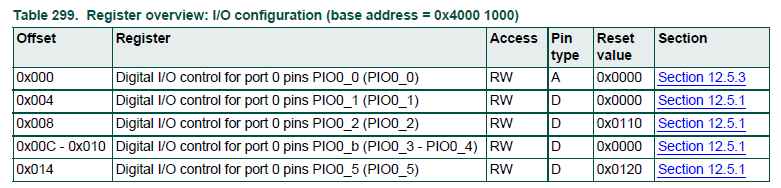
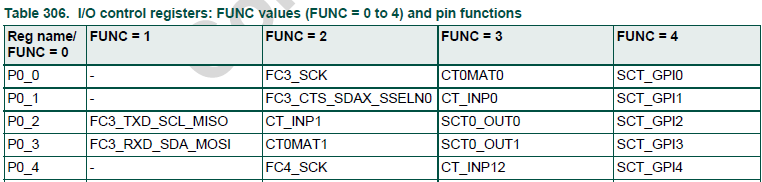
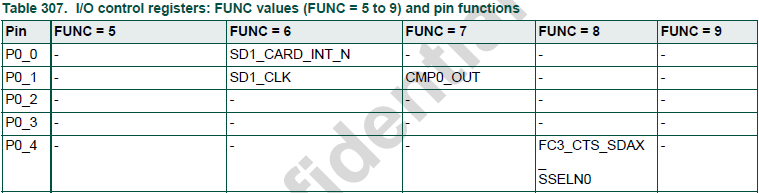
# IOCON

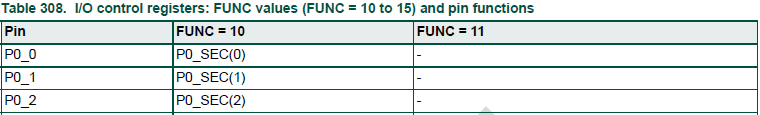
一个pin可以配置为多种用途，IOCON（I/O pin configuration）就是用来配置pin的功能

Niobe4可以配置的功能如下









## 使能IOCON clock(实际操作中这步可以省略)

CLOCK\_EnableClock(kCLOCK\_Iocon);

SYSCON->AHBCLKCTRLSET[index] = (1U << CLK\_GATE\_ABSTRACT\_BITS\_SHIFT(clk));

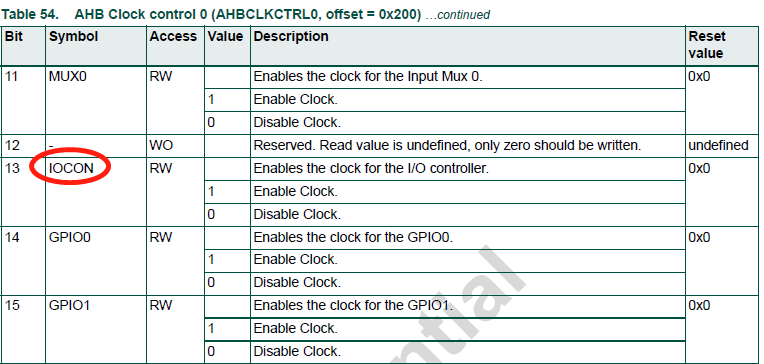
// 如上操作实际上就是SYSCON->AHBCLKCTRLSET[0] = (1U << 13);

### 直接使能clock（AHBCLKCTRL0）

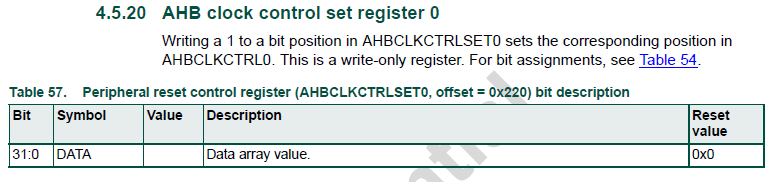
使用IOCON进行pin的配置前，需要使能IOCON的时钟（不使能时钟配置pin也能成功，这里我们就参照RM的要求进行配置）。RM中推荐的做法如下



SYSCON->AHBCLKCTRL0[13] = 1 使能IOCON clock。



### 使用辅助寄存器（AHBCLKCTRLSET/AHBCLKCTRLCLR）使能clock



浏览AHB clock control0的寄存器描述，可以看到。

RM推荐使用AHBCLKCTRLSET/AHBCLKCTRLCLR对AHBCLKCTRL进行配置。

查阅SYSCON寄存器，我们可以发现。

每个AHBCLKCTRL都对应有AHBCLKCTRLSET/AHBCLKCTRLCLR.

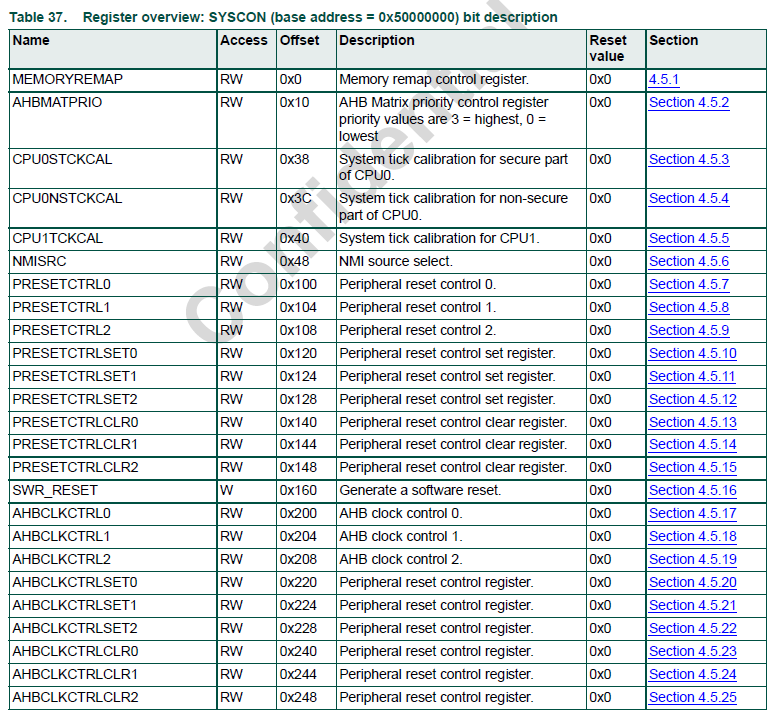
对辅助寄存器的操作对应到主寄存器的每个bit

具体用法就是：

AHBCLKCTRLSET0[13] = 1 //即配置AHBCLKCTL0[13] = 1,使能IOCON时钟

如果想要清除这个bit，依照如下操作

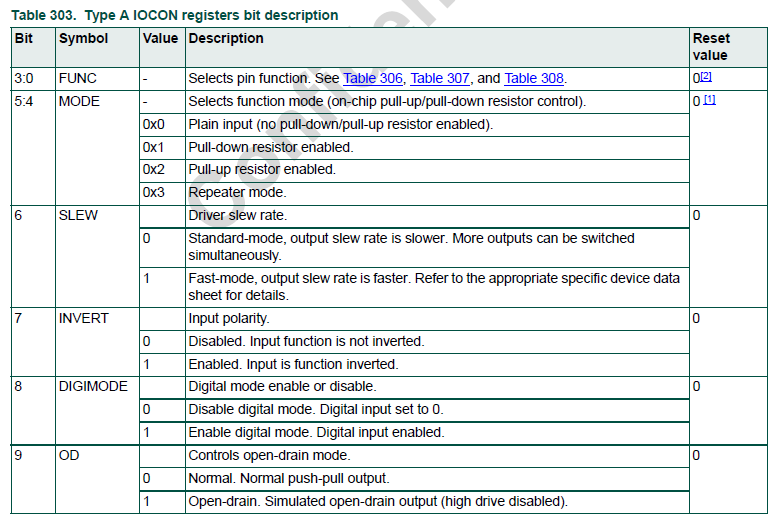
AHBCLKCTRLCLR0[13] = 1 //即配置AHBCLKCTL0[13] = 0，disable IOCON时钟

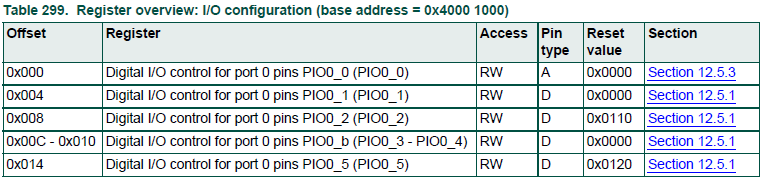


## 配置Pinmux

示例：配置P0\_1为GPIO，PULLUP，enable digital

IOCON->PIO[0][1] = 0 | 2 << 4 | 1 << 8 ;





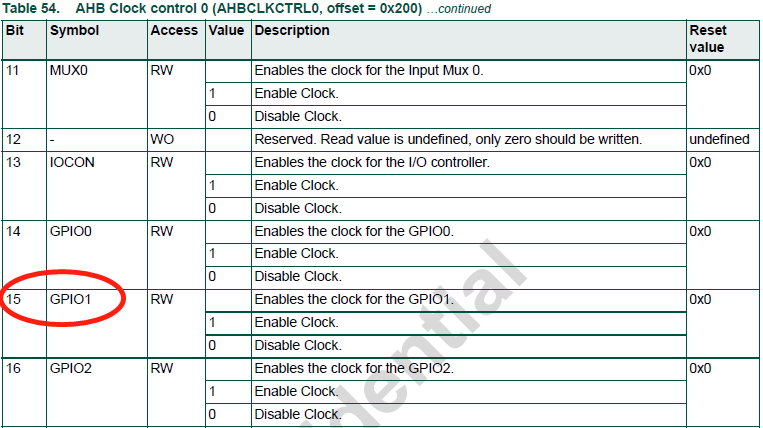
# GPIO

## 使能GPIO时钟

使能时钟的方式参考[使用辅助寄存器（AHBCLKCTRLSET/AHBCLKCTRLCLR）使能clock](#_使用辅助寄存器（AHBCLKCTRLSET/AHBCLKCTRLCLR)

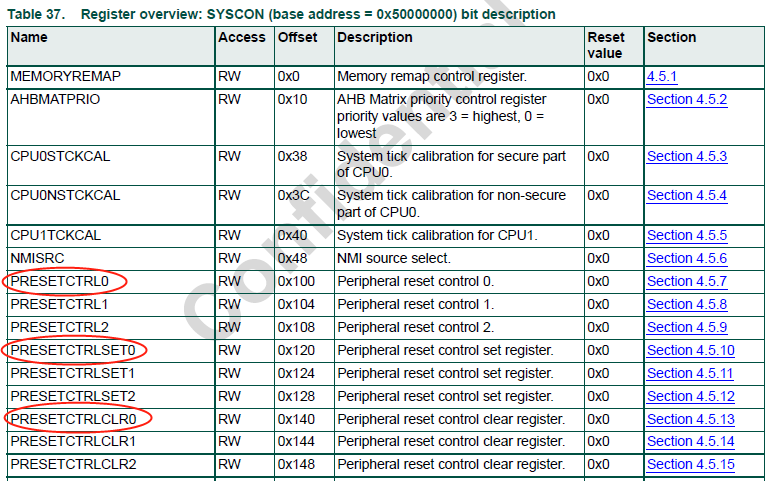
使能GPIO1\_4时钟

SYSCON->AHBCLKCTRLSET0 = 1 << 15



## Reset GPIO

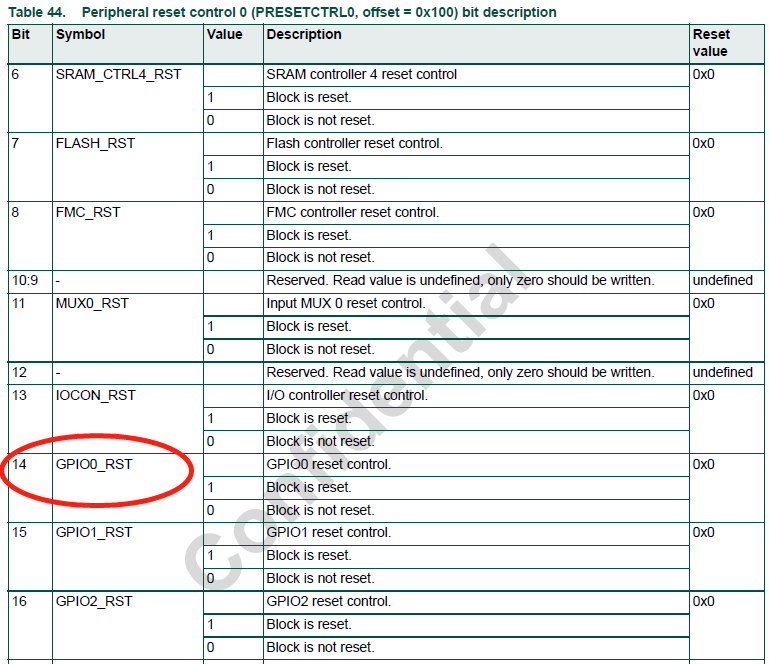
与clock的使能类似，SYSCON也提供了模块reset寄存器。每个reset寄存器也有相应的辅助寄存器



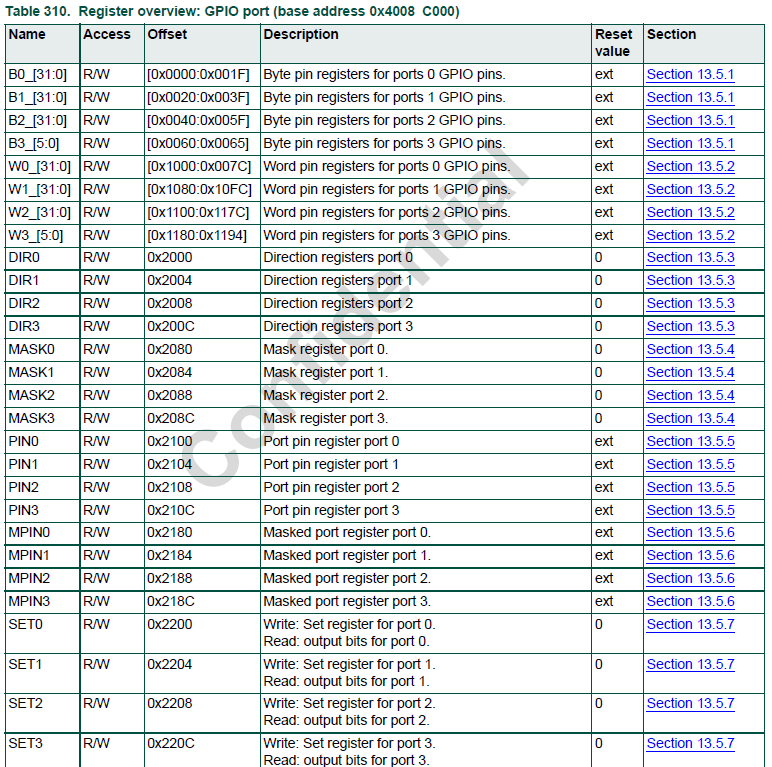
Reset GPIO1 :

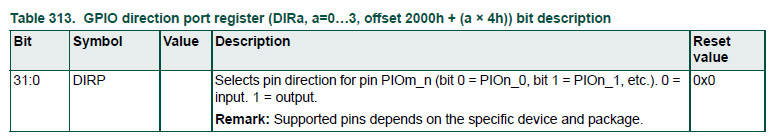
SYSCON->PRESETCTRLSET0 = 1 << 14 //reset GPIO0

SYSCON->PRESETCTRLCLR0 = 1 << 14 //清除reset标记



## 配置GPIO





GPIO->DIR1 = 1<<4 // P1\_4 配置为output

GPIO->CLR1 = 1 <<4 //清除P1\_4

GPIO->SET1 = 1 << 4 // P1\_4 置高电平

GPIO->NOT1 = 1 << 4 // 反转P1\_4