

# Hercules 设计学习

Hercules 是面向安全应用而设计的 MCU，可广泛应用于对安全有一定要求的汽车，交通运输，医疗，工业安全等领域。TMS570 USB 开发板和 RM48 USB 开发板是德州仪器公司推出的一款非常超值的针对 TMS570LS31x 系列和 RM48 系列的开发套件。

## 关于 Hercules TMDx570

TMDX570LS20SUSB 是一款 USB 套件，可用于快速评估 TMS570 MCU 的代码开发和性能。它包含经过 IEC61508 SIL3 认证的 TMS570LS20216（具有以锁步模式运行的双核 Cortex-R4F ARM CPU）、2MB 片上闪存、160KB RAM 和一组增强型外设模块。

USB 记忆棒包含集成 XDS100v2 仿真器，因此无需外部 JTAG 仿真器即可着手进行开发。该 USB 记忆棒还提供对某些关键外设、CAN 收发器和环境光传感器的访问（用于演示目的）。

TMS570 USB 套件还包含完整的软件开发环境，并且具有 CCStudio v4.1、Halcogen 外设驱动程序、闪存编程实用程序、HET 模拟器/调试器（含 Synapticad WaveViewer）、代码示例等等。



## Hercules 开发流程



使用HALCoGen，根据设定的配置生成代码

- 将代码导入到CCS V5
- 由用户根据需要完善代码功能
- 编译
- 下载到MCU中
- 运行



## 一. 先安装以下的软件

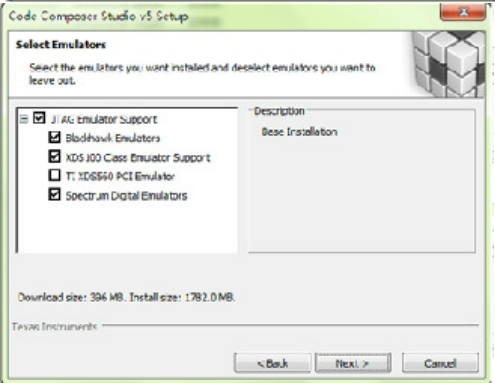
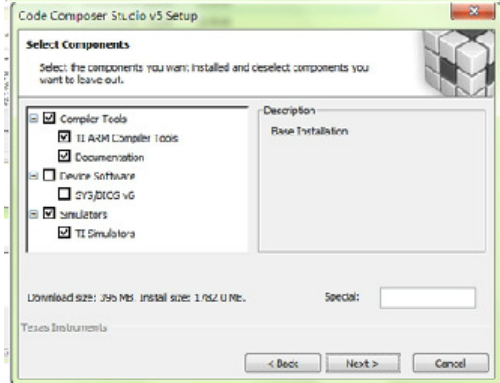
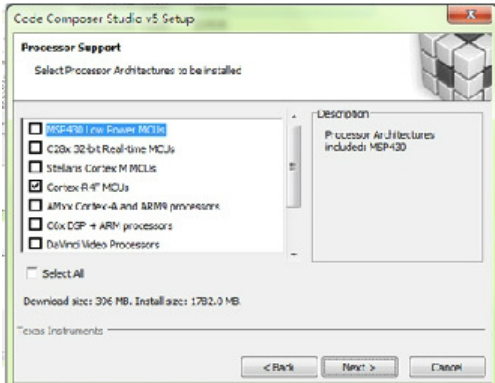
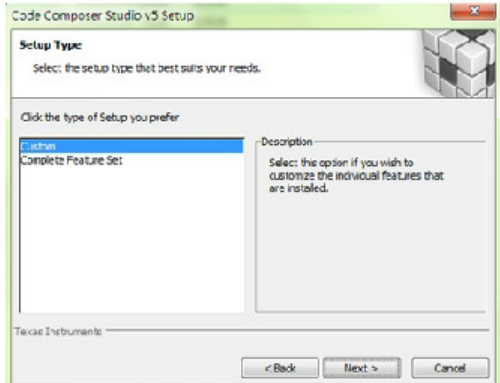
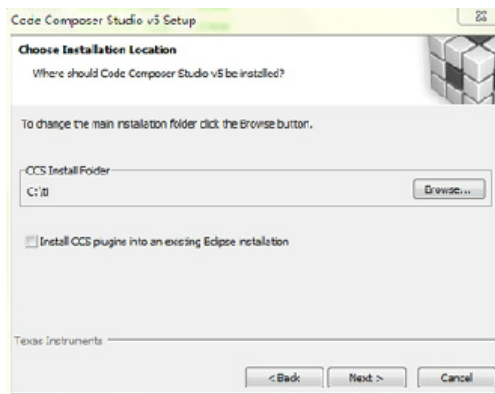
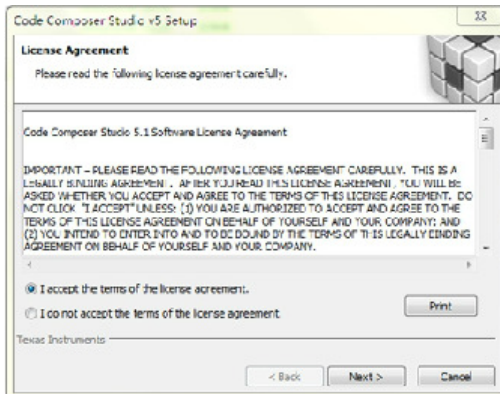


CCStudio  
快捷方式  
2 KB

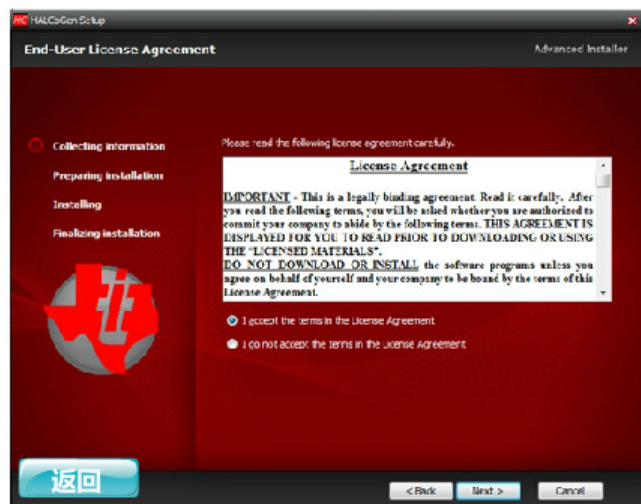


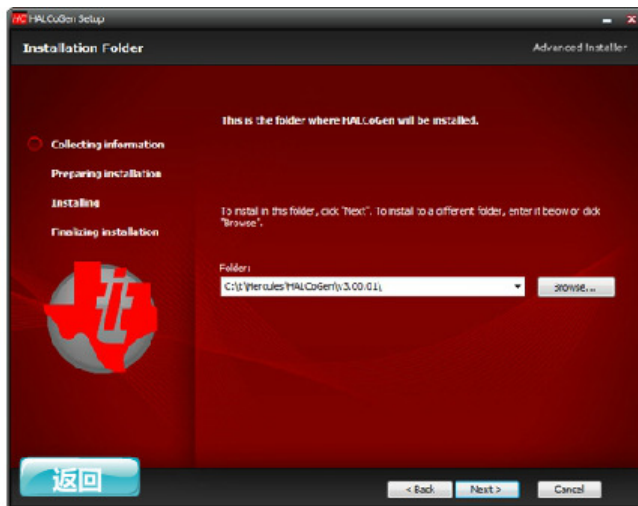
HALCoGen v3.03.01  
快捷方式  
1 KB

[安装CCS V5详细步骤](#)

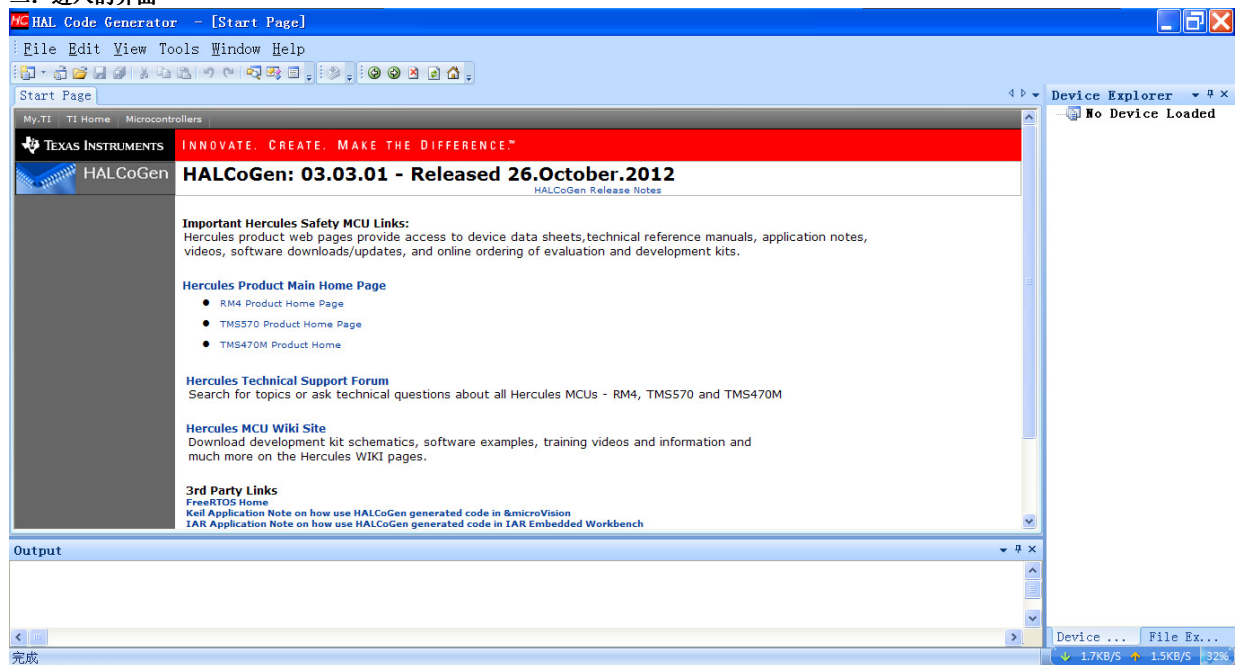


## 安装 HALCoGen





## 二. 进入的界面



TMS470\_P256\_BUTT\_LED\_BUZZ

程序实例

```
//het.h
```

```
void InitHET(void);
```

```
void Beep(void);
```

```
//system.h
```

```

// Init frequency
void InitFreq(void);

// Init buttons
void InitButtons(void);

// Init leds
void InitLeds(void);

// Init CP and WP
void InitCPWP(void);

//het.c
#include <iotms470r1a256.h>
#include "het.h"

void DelayHET_mS (unsigned long a) { while (--a!=0); }

void InitHET(void) {

    HETDOUT = 0x00; // All pins to low
    HETDIR |= 0x14; // HET pin 2 and pin 4 as output
}

void Beep(void) {
    HETDOUT |= 0x4; // HET[2] - high
    HETDOUT &= ~0x10; // HET[4] - low
    DelayHET_mS(3000);
    HETDOUT |= 0x10; // HET[2] - low
    HETDOUT &= ~0x4; // HET[4] - high
    DelayHET_mS(3000);
}

#include <iotms470r1a256.h>
#include <string,h>
#include "system.h"
#include "sci.h"
#include "het.h"

void Delay_mS (unsigned long a) { while (--a!=0); }

int main()
{
    // Init frequency
    InitFreq();

    // Init buttons
    InitButtons();

    // Init HET
    InitHET();

    // Init SCI - necessary, because Status led is connected to SCI Interface
    InitSCI1();

    while(1) {

        // press button 1 - status led ON
        if(GIODINA_bit.GIODIN1 == 0) {LedON();}

        // press button 2 - status led OFF
        if(GIODINA_bit.GIODIN2 == 0) { LedOFF(); }

        // press button 3 -> Beep buzzer
        if(GIODINA_bit.GIODIN3 == 0) { Beep(); }

    }
}

//sci.h
#include "sci.h"
#include <iotms470r1a256.h>

//SCI1
void InitSCI1() {

    //////////////////////////////////////
    // This is necessary because Led STAT is connected to SCI1CLK
    SCI1PC1_bit.CLK_FUNC = 0; // CLK is assign as IO
    SCI1PC1_bit.CLK_DATA_DIR = 1; // CLK as output
}

```

```

}

// Turn on led
void LedON(void) {
    SCIIPC1_bit.CLK_DATA_OUT = 0;
}

// Turn off led
void LedOFF(void) {
    SCIIPC1_bit.CLK_DATA_OUT = 1;
}

// sci.h

// Init SCI
void InitSCI1(void);
// Write char to SCI
void SCI1_write_char(unsigned char ch);
// Read char from SCI
unsigned char SCI1_read_char(void);
// Read char from SCI
unsigned char SCI1_read_char_nonstop(void);

// Init SCI
void InitSCI2(void);
// Write char to SCI
void SCI2_write_char(unsigned char ch);
// Read char from SCI
unsigned char SCI2_read_char(void);
// Read char from SCI
unsigned char SCI2_read_char_nonstop(void);

// Led on
void LedON(void);
// Led off
void LedOFF(void);

#include "system.h"
#include <iotms470r1a256.h>

// Init frequency
void InitFreq(void) {

    // global control clock
    GCR_bit.CLK_DIV_PRE    = 0x0;
    GCR_bit.MULT4         = 0x0;
    GCR_bit.RTI_CTRL      = 0x0;

    // enable peripheral clock
    PCR_bit.PENABLE      = 0x1; // enable
    PCR_bit.CLKDIV       = 0x0;

    // clock control register
    CLKCNTL_bit.CLKSR    = 0x1; // driven by ICLK
    CLKCNTL_bit.PPW_NOVR = 0x0;
}

// Init buttons
void InitButtons(void) {

    // Button 1 as input
    GIODIRA_bit.GIODIR1 = 0x0;

    // Button 2 as input
    GIODIRA_bit.GIODIR2 = 0x0;

    // Button 3 as input
    GIODIRA_bit.GIODIR3 = 0x0;
}

void InitCPWP(void) {

    // WP as input
    GIODIRB_bit.GIODIR2 = 0x0;

    // CP as input
    GIODIRB_bit.GIODIR3 = 0x0;
}

```

```

}

//Init leds
void InitLeds(void) {
    // led is connected to SCI interface (CLK)
}

void LedOn(void) {
    // CLK pin -> low
    SCI1PC1_bit.CLK_DATA_IN = 0x0;
}

void LedOff(void) {
    // CLK pin -> high
    SCI1PC1_bit.CLK_DATA_IN = 0x1;
}

```

## 摘自：EEWORLD 论坛

### Hercules 介绍

作为第一个在 Hercules 板块中发帖滴人，怎么着也要发点儿资料！

上点儿介绍性的内容吧，好多人应该都不知道这是啥。

什么是 Hercules? Hercules 安全微处理器是基于 TI 针对汽车电子市场的 20 多年安全关键型系统专业技术、行业协作而发展成熟的硬件。该平台包含三个基于 ARM® Cortex™ 的微处理器系列 (RM48x、TMS570 和 TMS470M)，可提供可扩展的性能、连接、内存和安全功能。与强烈依赖于软件以获得安全功能的某些微处理器不同，Hercules 微处理器在硬件内实施安全保护，从而使性能最佳化并减少软件使用量。

Hercules 安全微控制器平台包括三个基于 ARM® Cortex™ 的微控制器系列：TMS470M、TMS570 和 RM4x。Hercules 平台专门设计用于 IEC 61508 和 ISO 26262 安全关键型应用，可提供高级的集成安全功能，同时提供可扩展的性能、连接和内存选项。

#### [RM4x 系列](#)

RM48x 是 Hercules 安全微处理器系列中性能最高的产品。基于运行速度高达 220 MHz 的 ARM® Cortex™-R4F 浮点内核，它包括闪存和连接选项。为满足 IEC 61508 SIL-3 安全标准和支持要求，已在硬件中集成多个安全功能。

#### [TMS570LS 系列](#)

Hercules TMS570LS 安全微处理器系列让客户能轻松开发安全关键型运输应用产品。这款基于 ARM® Cortex™-R4F 的系列微处理器可提供多种性能、存储器和连接选择，已开发出满足 ISO 26262 ASIL D 和 IEC 61508 SIL-3 安全标准和符合 AEC-Q100 汽车电子规范的产品。

#### [TMS470M 系列](#)

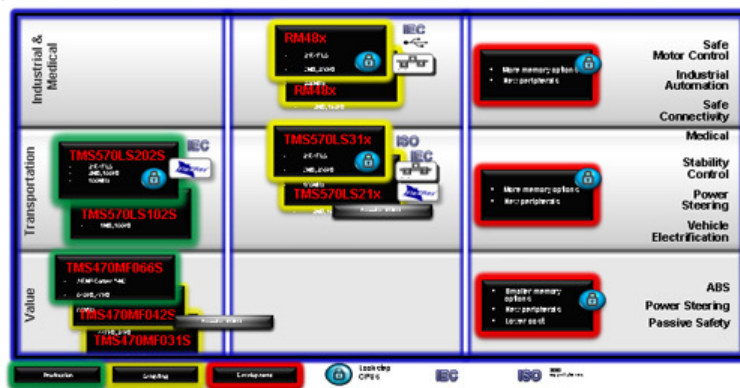
TMS470M 安全微处理器系列基于广泛采用的、运行速度为 80 MHz 的 ARM® Cortex™-M3 CPU。此系列提供多种闪存和 RAM 存储器选项，并提供 CAN 和 LIN 连接及其灵活的控制外设。包括 CPU 和 RAM 自检 (BIST) 引擎、ECC 和 奇偶校验等内置安全功能。TMS470M 安全微处理器同样符合 AEC-Q100 标准。

### Hercules 入门小知识

什么是 Hercules?

Hercules 安全微处理器是基于 TI 针对汽车电子市场的 20 多年安全关键型系统专业技术、行业协作而发展成熟的硬件。该平台包含三个基于 ARM® Cortex™ 的微处理器系列 (RM48x、TMS570 和 TMS470M)，可提供可扩展的性能、连接、内存和安全功能。与强烈依赖于软件以获得安全功能的某些微处理器不同，Hercules 微处理器在硬件内实施安全保护，从而使性能最佳化并减少软件使用量。

产品系列  
现已推出全部 Hercules 3 产品系列，具有备用存储器和已计划的外设选项。



#### [RM4x 系列](#)

RM48x 是 Hercules 安全微处理器系列中性能最高的产品。基于运行速度高达 220 MHz 的 ARM® Cortex™-R4F 浮点内核，它包括闪存和连接选项。为满足 IEC 61508 SIL-3 安全标准和支持要求，已在硬件中集成多个安全功能。

#### [TMS570LS 系列](#)

Hercules TMS570LS 安全微处理器系列让客户能轻松开发安全关键型运输应用产品。这款基于 ARM® Cortex™-R4F 的系列微处理器可提供多种性能、存储器和连接选择，已开发出满足 ISO 26262 ASIL D 和 IEC 61508 SIL-3 安全标准和符合 AEC-Q100 汽车电子规范的产品。

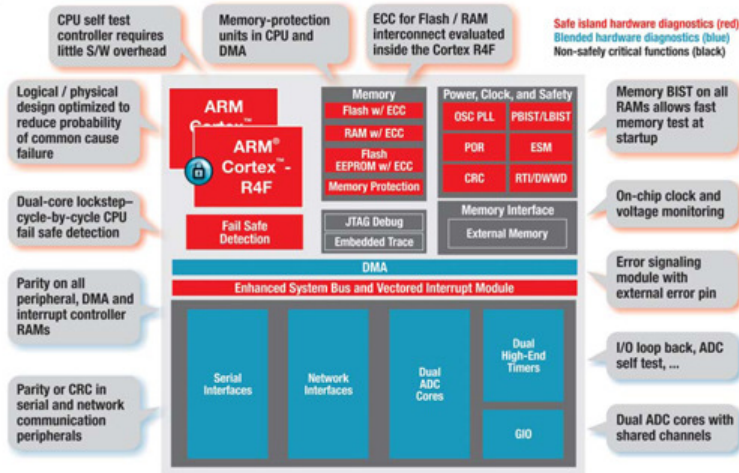
#### [TMS470M 系列](#)

TMS470M 安全微处理器系列基于广泛采用的、运行速度为 80 MHz 的 ARM® Cortex™-M3 CPU。此系列提供多种闪存和 RAM 存储器选项，并提供 CAN 和 LIN 连接及其灵活的控制外设。包括 CPU 和 RAM 自检 (BIST) 引擎、ECC 和 奇偶校验等内置安全功能。TMS470M 安全微处理器同样符合 AEC-Q100 标准。




### 硬件中的安全功能

RM48x 和 TMS570 双核 CPU 锁步构架在消除冗余系统要求的同时也使开发得以简化,从而降低了成本。CPU 硬件内置自检 (BIST) 无需复杂安全软件和代码大小费用即可检测到潜在缺陷。CPU 输出的硬件比较提供即时安全响应时间,而不会对其它性能产生影响。ECC 逻辑集成在 CPU 中,这会对存储器和总线起到保护作用。使用 HW BIST 可对所有 RAM 存储器进行测试,从而实现高诊断覆盖范围,且内存保护单元 (MPU) 对应用软件中的确定性错误起到帮助防范作用。



### Hercules 软件

通过基于硬件抽象层代码生成工具的 GUI (HALCoGen), TI 让您可轻松开始进行任何 Hercules 安全 MCU 的软件开发。



[HALCoGen](#)

- 高抽象层的用户输入
- 基于图形的代码生成
- 轻松配置
- 新项目快速入门
- 支持 CCS、IAR 和 KEIL IDE

### Hercules 安全 MCU

Hercules MCU 具有集成安全功能、开发工具和广泛产品范围,让您轻松开始下一安全应用。

#### 学习 Hercules 基础了解

#### Hercules 安全微处理器

Hercules 微处理器基于 ARM® Cortex™ 的微处理器系列 (RM48x、TMS570 和 TMS470M), 基于 TI 针对汽车电子市场的 20 多年安全关键型系统专业技术、行业协作而发展成熟的硬件。在硬件内实施安全保护,从而使性能最佳化并减少软件使用量。

符合 ISO 26262 ASIL-D

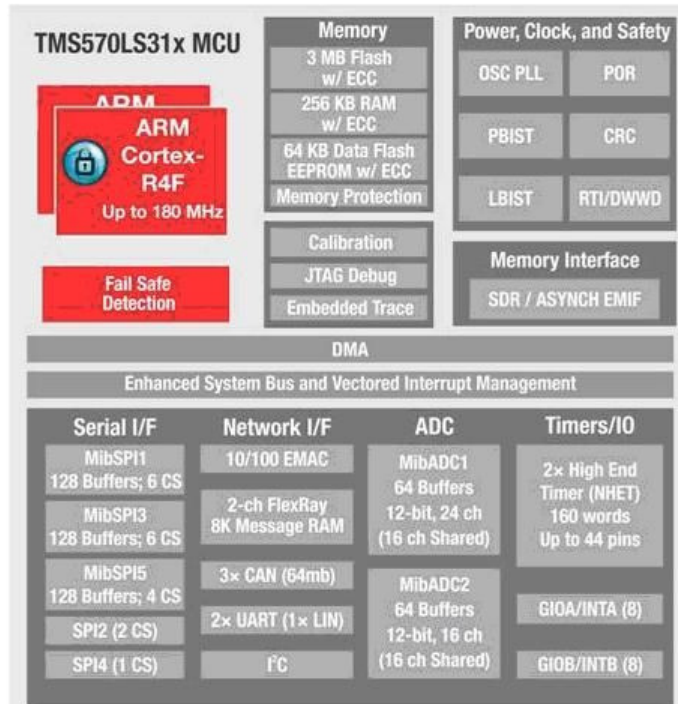
符合 IEC 61508 SIL-3 安全标准

符合 AEC-Q100 汽车电子规范

### Hercules 安全微处理器三大系列

系列	内核	主频	FLASH	RAM	Data Flash
RM4x 系列	ARM® Cortex™-R4F	220MHz 的浮点性能	3072K	256K	64K
TMS570LS 系列	ARM® Cortex™-R4F	180MHz 的浮点性能	3072K	256K	64K
TMS470M 系列	ARM® Cortex™-M3	80MHz	640K	64K	

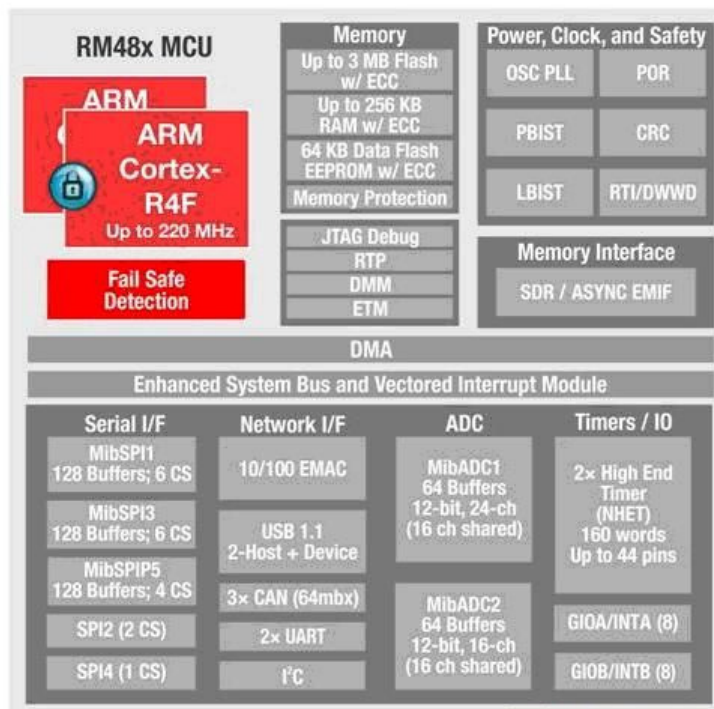
TMS570LS 系列方框图



Packages: LQFP: 144 pin - 20 x 20; nFBGA: 337 pin - 16 x 16, 0.8 mm pitch  
-40 to 125°C temperature range



RM4x 方框图

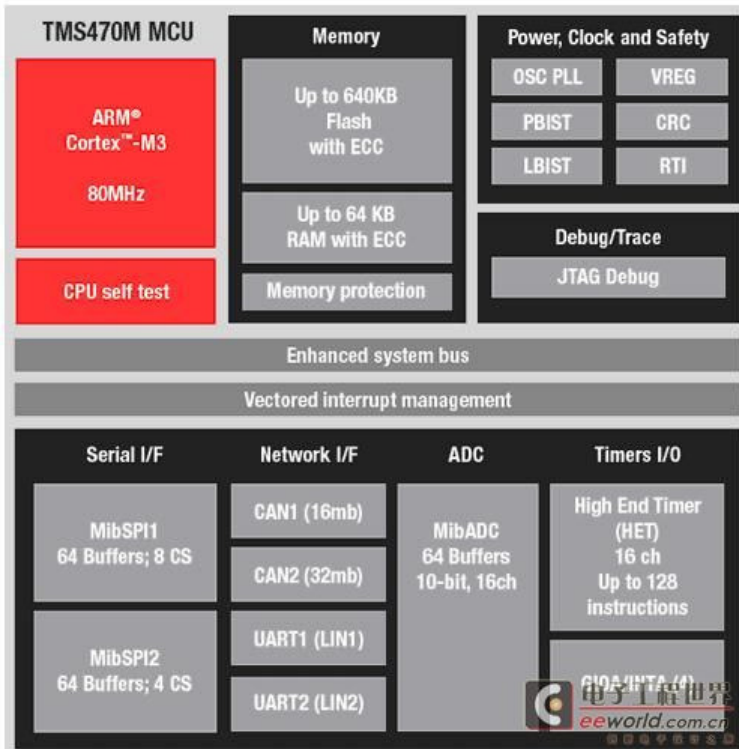


Packages: LQFP: 144 pin - 20 x 20; nFBGA: 337 pin - 16 x 16, 0.8 mm pitch  
-40 to 105°C temperature range





TMS470M 系列方框图



关键字了解

**LBIST:** CPU 逻辑内置自检

**PBIST:** 存储器内置自检

**CRC:** 循环冗余校验器模块

**ECC:** 错误更正代码

ISO 26262

ISO 26262 是一个功能安全标准，“道路车辆-功能安全”。

功能安全特性的组成部分，每个产品开发阶段，包括从规范，设计，实施，集成，验证，确认，生产发布。ISO 26262 标准是适应汽车电气/电子系统的功能安全标准 IEC 61508。ISO 26262 定义的整个生命周期的所有汽车电子和电气安全相关系统的功能安全汽车设备适用。

第一版是 2011 年 11 月 11 日公布，并适用于电气和/或电子系统安装在“系列”，最大总重量 3500 千克生产乘用车。它的目的是解决可能出现的危害行为的电子和电气系统的故障造成的。

该标准由 9 个规范性，和信息引导部分 10。

像其父标准 IEC 61508，ISO 26262 是基于风险的安全标准，其中危险操作的情况下，定性的风险评估和安全措施的定义，以避免或控制系统故障检测或控制随机硬件故障，或减轻其影响。

提供汽车安全生命周期（管理，开发，生产，经营，服务，退役），并支持剪裁必要的活动，在这些生命周期的各个阶段。

涵盖功能安全方面的整个发展过程中（包括需求规格说明，设计，实施，集成，验证，验证和配置等活动）。

提供汽车专用的基于风险的方法来确定风险类别（汽车安全完整性等级，ASILs）。

使用 ASILs 用于指定项目的必要的安全要求，为实现可接受的残余风险。

提供验证和确认的措施，以确保足够的和可接受的安全水平的要求正在实现。[ 1 ]

VDC 研究报告，符合 ISO 26262 和 AUTOSAR 预计在未来两年内显著增加[ 2 ]因此，许多功能安全服务提供商，如千伏安或 SGS，创造了培训计划，以帮助了解各种安全进程。[ 3 ] [ 4 ]以及法律责任和所涉及的实现合规性。

IEC 61508

IEC 61508 是一个国际标准，在工业中应用的规则。它的题目是电气/电子/可编程电子安全相关系统（E/E/PE，或 E/E/PES）的功能安全。

IEC 61508 是一个基本的功能安全标准，适用于各种行业。它规定：“部分的整体安全有关的 EUC（在控制设备）和 EUC 控制系统依赖于正确的 E/E/PE 安全相关系统的功能安全功能，其他技术安全相关系统和外部风险降低设施。”

该标准涵盖了完整的安全生命周期，并可能需要解释开发部门的具体标准。它有它的起源在过程控制行业。

安全生命周期有 16 个阶段，这大致可以分为如下三组：

阶段 1-5 地址解析

阶段 6-13 地址实现

阶段 14-16 地址操作。

所有的相有关的安全系统的功能。

该标准有七个部分：

零件 1-3 包含的标准（规范）的要求，

零件 4-7 指导方针和发展的例子，因此翔实。

标准的核心是风险和安全功能的概念。的风险是频率的函数（或可能性）的危险事件和事件后果严重性。应用安全功能，包括 E/E/PES 和/或其他技术的风险降低到可接受的水平。而其他的技术可以在降低风险，只有那些依赖于 E/E/PES 安全功能所涵盖的具体要求，IEC 61508。

[http://en.wikipedia.org/wiki/IEC\\_61508](http://en.wikipedia.org/wiki/IEC_61508)

学习 Hercules “安全”系统基础知识

## 一、基本功能安全概念

所有系统都有的可以量化的试销概率，绝对安全的系统是不存在的

对于每个应用，都有一个可以容忍的不会导致不可接受风险的失效概率

每个应用的可接受的失效概率都不同，取决于系统故障时造成的直接或间接的伤害损失的可能性

相同级别的风险可以被量化，被称作安全完整性等级 **Safety Integrity Level**,简称 **SIL**

功能安全示例：一个过热保护装置当电机过热时，会断开电机。这是一个功能安全示例。而一个可以忍受很高温度的特制电机则不是功能安全

在 IEC61508 中的定义

安全是避免系统遭受对人身财产安全造成的直接或间接的损害或者对环境造成不良影响等不可接受的风险

功能安全是安全的一部分，取决于对系统或装置对输入正确的响应

在 ISO26262 中的定义

避免由于人工操作及其他原因发生的故障造成不可接受的风险

## 二、安全性失效和安全性失效的起因

功能安全系统的失效（广义）分类

### 1、系统性失效

A、有设计或是生产过程中的失效或者说错误所产生

即使是最佳的方法，通常也都会有一个失效的结果伴随

可以通过持续严格的过程改善来降低系统性的失效率。

B、软件失效

\*对策\*

保证质量和提高可靠性的方法在一定程度上可以解决大部分由系统性失效所引起的安全性失效

### 2、随机性失效

有产品生产和使用过程中自身固有的随机缺陷所产生

随机失效率一般无法被降低；必须依靠在应用中对随机失效的检测和处理上来改善或解决。

## 三、安全目标

安全目标 确定相应的功能安全等级

安全等级指安全完整性等级 **SIL** (**Safety Integrity Level**)

不同安全标准有不同的尺度和不同的命名分类

比如 **ISO26262** 和 **IEC61508** 的安全等级 **SIL**

**ISO26262**：主要为了满足汽车行业的特殊需求所制定，用来细化 **IEC61508**

**IEC61508**：定义了 4 层安全完整性等级

SIL-1

SIL-2

SIL-3

SIL-4



**ISO26262** 除了定义了 4 级安全等级之外还定义了质量管理等级

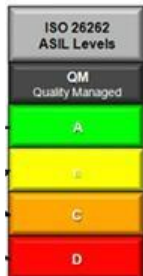
ASIL-A

ASIL-B

ASIL-C

ASIL-D

QM



**IEC61508** 的 **SIL** 等级和 **ISO26262** 的 **ASIL** 等级之间没有直接的对应关系的



#### 四、安全系统的架构

##### 1、常用架构

“XooY” ---X out of Y

Y=冗余通道数

X=如果满足系统需要所需要的通道数

##### 2、系统架构分类

##### A、1001 系统架构



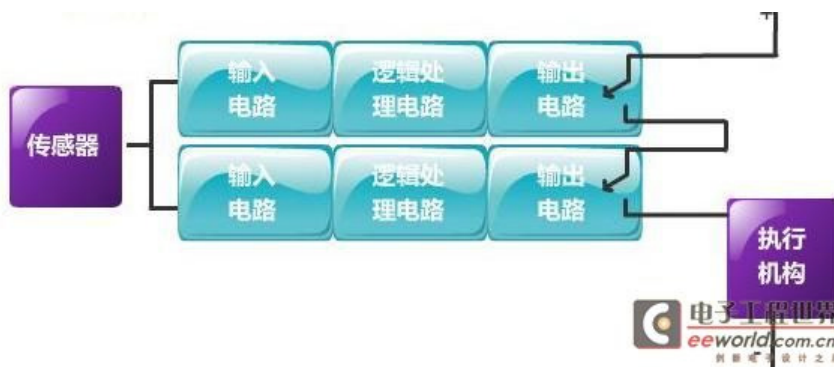
特点:

子系统失效将导致整个系统失效

无内部诊断

无故障冗余

##### B、1oo2 系统架构



特点:

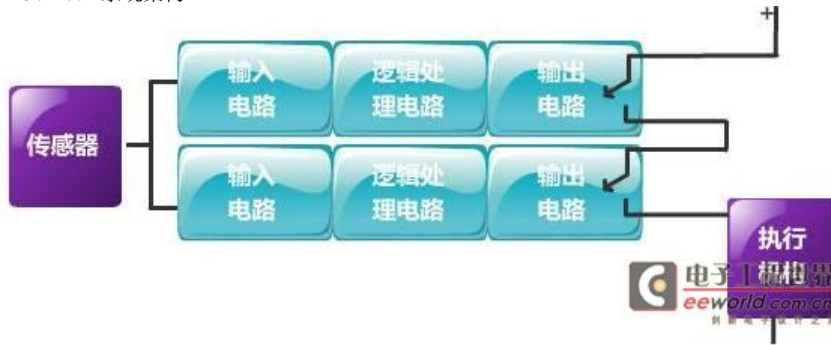
2 个子系统中任何一个失效将导致系统失效

两路子系统有独立 IO 的控制器

每个控制器都可以控制输出

任何一个失效将导致系统失效

C、2oo2 系统架构



特点:

2 个子系统都失效将导致系统失效

每个控制器都可以控制输出

可保证系统在其中一路有异常断开时，保证系统仍然可靠闭和  
子系统开路有故障冗余

\*\*\*\*\*疑问\*\*\*\*\*

\*按照许工讲解时，这里输出应该是并联的，可示意图还是串联的！\*

\*\*\*\*\*

D、1oo1D 系统架构



特点:

1oo1D 带诊断通道系统

诊断通道可以禁止系统输出

诊断电路会导致额外的失效概率

Hercules 处理器 就是 1OO1D 系统!

E、2oo3 系统架构



特点:

## TMR(三重模块冗余)系统

### Hercules 安全 MCU 汽车运输设计案例

很多人都想知道 Hercules 的一些案例 下面分享一个转我们台湾同胞的一贴

Hercules TMS570 安全微控制器连同 TI 首款功能型安全 PMIC 與業界首款功能型安全馬達驅動器組成「安全馬達控制晶片組」,可用最低軟體開銷達到最大故障檢測 (failure detection) 與減緩 (mitigation) 能力。全新馬達控制安全晶片組採用 SafeTI-26262 及 SafeTI-61508 設計套件,能協助客戶輕鬆通過 ISO 26262 及 IEC 61508 認證,並加速安全關鍵型汽車與運輸馬達控制應用的上市時程。最新款 Hercules TMS570 安全微控制器產品內含 36 種配置,客戶可依據應用特定需求選擇,相當適合進階駕駛輔助系統 (advanced driver assistance system)、電動輔助轉向 (electric power steering)、油電混合車 (hybrid and electric vehicle)、鐵道推進控制 (rail propulsion control)、航空防滑控制 (aviation anti-skid control) 與越野車 (off-road vehicle) 等應用。新款 Hercules TMS570LS12x/11x 浮點安全微控制器具有擴充的馬達控制功能,提供了更多的記憶體與效能配置,而 Hercules TMS570LS04x/03x 安全微控制器提供更小型封裝的低成本入門級解決方案,具備整合式馬達控制介面。相容型 PMIC 將多個電源供應和安全功能整合在單一裝置中,可縮減設計時間與電路板空間。該馬達驅動器整合防護與監控功能的功能型安全架構,在市場上率先支援啟動/停止功能。Hercules TMS570 安全微控制器的特性包括:

1. 擴大效能及記憶體選項,從 80 MHz 到 180 MHz 鎖步 ARM Cortex-R4 核心,以及 256KB、384KB、1MB 和 1.25MB 快閃記憶體,可為設計人員的多功能安全性產品提供更多的選擇與可擴充性。
2. 元件符合業界安全標準,進而在系統設計中更容易達到 ISO 26262 與 IEC 61508 標準。
3. 硬體安全特性提供高水準線上診斷,其具備鎖步核心 (lockstep cores)、用於 CPU 和匯流排主控器 (bus master) 的記憶體保護、具有單位元錯誤糾正與雙位元錯誤偵測 (single-bit error correction and double-bit error detection; SECDED) 的快閃記憶體與 RAM 錯誤修正碼 (ECC)、偵測潛在延遲故障 (latent faults) 的 CPU 及 RAM 內建自我測試 (built-in-self-test; BIST)、可根據安全錯誤採取適當措施的智慧型錯誤訊號傳輸模組、週邊 RAM 的奇偶校驗 (parity)、冗餘類比數位轉換器 (ADC) 和計時器,以及連續的電壓和時脈監控。
4. 整合型馬達控制功能,具有內建增強型脈衝寬度調變器 (pulse-width modulator; PWM)、感測器捕獲和正交編碼器 (quadrature encoder) 介面,可免除用於馬達控制的多個外部組件。32 通道計時器協同處理器可做為一個性質不同的冗餘馬達控制通道,並檢查馬達控制迴路中脈衝寬度調變器的完整性。  
與去年推出並預定於 2012 年 10 月 31 日量產的 Hercules TMS570LS31x/21x 安全微控制器接腳相容 (僅限 Hercules TMS570LS12x/11x 浮點安全微控制器)。
5. 內建 FlexRay、CAN、乙太網路、LIN 連結,可實現標準汽車網路並為資料收集提供多種選項。

首款功能型安全 PMIC TPS65381-Q1 的特性包括:

1. 單個裝置中包含了多個電源軌,包括一個具有內部 FET 的非同步降壓開關模式電源轉換器,其負責把輸入的電池電壓轉換為 6V 前置穩壓器 (pre-regulator) 輸出,該輸出可為其它穩壓器供電。兩個具有內部 FET 的線性穩壓器可為一個控制區域網路 (CAN) 和微控制器輸入/輸出 (I/O) 供電。一個線性穩壓器控制器負責供電給微控制器核心。這種高整合度縮短了設計時間,同時也節省寶貴的電路板空間。
2. 一個附加的整合型感測器電源同時為接地短路 (short-to-ground) 和電池短路 (short-to-battery) 提供保護,其可為位於電子控制單元 (ECU) 外部的感測器供電。
3. 功能型安全架構整合以下特性,例如問答型看門狗 (watchdog)、MCU 錯誤訊號監視器、內部振盪器的時脈監控、時脈監控器的自我檢查、非揮發性記憶體的循環冗餘校驗 (CRC) 及一個用於微控制器的重設電路。此外, BIST 可監視啟動時的裝置功能,而一種專用的診斷狀態則允許微控制器檢查 PMIC 的安全功能。這些嵌入式安全特性免除了增設一個額外監視微控制器的需要,並縮減了成本與電路板空間。功能型安全馬達驅動器 DRV3201-Q1 的特性包括:
1. 專為汽車三相無刷 DC 馬達設計的橋接驅動器,提供一般 N 通道 MOSFET 的六個專用驅動器,最高可達 250nC 電荷。可程式源極/汲極 (source/sink) 驅動電流,簡易進行輸出斜率調整。
2. 市場上首款支援啟動/停止功能的馬達驅動器,在電池電壓降至 4.75 V 情況下也可完全控制功率級 (power stage)。
3. 功能型安全架構整合多項特性,例如電壓汲源極 (voltage drain-to-source; VDS) 監測、相位比較器 (phase-comparator)、擊穿保護 (shoot-through protection)、停滯時間控制 (dead-time control)、溫度警告與保護、欠電壓與過電壓保護的電池電壓偵測。

### 学习将 hal 产生的代码加载到 CCS5 中一般流程

首先使用 HAL 生成一个工程,利用 HAL 生成代码后把生成的所有代码在 CCS5 新建的工程中通过 add file 加入,再将系统提供的 CMD, CCMXL 及 GEL 文件添加到工程中去。

1. 选中.ccxml 文件
2. 选择[Advanced]选项卡
3. 选择[Spectrum Digital XDS510USB\_Emulator\_0]
4. 在右边[TCK Mode]下拉菜单中选择 Adaptive - Synchronous 项。

然后再尝试连接。

如果仍然无法连接,请尝试将频率降低至如 6MHz。

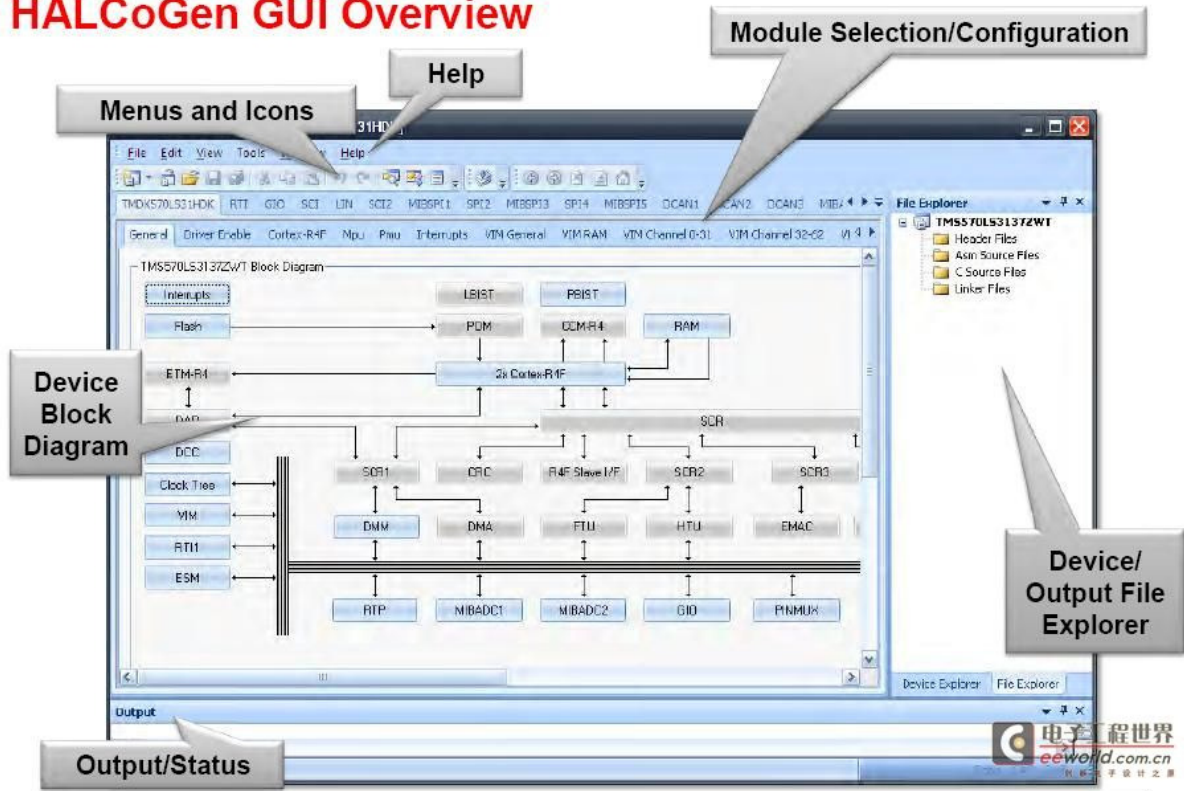
### hercules 的 GUI 代码生成器 HALCoGen 了解

HALCoGEN 可提供图形用户界面,允许用户配置外设、中断、时钟和其他微处理器参数。配置器件后,用户可生成外设初始化和驱动程序代码,可导入 CCS、IAR 工作平台或 Keil uVision 中。这点好像比之前 430 的那个图形工具先进了点,能导出到其它的编译环境用了。

打开 HALCoGEN 界面介绍如下



# HALCoGen GUI Overview



这个 GUI 工具 目前支持的外设有

RTI , System Module , GIO, SCI/LIN, CAN , SPI,Timer Co-processor (HET),ADC.总体一览图如下

