

嵌入式期中作业——

基于 MC9S12XEP100 的计算器设计

机器人 1702 班 邓志豪 20173878

Question1. 对所设计的计算器系统进行需求分析

- 系统功能：

- 1) 实现浮点型或整型加、减、乘、除、求倒数、求余数运算功能
- 2) LCD 显示运算式和运算结果
- 3) 退格功能，可删除输入的数字或符号
- 4) 可算整数的 N 次方
- 5) 实现二进制的转换
- 6) DEL 键有清屏功能
- 7) 错误信息显示
- 8) 实现触摸屏操作及显示功能

- 系统的性能预期

系统拟采用 MC9S12XEP100 MCU 进行计算器系统的设计，飞思卡尔的 S12XE 系列控制器主要应用于汽车微控制器的设计，S12XE 系列 16 位微控制器 (MCU) 内置 XGATE 协处理器，集 32 位性能和 16 位架构的所有现有优势于一身，如低成本、代码高效和出色电磁兼容性等。因此其性能足够支撑市面上大部分科学计算器的计算需求。

- 系统的功耗设计

总的来看，也许 freescale 的 S12 系列单片机的功耗没有 msp430 的低，但是他具有全静态的“等待”和“停止”两种模式，从总体上降低计算器的功耗。且其新近推出的几款超低功耗已经与 msp430 的不相上下！其强劲的低功耗性能个人感觉比较合适此科学计算器的设计。

- 系统的物理尺寸设计



Fig1. MC9S12XEP100 板图

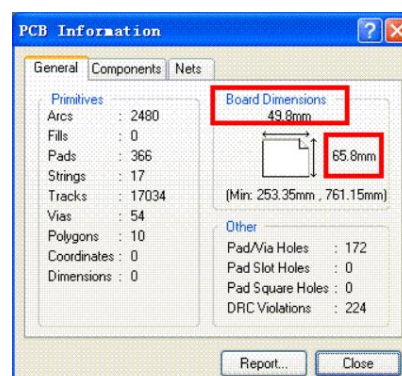


Fig2. 开发板大致尺寸

- 系统工作环境设计

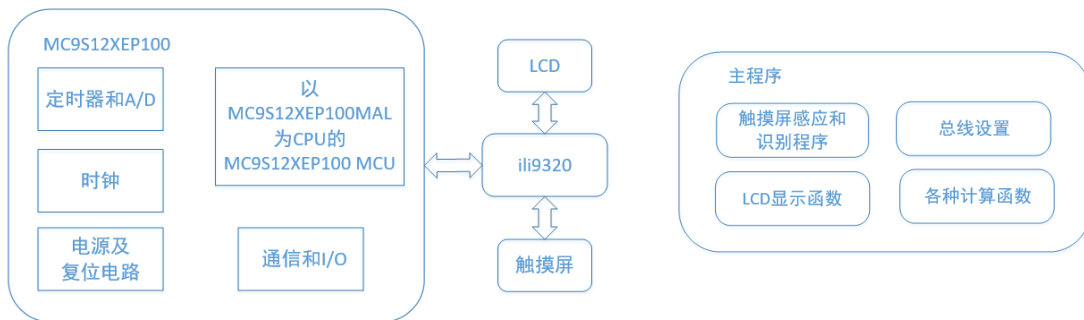
- 1) 工作电压：3.3V 至 5.0V 之间
- 2) 温度范围：-40°C 至+125°C 之间
- 3) 防灰防尘
- 4) 防水功能需要在计算器外壳方面下功夫

Question2. 对计算器进行系统级设计

此部分暂时仅列出大致结构框图，具体部分在期末作业中展开：

1) 硬件部分（仅列出部分重要部分）：

2) 软件部分：



Question3. 对系统硬件子系统进行设计

Question3-1. 处理器选型

本设计基于 MC9S12XEP100 MCU，处理器采用的是飞思卡尔 S12XE 系列处理器



其具体配置如下所示：

应用

- 汽车车身应用
- 车身中央控制
- 网关
- 智能配线盒

主要特性

内存

- 128 KB至1 MB嵌入式闪存，带纠错码（ECC）
- 12 KB至64 KB RAM
- 高达32 KB的data-flash，256B可访问扇区
- 模拟EEPROM

系统

- 50 MHz 16位CPU12X，前向兼容MC9S12指令集
- XGATE I/O协处理器模块，高达100 MHz的总线频率。所有外设和RAM之间的往来数据传输无需CPU干预或等待
- 增强型中断模块
- 内存保护单元
- 系统可以在监控或用户模式下运行
- 带有单线接口的后台调试模块（BDM）
- 3.3V至5.0V工作电压
- 温度范围：-40°C至+125°C
- 片上电压调节（VREG）

优点

卓越的系统性能：将CPU总线频率提高到50 MHz，CPU吞吐率也受益于XGATE协处理器性能的增强。

扩展性和兼容性：将S12X内存规模扩展到最大1 MB，并为S12XD系列提供了升级路径。

通信和I/O

- 高达5个MSCAN模块，兼容CAN 2.0 A、B
- 高达3个串行外设接口（SPI）模块
- 高达8个支持LIN的串行通信接口（SCI）
- 高达152个通用输入/输出（GPIO）引脚和2个仅输入引脚

定时器和A/D

- 高达2个独立ADC转换器，具有8/10/12位分辨率和用于最多32个模拟输入通道的复用器
- 8通道8位或4通道16位脉宽调制器
- 定时器（TIM），带有用于输入捕捉或输出比较的8个16位通道
- 增强型捕捉定时器（ECT），带有用于输入捕捉或输出比较的8个16位通道
- 定期中断定时器（PIT），高达最多2个8位通道

时钟

- 带内部滤波的锁相环（IPLL）— 无需外部组件
- 从省电的STOP模式快速唤醒，并立即执行程序

规范

- 封装选项
- 208引脚MAPBGA
- 144引脚和112引脚LQFP
- 80引脚QFP

更高的系统完整性：在MCU层，诸如ECC监控模式和MPU这样的特性使AUTOSAR的集成更简便。

灵活性：提供更大内存、增强型ATD和大量外设及封装选项支持客户要求

开发支持：利用并扩大了目前市场上可用于S12和S12X系列的广泛硬件和软件开发工具套件。

Question3-2. 硬件模块划分设计

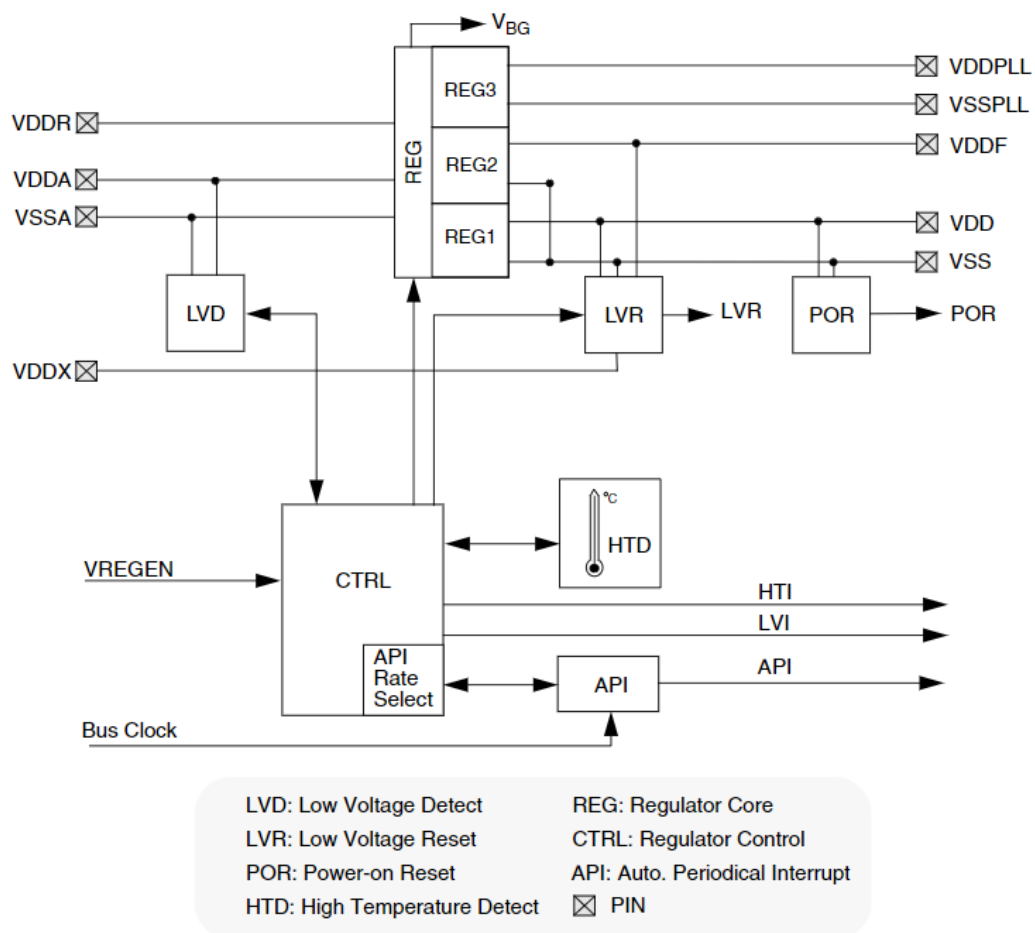
鉴于本设计采用飞思卡尔 MC9S12XEP100 MCU 进行设计，这里对其重要模块进行具体描述（具体部分可以查阅 MC9S12XEP100 Reference Manual）：

1) 电源电路

VREG_3V3 模块是一个三路输出稳压器，提供两个单独的 1.84V（典型值）电源，其可提供的电流量和一个 2.82V（典型值）电源有所不同。稳压器的输入电压范围为 3.3V 至 5V（典型值）。

VREG_3V3 模块具有以下独特功能：

- 三个带隙基准的并行线性稳压器
- 带低压中断（LVI）的低压检测（LVD）
- 上电复位（POR）
- 低压复位（LVR）
- 带有高温中断（HTI）的高温检测（HTD）
- 自动周期性中断（API）



2) 时钟和复位电路

设计采用时钟和复位发生器 (S12XECRG) 模块, 该模块的主要特点是:

- 带有内部滤波器的锁相环 (IPLL) 倍频器

- 参考分频器
- 后分频器
- 可配置的内部滤波器 (无外部引脚)
- 可选的调制频率, 用于定义的抖动和降低的发射量
- 自动锁频检测器
- 进入或退出锁定状态的中断请求
- 没有参考时钟的自时钟模式

- 系统时钟发生器

- 时钟质量检查
- 用户可以选择从自时钟停止中快速唤醒以节省功耗并立即执行程序
- 时钟开关, 适用于基于振荡器或 PLL 的系统时钟

- 具有超时清除窗口的计算机正确操作 (COP) 看门狗定时器。

- 可通过以下几种方式产生系统复位:

- 上电复位
- 低压复位
- 非法地址重置
- COP 重置
- 时钟丢失重置
- 外部引脚重置

- 实时中断 (RTI)

此外, 还有四种操作模式可供选择:

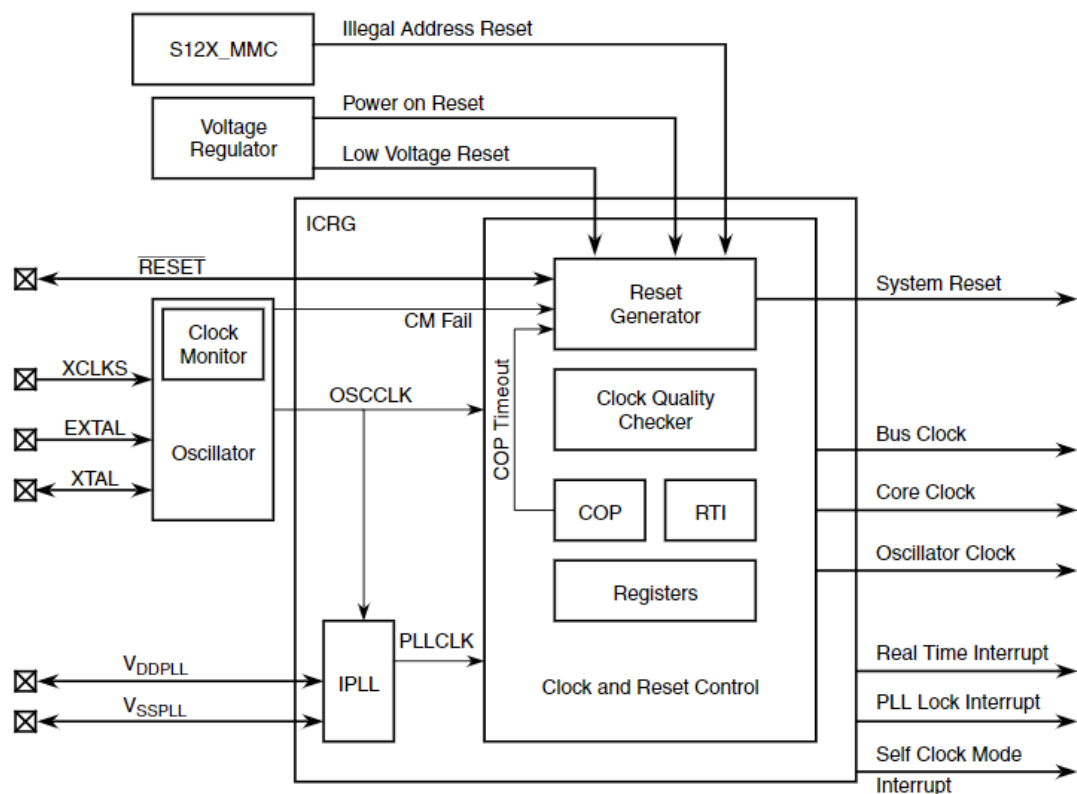
- Run Mode

- Wait Mode

- Stop Mode

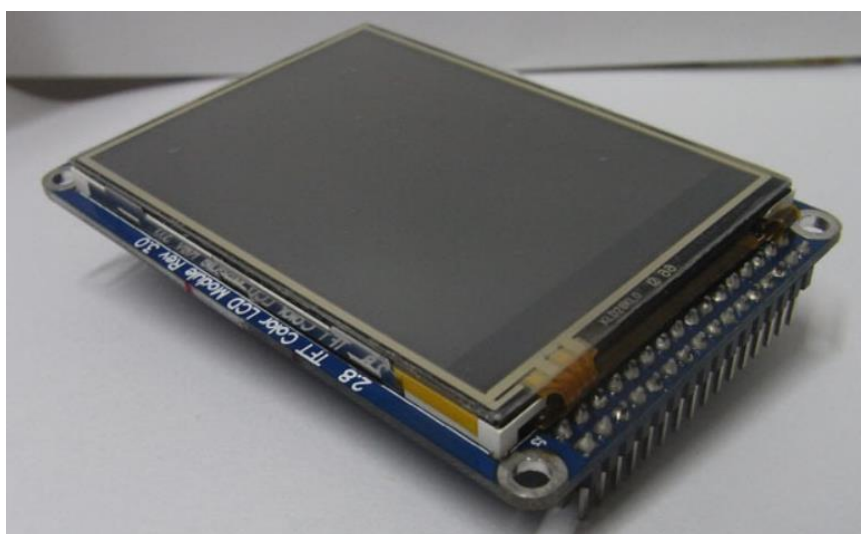
- Full Stop Mode
- Pseduo Stop Mode

- Self Clock Mode



鉴于本设计采用飞思卡尔 MC9S12XEP100 MCU 进行设计，很多模块已经集成好，额外附加模块仅讨论 LCD 与触摸屏模块的选取。为了同时兼顾 LCD 和触摸屏功能，让具体程序实现更为简单，这里采用 ili9320 LCD 模块进行设计。

- 市价：50~70 左右
- 多尺寸可选择：可选 320*240、240*320，3.2 寸适合所选 MCU 尺寸
- 彩屏、支持 26 万色



Question4. 对软件子系统进行设计

Question4-1. 操作系统选型

首先本设计不选用具体操作系统而直接使用裸机。考虑到其 16 位的 CPU 仅有 50MHZ 的主频，其较小的内存以及其他相关因素影响，无法直接支撑操作系统，这里不采用具体的操作系统进行直接部署。作为取代，可以采用 Windows 下 keil uVision5 进行编译后，烧写进 MC9S12XEP100 MCU，进而对其进行项目部署，也可以在 Linux 系统下采用 Vscode 作为编辑环境，搭配其 PlatformIO IDE 插件进行相关开发。本设计采用前者作为项目开发。

Question4-2. 编程语言与开发工具选型

编程语言选择 C/C++ 语言进行编程，开发工具选择 keil uVision5。

首先我们明确一下什么是嵌入式开发，嵌入式开发就是指在嵌入式操作系统下进行开发，一般常用的系统有 WinCE,linux,android 等。另外，用 c,c++ 或汇编开发；用高级处理器，arm7,arm9,arm11,powerpc,mips,mipsel 等，或加上操作系统也属于嵌入式的开发。而操作系统（包括内核、驱动等）是使用 C 语言和汇编编写的，Windows，UNIX，Linux 都是这样的。绝大部分是 C 语言，极少情况使用汇编写的。

接下来我们看看具体 C/C++ 语言在嵌入式开发中有什么有点：

- C 语言具有出色的移植性，能够在多种不同体系结构的软件或硬件平台上运行；

可移植性并不是指所写的程序不作修改就一定可以在任何计算机上运行，而是指当系统环境改变时，程序无需作很多修改或不修改就可运行。

使程序可移植第一步就是使用标准库函数，并且把它们和 ANSI / ISO C 标准中定义的头文件放在一起使用。

第二步是尽可能使所写的程序适用于所有的编译程序，而不是仅仅适用于你现在所使用的编译程序。

第三步是把不可移植的代码分离出来。

- C 语言具有简介紧凑、使用灵活的语法机制，并能直接访问硬件；

能直接访问硬件：指在 C 语言中，我们可以通过指针控制内存以及寄存器。嵌入式系统是软硬结合的东西，所以我们选择学习能直接访问硬件的高级语言——C 语言。

我汇编和 C 语言都可以直接访问硬件。但汇编的语法较为简单，所以对硬件做初始化操作时选择汇编，而对硬件做初始化时选择 C 语言（C 语言是高级语言中最低级的，但是 C 语言也具有高级语法）

- C 语言具有很高的运行效率。

很高的运行效率：C 语言是高级语言中最低级的语言，可以直接编译生成硬件识别的机器码文件。而其他的高级语言需要相应的语言解释器做解释后，再编译。接下来通过 C 语言与 java 语言的比较，来帮助读者理解为何 C 语言相比其他高级语言执行效率较高。