

ARM 实验（Cortex-M4）

实验一 时钟选择与 GPIO 实验

一. 实验目的

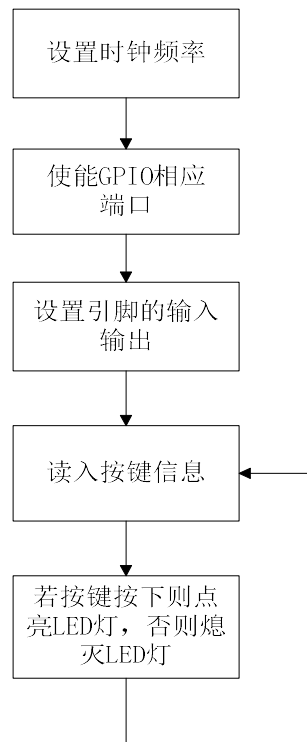
熟悉 ARM 的集成开发环境 KEIL uVision5，能够自行建立一个实验用工程项目。
理解 CPU 的时钟信号，了解不同时钟对电源消耗的不同。
掌握 GPIO 的工作原理，能够结合 GPIO 的输入与输出功能进行实验。

二. 实验要求

1. 分别使用内部 16M 的 PIOSC 时钟，外部 25M 的 MOSC 时钟，以及 PLL 时钟进行 GPIO-PF0 的闪烁。参阅实验 exp1-1.c 的程序。
进入 DEBUG 模式，设置断点，观察设置时钟后时钟返回值的大小是否符合设定。
示例程序为正常时，慢闪 PF0，当按下 USR_SW1 时，快闪 PF0。
2. 修改程序
当按下 USR_SW1 键时，点亮 LED_M0，放开时，熄灭 LED_M0
当按下 USR_SW2 键时，点亮 LED_M1，放开时，熄灭 LED_M1
3. 修改程序
当第一次短按 USR_SW1 键时，闪烁 LED_M0，
当第二次短按 USR_SW1 键时，熄灭 LED_M0
当第三次短按 USR_SW1 键时，闪烁 LED_M1
当第四次短按 USR_SW1 键时，熄灭 LED_M1
如此循环往复
4. 步进电机的驱动。循环执行以下过程，观察红色指针的转动：
 - PF3-ON，Delay（10ms）；PF3-OFF
 - PF2-ON，Delay（10ms）；PF2-OFF
 - PF1-ON，Delay（10ms）；PF1-OFF
 - PF0-ON，Delay（10ms）；PF0-OFF参阅实验 exp1-4.c 的程序。
修改程序采用单四拍、双四拍或八拍方式，依次给 PF0~PF3（或者 PF3~PF0）对应的各个相输入高电平信号，使红色指针顺时针一圈，逆时针一圈，循环往复。
改变延时时间可以得到不同的转动速度

三. 实验框图

实验要求 2:



四. 实验结果

应能现场演示实验的效果。

五. 讨论

1. 人为修改内部时钟或外部时钟，如将内部时钟改为 8M，或将外部时钟改为 30M，会有什么结果？
2. 在使用 PIOSC 及 MOSC 时，能否生成非晶振频率？如 1M 或 10M？
3. 能否将 PLL 时钟调整到外部时钟的频率以下？如将 25M 外部时钟用 PLL 后调整为 20M？
4. 在使用 PLL 时，系统频率最小值及最大值分别为多少？
5. 将 PLL 后的时钟调整为最大值 120M，LED 闪烁会有什么变化？为什么？
6. `GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_0, GPIO_PIN_0)` 此函数中，每个函数项的意义。第三个函数项为 `GPIO_PIN_0`，如果改为 1 或改为 2，或其他值，分别有什么现象？
7. 结合硬件说明 `GPIOPinConfigure` 行的作用。如果此行注释，在 WATCH 窗口中观察 `key_value` 值会有什么变化。

六. 扩展实验 GPIO、中断与 SysTick 实验

实验目的：掌握 GPIO 中断编程和 SysTick 定时器的使用。

实验要求：

- 5、 使用 GPIO 中断方式实现实验要求 3。
- 6、 修改上述程序，使用 SysTick 定时器来控制 LED_M0 和 LED_M1 灯的亮灭和闪烁：
 - 当第一次短按 USR_SW1 键时，LED_M0 以 1s 的周期闪烁
 - 当第二次短按 USR_SW1 键时，熄灭 LED_M0
 - 当第三次短按 USR_SW1 键时，LED_M1 以 2s 的周期闪烁
 - 当第四次短按 USR_SW1 键时，熄灭 LED_M1