 61508 standards, respectively.

## 3.3 Concrete Application of System-Level Decomposition

**Example: SIL 3 Decomposition of Industrial PLC System**

**System-level safety goal: SIL 3.** Decomposed into three subsystems:

**Subsystem 1 - I/O Fault Isolation (SIL 2)**

- Digital input fault detection
- Digital output fault detection
- Cut off output when fault occurs
- Diagnostic coverage: ≥90%

**Subsystem 2 – Dual MCU Fault Detection (SIL 2)**

- Watchdog timeout detection
- Memory ECC checking
- Internal diagnostic self-test
- Diagnostic coverage: ≥90%

**Subsystem 3 - Communication Safety (SIL 2)**

- FSoE protocol fault detection
- Network ring redundancy (DLR)
- Message verification and timeout
- Diagnostic coverage: ≥99%

**Combination strategy:**

- Three SIL 2 subsystems through independent hardware and detection mechanisms
- System-level diagnostic coverage exceeds 99%
- Single-point fault does not lead to loss of safety function (HFT≥1)
- Result: entire system achieves SIL 3

## 3.4 Role and Responsibility Division

In functional safety projects, clear role division among parties is critical:

**Customer/System Integrator Responsibilities**:

- Conduct HARA and define system-level safety goals
- Formulate safety concept and decompose safety goals to subsystems
- Select appropriate TI components
- Develop safety-related software
- Perform system-level verification and testing
- Prepare safety case documentation

**TI Responsibilities**:

- Develop functional safety MCU/MPU using certified development processes
- Provide component-level safety analysis (FMEDA)
- Provide Safety Manual and diagnostic library (SDL)
- Act as SEooC (Safety Element out of Context)
- Not responsible for system integration, application software development, and system evaluation

**Independent Assessment Body (e.g., TÜV SÜD) Responsibilities**:

- Independently assess and certify TI components
- Review customer system safety case
- Issue functional safety certification certificates

# 4 TI Chip Safety Architecture

## 4.1 MCU-Level Safety Architecture

Below shows TI Functional Safety MCU Safety Architectures with three common architectures.

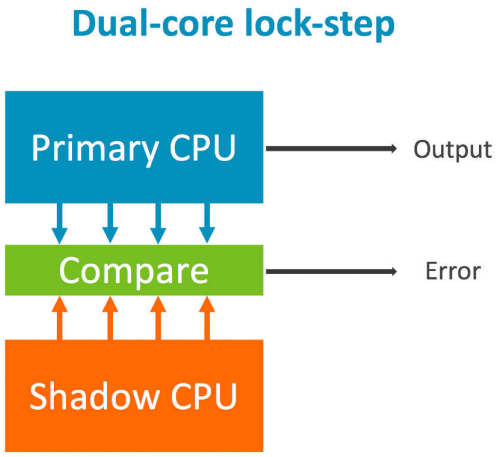


**Figure 4-1. Dual Core Lock Step**

**Dual Core Lock Step (DCLS)**: Two identical CPU cores running in parallel, hardware comparator detecting mismatches, immediate interrupt on detection.

Characteristics:

- Two CPU cores execute identical instructions with synchronized timing
- Hardware comparator continuously compares computation results
- Any mismatch immediately triggers error handling and system halt
- Fault detection latency: Single cycle (minimal)

Advantages:

- Low cost, single-chip implementation
- Extremely fast fault detection
- Simple architecture

Disadvantages:

- Cannot detect common-cause failures (e.g., shared clock fault)
- No fault tolerance, only detection capability
- If one core fails, entire system stops

Applications: SIL 2 HFT=0 systems

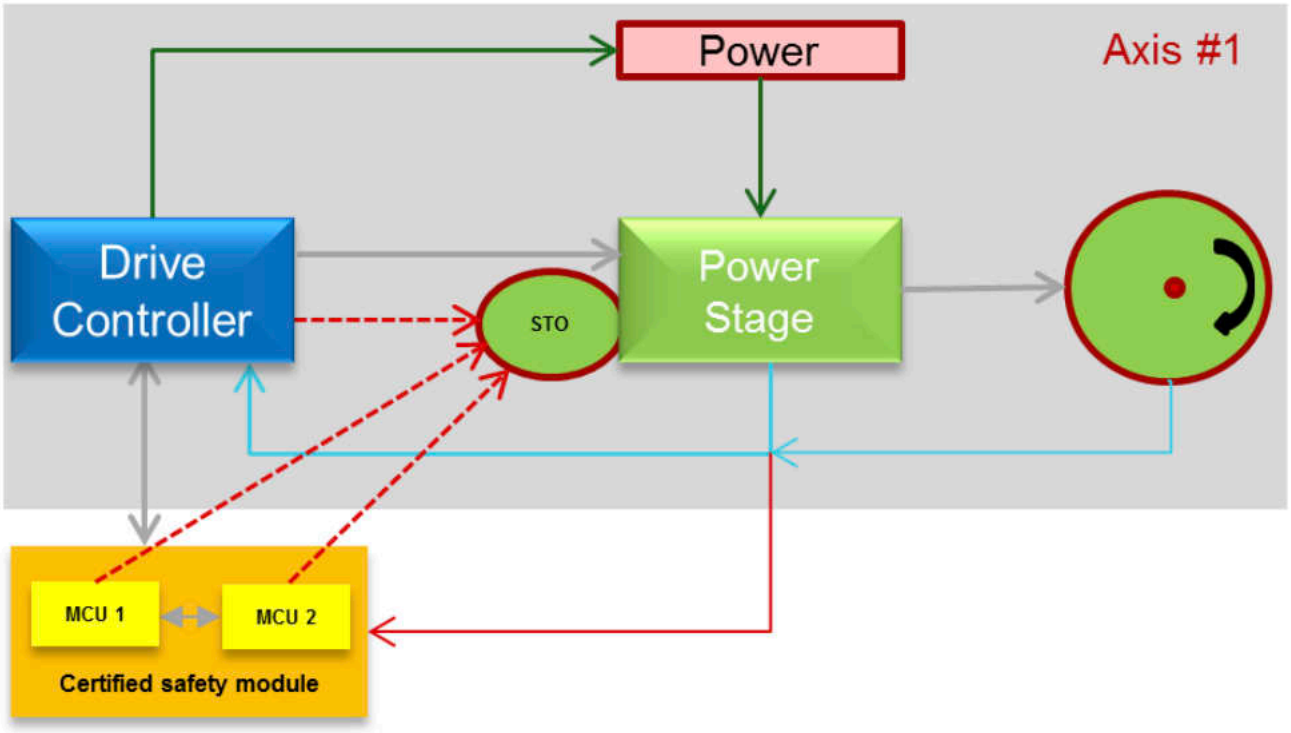


**Figure 4-2. Homogeneous Redundancy**

**Homogeneous Redundancy (HFT=1)**: Two identical MCUs executing independent tasks, cross-checking via SPI/I2C, eliminating common-cause failures(CCF).

Characteristics:

- Two identical MCUs run parallel tasks independently
- Cross-verification through I2C, SPI, or other isolated interfaces
- Periodic result comparison, discrepancy triggers safety action
- Fault detection latency: 10-100ms (configurable)

Advantages:

- Can detect most common-cause failures through independent design:
- Hardware fault tolerance: system continues operation after single MCU failure
- Both cores can be monitored and managed independently
- Supports complex safety logic

Disadvantages:

- Requires two MCUs, increased cost
- Cross-check communication adds latency
- More complex integration

Applications: SIL 3 HFT=1 systems

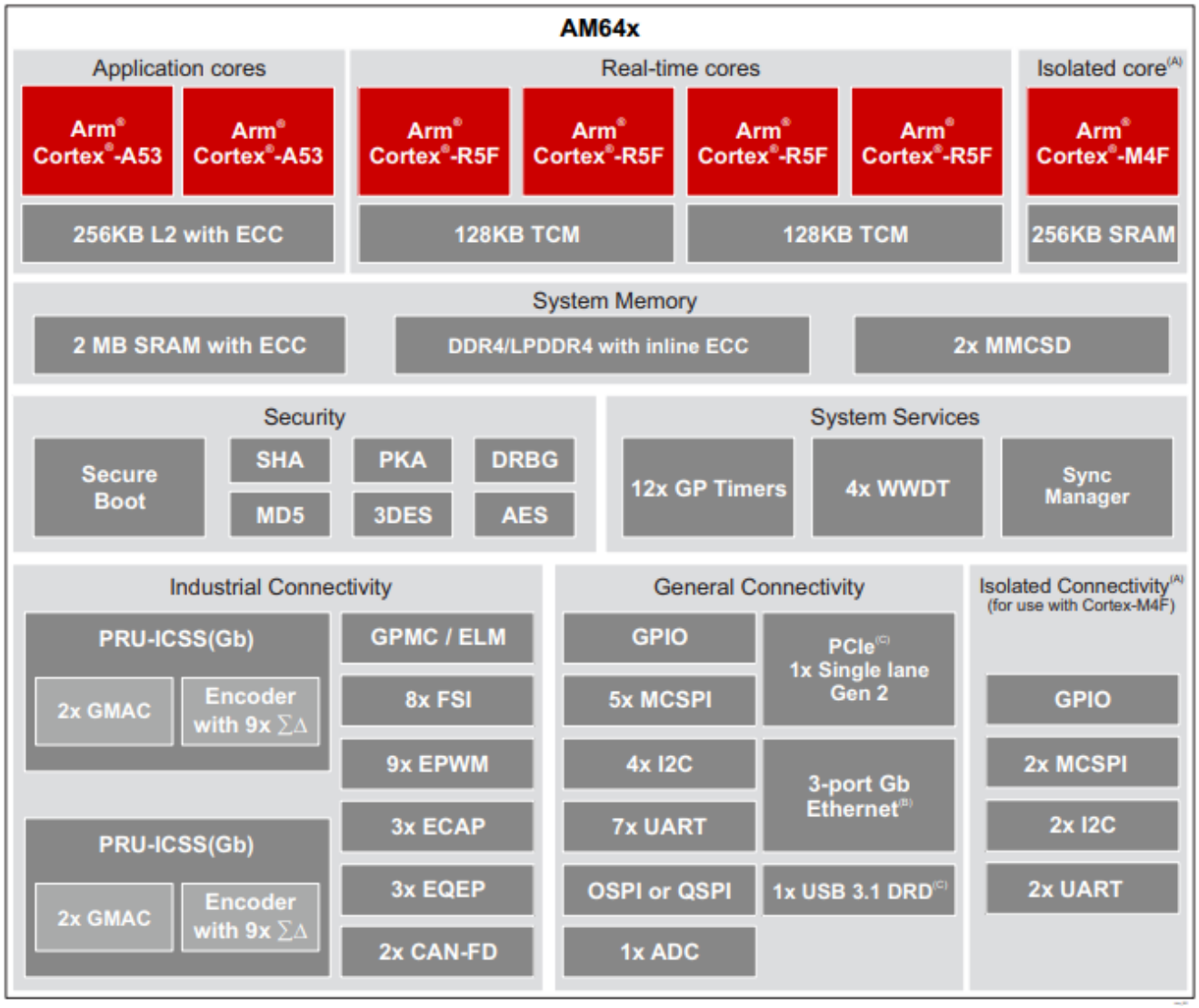


**Figure 4-3. AM64 one page**

**Heterogeneous Redundancy (HFT=0)**: Different MCU types or diversified software implementation, detecting system-level failures.

Characteristics:

- Uses processor architectures with MCU + MPU combination.
- Implements different software implementations on different cores, different SIL requirements. Use high SIL core to monitor low SIL core tasks.
- Hardware isolation between cores, including periphery, Power, Memory.

Advantages:

- Can detect certain system-level design defects
- Maximum redundancy and fault tolerance
- Low cost among all approaches

Disadvantages:

- High development complexity
- Requires deep expertise in multiple platforms

Applications: SIL 2/3 or special ASIL-D requirements

## 4.2 Integrated Safety Mechanisms/Technology in TI MCU/MPU

TI provides certified software diagnostic libraries(SDL) for each functional safety MCU/MPU.

**SDL Library Typical Contents**:

- CPU core and bus diagnostic routines
- Memory test algorithms
- Peripheral Built-In Self-Test (P-BIST)
- Logic Built-In Self-Test (L-BIST)
- Clock, voltage, temperature monitoring
- Interrupt and exception handling

**TÜV Certification Note**: SDL has been independently assessed by evaluation bodies (such as TÜV SÜD) and can satisfy specific SIL/ASIL levels.

Also TI MCU/MPU can achieve Hardware SIL2, systematic SIL3 capability, which integrate much safety measures as below.

### *4.2.1 Freedom From Interference (FFI) Design*

FFI is the key technology when running different safety-level tasks on a single SoC. It can eliminate

cascading failures and dependencies between different ASIL/SIL level components, anc ensure that low-level component failures do not propagate to high-level safety islands.

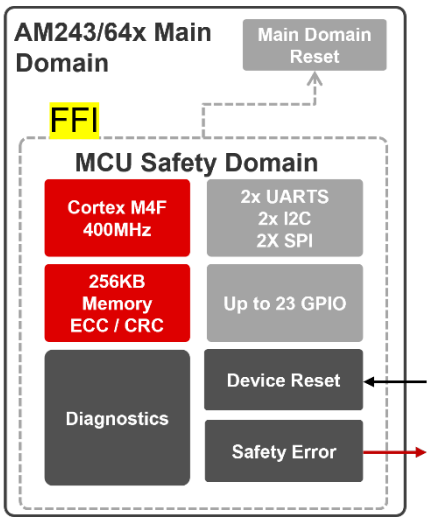


**Figure 4-4. FFI Implementation in TI MCU/MPU**

FFI external side: Non-Safe Domain - Standard applications, communication, non-critical tasks

FFI internal side: Safety Island - Safety-critical tasks, independent R5F or M4F core

FFI Isolation mechanisms:

- Firewall (Firewall) - Controlling bus access
- Timeout Gaskets (Timeout Gaskets) - Protecting communication paths
- Independent clock/power/reset
- Dedicated interrupt control

Bidirectional arrows: Controlled interfaces (SPI/I2C) for necessary communication

Status indicator: Safety island can independently monitor and restart non-safe domain.

**FFI Benefits**

- **Cost Optimization**: Integrate multiple safety-level functions in single MCU, reducing total BOM
- **Integration Level**: No need for separate external safety MCU
- **System Complexity**: Reduce integration difficulty and test effort

### 4.2.2 Memory Protection and ECC Technology

Data path in MCU/MPU: Original data → ECC encoding → Storage → ECC verification → Correction/Detection

ECC is the core technology for protecting memory from random failures.

- Single-bit error (SBE) detection and correction process
- Multi-bit error (DBE) detection process
- Hamming code or SECDED code parity bit generation.

Take AM243/AM64 as example, see below ECC coverage.

*Figure 11 ECC Coverage in Key Memory*

| Memory Type | ECC Type | Range | Notes |
|---|---|---|---|
| L1 I-Cache | Parity | A53 cores | Detection only |
| L1 D-Cache | ECC SECDED | A53 cores | Single-bit correction capable |
| L2 Cache | ECC SECDED | Shared 512KB | Single-bit correction capable |
| R5F SRAM | ECC SECDED | 64KB per core | All R5F memory covered |
| MCUSS SRAM | ECC SECDED | 512KB shared | Safety island memory |
| DDR EMIF | ECC optional | Main memory | Customer configurable |

### 4.2.3 Other Integrated Safety Mechanisms

*Figure 12 Other mechanisms in TI MCU/MPU*

| Safety Mechanism | Description | Detected Content | Coverage |
|---|---|---|---|
| **ECC/EDAC** | Error Detection and Correction | Single-bit flips in memory | High |
| **Memory Protection Unit (MPU)** | Prevents unauthorized access | Stack overflow, buffer overflow, illegal access | Medium |
| **Watchdog (WD)** | Processor hang detection | CPU dead loop, system stuck | High |
| **Dual Clock Comparator (DCC)** | Clock anomaly detection | Clock frequency deviation | High |
| **Error Signaling Module (ESM)** | Centralized fault management | Routes all diagnostic faults and responses | Very High |
| **Power-on Reset (POR/BOR)** | Power supply monitoring | Abnormal voltage conditions | Medium |

# 5 Functional Safety PLC Architecture Design

## 5.1 Necessity and Application Scenarios of Functional Safety PLC

### Why Functional Safety PLC is Needed

Modern industrial automation faces challenges：

1. Increasing Regulatory Safety Requirements

- EU Machinery Directive mandates functional safety
- ISO 10218 (robots) requires PL d / SIL 2
- IEC 61800-5-2 (variable drives) requires SIL assessment

Non-compliant products cannot enter market

2. Human-Robot Collaboration Becomes Trend

- Collaborative robots coexist with humans
- Any fault may cause personal injury
- Traditional PLC cannot provide adequate safety protection

3. System Complexity Increases

- Multiple independent subsystems need coordination
- Communication failures may cause cascading failures
- Reliable fault detection and recovery mechanisms needed

4. Availability and Maintainability Requirements

- Production stoppage cost is high (tens of thousands per minute)
- Rapid fault detection and automatic recovery needed
- Detailed diagnostic information required

## 5.2 Functional Safety PLC Architecture Design

Functional safety PLC key concepts are layering, isolation, and redundancy.

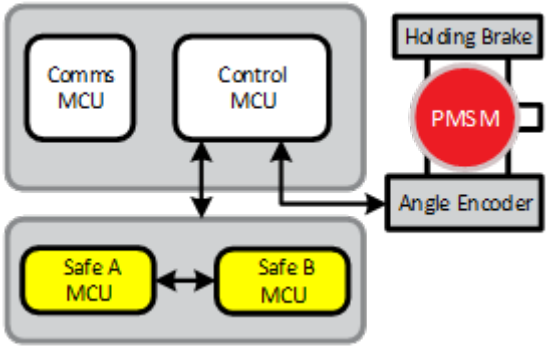


**Figure 5-1. Separate Safety MCU with Standard MCU System Concept**

### Control Processor (Central Processor)

- High-performance MPU (e.g., AM64x or AM243x)
- Run PLC scan cycles and user logic
- Handle multiple industrial communication protocols

### Safety Subsystem (Safety Subsystem)

- Two independent safety MCUs (AM243x or C2000)
- Separate SRAM, ROM and peripherals for each
- Cross-check via SPI interface
- Output: Fault diagnosis, status reporting
- Isolated relay driver or direct GPIO control

**Comm Redundancy Management (DLR / Ring Network)**

- AM243x/AM64x built-in DLR supervisor manager
- Maintain two independent communication paths
- PRU_ICSSG performs fast fault detection and transfer
- Ethernet switch-based redundancy
- Handled at industrial Ethernet link layer

**I/O Modules**

- Modular design
- Support various I/O types (DI/DO/AI/AO)
- Optional fault detection (high-end modules)

## 5.3 Design Implementation Cases

AM64/AM243 can support hardware SIL2, system SIL3. See details in AM243/AM64 datasheet. AM64 and AM243 are pin to pin compatible.

More importantly, AM64/AM243 integrate 2 * PRU_ICSSG sub-system can achieve Industry real-time communication via loading different firmware into PRU_ICSSG.

Each PRU_ICSSG includes:

- Two PRU cores (programmable logic)
- Gigabit Ethernet MAC (RGMII/SGMII)
- Real-time data buffer (DPRAM)

Below shows 2 * AM243 as FuSa MCU module, and use 1 another AM243 as comm module. Combined with the system concept in last chapter, give TI FuSa PLC solution.

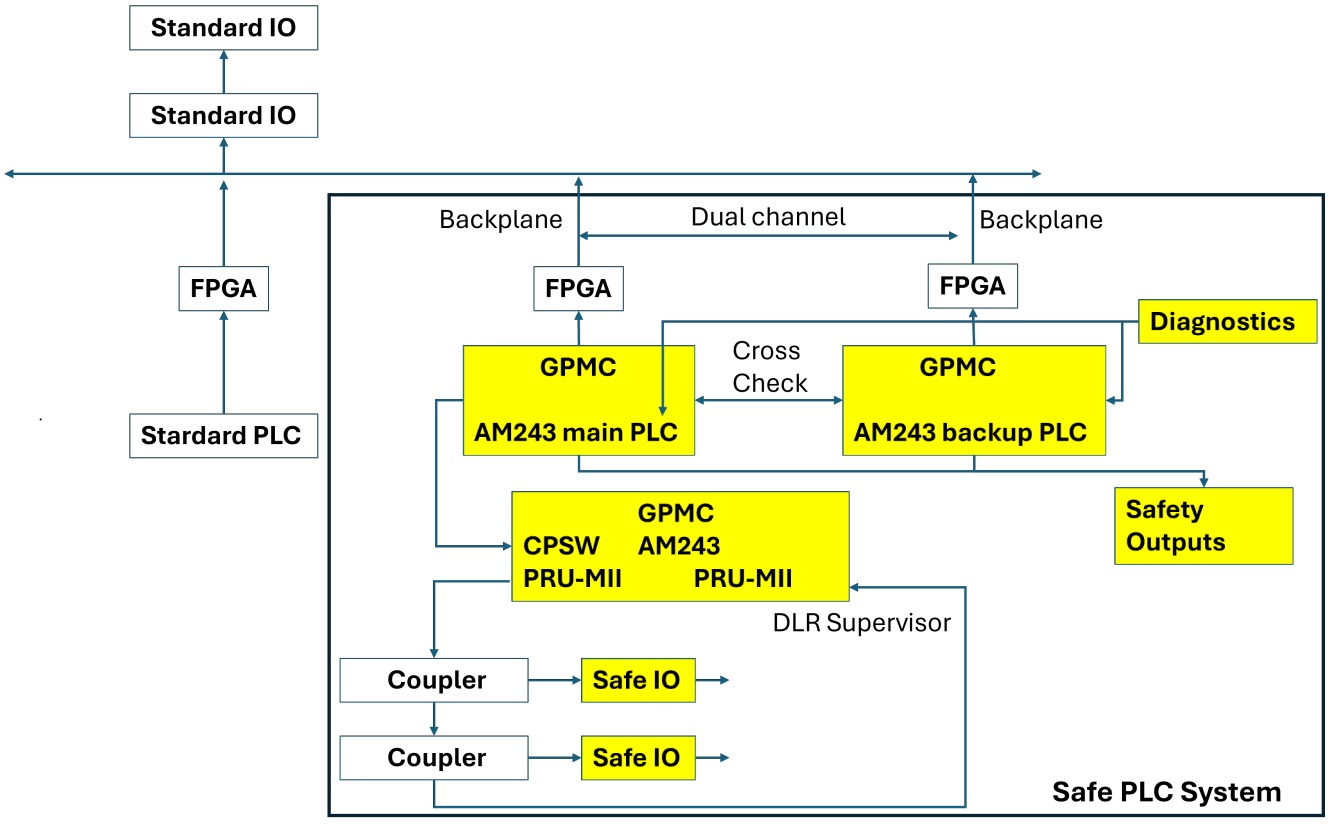


**Figure 5-2. Functional Safety PLC Architecture example**

Combined with standard PLC, 2 * AM243 serve as FuSa PLC system to achieve SIL3, HFT = 1, CAT 3. Use SPI or I2C to implement cross check between 2 * AM243.

Use one more AM243 running as comm module to manage communication, run DLR supervisor connecting with external coupler to expand safe IO numbers.

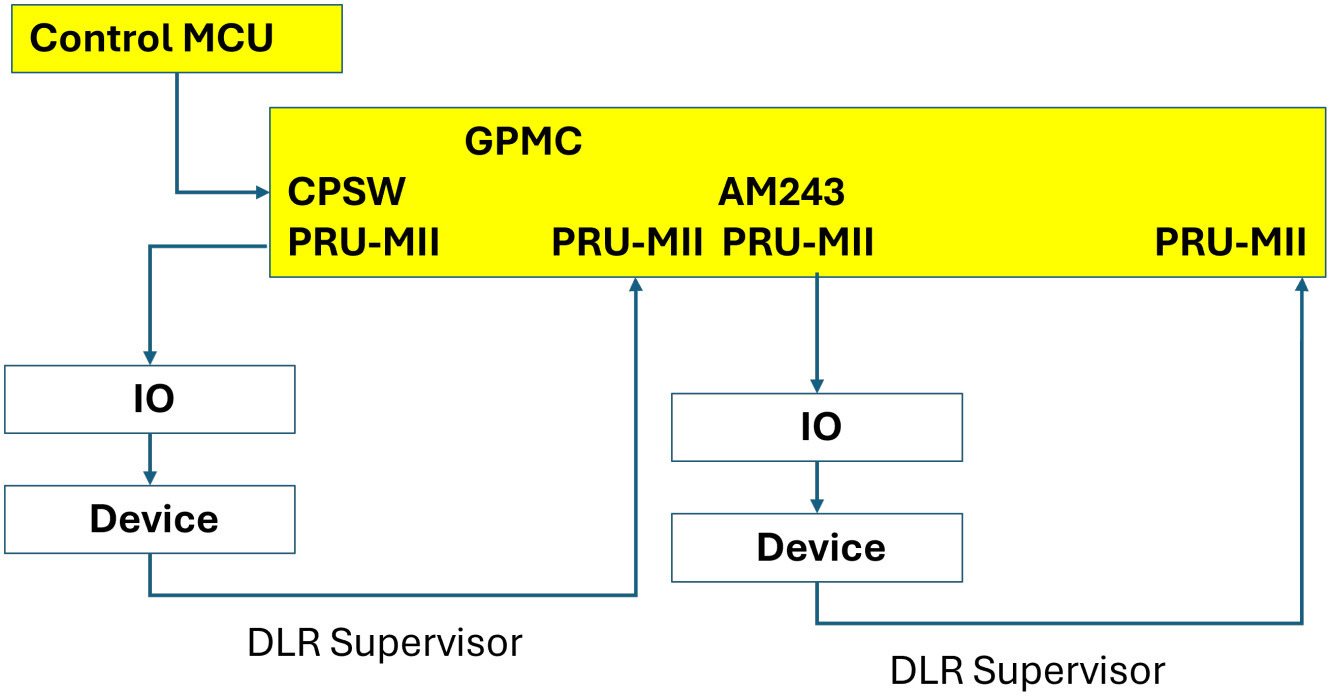


**Figure 5-3. Redundant Comm Module example**

AM243 integrate two PRU_ICSSG subsystems, and can support up to five MAC ports, four MAC ports in PRU_ICSSG, one MAC ports in CPSW.

- One PRU_ICSSG supports two MAC ports, DLR supervisor running on one R5F core.
- Another one PRU_ICSSG supports 2 MAC ports, DLR supervisor running on one R5F core.
- One MAC in CPSW, connecting with control MCU.

**Key Architecture Characteristics**

**Common Hardware Interface**:

- Backplane protocol identical across all series
- I/O module form factor standardized
- Power connectors standardized
- Allows mixing and matching different modules

Advantages:

- Customers need not learn multiple systems
- Upgrade cost minimal (mainly CPU card)
- Inventory management simplified

**Implemented Benefits**

1. Cost Optimization

- Use unified processor platform
- Share bottom-layer drivers and toolchain
- Different customer needs get different price points

2. Rapid Time to Market

- Reuse components and designs
- Shorten development cycle for each new product
- Accelerate functional safety certification (reference designs pre-certified)

3. Maintainability

- Unified codebase and documentation
- Easier technical support and updates
- Consistent user training content

4. Market Coverage

- Small devices use low-end configuration
- Medium devices use mid-range configuration
- Large and high-risk applications use high-end configuration
- Single product line covers entire market

5. Future Upgrade Capability

- When industrial standards evolve (new protocol support)
- Customers can get new functionality via firmware upgrade
- No hardware replacement needed

## 5.4 TI Functional Safety Design Resources

### System-level Support

- Reference Designs
  - Safe Torque Off (STO) reference design
  - Safety I/O module reference design
  - DLR redundancy management reference design
  - Pre-assessed and certified by TÜV

### Component-level Resources

- Functional Safety Manual (Safety Manual)
- FMEDA analysis reports
- Diagnostic capability checklist
- Application notes

### Software Support

- Software Diagnostic Library (SDL)
- Pre-certified source code (TÜV approved)
- Complete API documentation
- Example code and integration guide

### Toolchain Support

- SDK (Software Development Kit)
  - Including RTOS, drivers, protocol stacks
- SafeTI Compiler Qualification Kit
  - Meets ISO 26262/IEC 61508 compiler verification
- Debugging tools
- Code analysis tools

### Documentation and Guidance

- White papers and application notes
- Functional safety best practices
- Standard mapping guides
- Architecture design guidance]

# 6 Summary

This application note provides practical guidance for designing industrial Programmable Logic Controllers (PLCs) that comply with functional safety standards IEC 61508 and ISO 13849-1 using Texas Instruments (TI) microcontrollers and microprocessors. It addresses the complete safety system lifecycle from fundamental concepts through implementation and real case studies.

**Value for Customers**

**For System Integrators:**

- 40-60% reduction in development time through pre-certified reference designs
- Significantly shortened certification cycles using proven architectures
- Lower total cost of ownership through component reuse

**For Equipment Manufacturers:**

- Clear pathways to meet ISO 10218 (robots), IEC 61800-5-2 (drives), and domain standards
- Comprehensive fault detection and redundancy reduces liability risk
- Flexible architecture supporting simple to complex applications

**For End Users:**

- Systems achieving >99.5% availability through proper redundancy
- Comprehensive diagnostics enabling predictive maintenance
- Future-proof designs upgradeable through firmware changes

**Critical insight**

**Architecture Selection**:

- Simple safety applications use single safety MCU; complex systems use layered MCU+MPU approach; network-critical applications add DLR redundancy.

**Diagnostic Strategy**:

- Rather than implementing all diagnostics, strategically select ECC, watchdog, FSoE, and periodic self-tests to achieve target SIL/ASIL cost-effectively.

**Communication Reliability**:

- FSoE protocol maintains safety over standard Ethernet through application-layer mechanisms; DLR ensures <3ms network failover without compromising safety response time.

# 7 References

1. AM64x Sitara™ Processors data manual
2. AM243x Sitara™ Microcontrollers data manual

# IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you fully indemnify TI and its representatives against any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale, TI’s General Quality Guidelines, or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products. Unless TI explicitly designates a product as custom or customer-specified, TI products are standard, catalog, general purpose devices.

TI objects to and rejects any additional or different terms you may propose.



Last updated 10/2025