# 第二讲笔记

操作GPIO设备

可以直接操作寄存器

也可以通过自己封装的函数调用寄存器配置进而实现对GPIO的控制

0010 0010

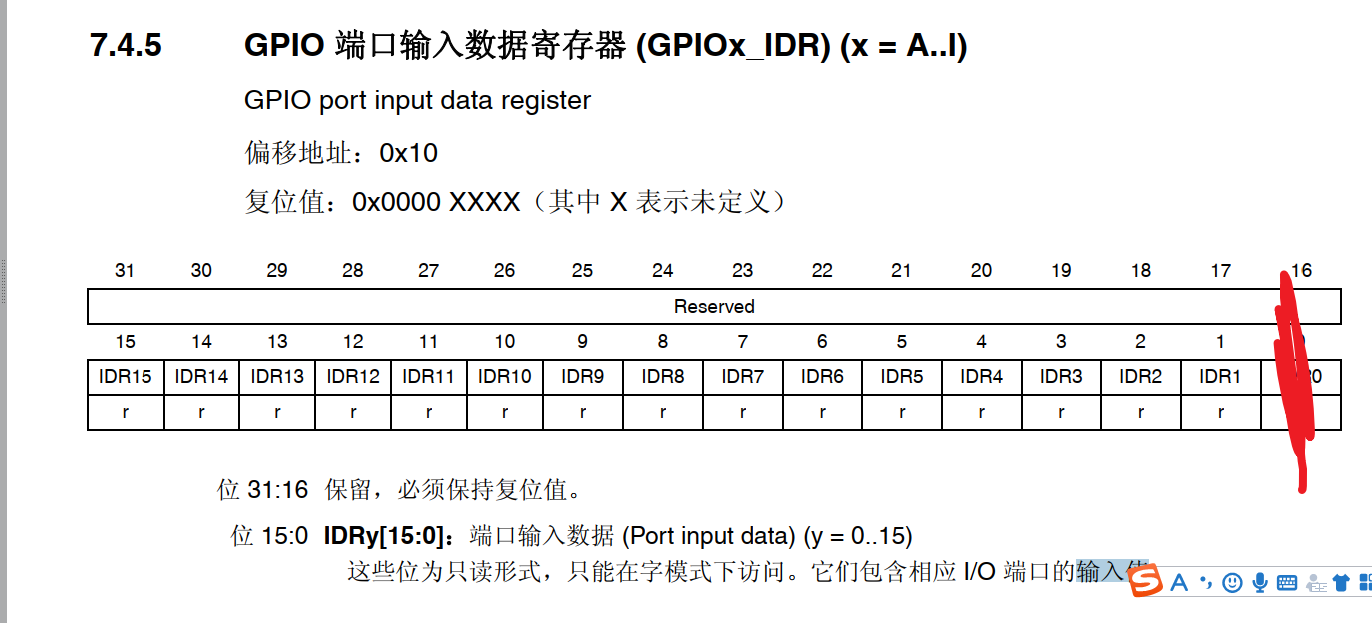
（0000 0010）&（1<<5）

0001 0000

0000 0000

0010 0000 真

0000 0001=(0010 0000>>5)真



If()

{

}

Else

{

}

typedef enum

{

GPIO\_Mode\_IN = 0x00, /\*!< GPIO Input Mode \*/

GPIO\_Mode\_OUT = 0x01, /\*!< GPIO Output Mode \*/

GPIO\_Mode\_AF = 0x02, /\*!< GPIO Alternate function Mode \*/

GPIO\_Mode\_AN = 0x03 /\*!< GPIO Analog Mode \*/

}GPIOMode\_TypeDef;

管理模式的枚举

typedef enum

{

GPIO\_Low\_Speed = 0x00, /\*!< Low speed \*/

GPIO\_Medium\_Speed = 0x01, /\*!< Medium speed \*/

GPIO\_Fast\_Speed = 0x02, /\*!< Fast speed \*/

GPIO\_High\_Speed = 0x03 /\*!< High speed \*/

}GPIOSpeed\_TypeDef;

管理速度的枚举

typedef enum

{

GPIO\_PuPd\_NOPULL = 0x00,

GPIO\_PuPd\_UP = 0x01,

GPIO\_PuPd\_DOWN = 0x02

}GPIOPuPd\_TypeDef;

管理上下拉模式