

GPIO

GPIOA/B/C/D/E/F/G,一共16X7=112个IO\
GPIOH.0, GPIOH.1 外加这2个IO\
一共114个IO口

GPIO工作方式

GPIO port mode register

偏移地址：0x00

复位值：

0xA800 0000（端口 A）

0x0000 0280（端口 B）

0x0000 0000（其它端口）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]	
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[1:0]	
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

位 2y:2y+1 **MODERy[1:0]**：端口 x 配置位 (Port x configuration bits) (y = 0..15)
这些位通过软件写入，用于配置 I/O 方向模式。
00：输入（复位状态）
01：通用输出模式
10：复用功能模式
11：模拟模式

输入

输入浮空、输入上拉、输入下拉、模拟（输入）、复用（输入）

输出

开漏式输出、推挽式输出、模拟（输出）、复用（输出）

最大输出速度：2MHZ、25MHZ、50MHZ、100MHZ

推挽输出

两个MOS管都激活。可以输出强高\低电平，驱动数字器件。

开漏输出

仅N-MOS管激活。只可以输出强低电平，高电平得靠外部电阻拉高。
要得到高电平状态需要上拉电阻。

复用

模拟

相关寄存器

每组GPIO端口的寄存器包括：

一个端口模式寄存器	(GPIOx_MODER)	}	4个32位配置寄存器
一个端口输出类型寄存器	(GPIOx_OTYPER)		
一个端口输出速度寄存器	(GPIOx_OSPEEDR)		
一个端口上拉下拉寄存器	(GPIOx_PUPDR)		
一个端口输入数据寄存器	(GPIOx_IDR)	}	2个32位数据寄存器
一个端口输出数据寄存器	(GPIOx_ODR)		
一个端口置位/复位寄存器	(GPIOx_BSRR)		
一个端口配置锁存寄存器	(GPIOx_LCKR)		
两个端口位复用功能寄存器	(GPIOx_AFR1 & GPIOx_AFR2)		

- ◆ 如果配置一个IO口需要2个位，那么刚好32位寄存器配置一组IO口16个IO口
- ◆ 如果配置一个IO口只需要1个位，一般高16位保留

如下图，左边这个32为地址，即为GPIOC的所有寄存器的基地址

存储器和总线架构

表 2. STM32F4xx 寄存器边界地址 (续)

边界地址	外设	总线	寄存器映射
0x4004 0000 - 0x4007 FFFF	USB OTG HS	AHB1	第 1130 页的第 31.12.6 节: OTG_HS 寄存器映射
0x4002 9000 - 0x4002 93FF	以太网 MAC		第 924 页的第 29.8.5 节: 以太网寄存器映射
0x4002 8C00 - 0x4002 8FFF			
0x4002 8800 - 0x4002 8BFF			
0x4002 8400 - 0x4002 87FF			
0x4002 8000 - 0x4002 83FF			
0x4002 6400 - 0x4002 67FF	DMA2		第 229 页的第 9.5.11 节: DMA 寄存器映射
0x4002 6000 - 0x4002 63FF	DMA1		
0x4002 4000 - 0x4002 4FFF	BKPSRAM		
0x4002 3C00 - 0x4002 3FFF	Flash 接口寄存器		第 3.8 节: Flash 接口寄存器
0x4002 3800 - 0x4002 3BFF	RCC		第 171 页的第 6.3.32 节: RCC 寄存器映射
0x4002 3000 - 0x4002 33FF	CRC		第 85 页的第 4.4.4 节: CRC 寄存器映射
0x4002 2000 - 0x4002 23FF	GPIOI		第 192 页的第 7.4.11 节: GPIO 寄存器映射
0x4002 1C00 - 0x4002 1FFF	GPIOH		
0x4002 1800 - 0x4002 1BFF	GPIOG		
0x4002 1400 - 0x4002 17FF	GPIOF		
0x4002 1000 - 0x4002 13FF	GPIOE		
0x4002 0C00 - 0x4002 0FFF	GPIOD		
0x4002 0800 - 0x4002 0BFF	GPIOC		
0x4002 0400 - 0x4002 07FF	GPIOB		
0x4002 0000 - 0x4002 03FF	GPIOA		
0x4001 5400 - 0x4001 57FF	SPI6	APB2	第 769 页的第 27.5.10 节: SPI 寄存器映射
0x4001 5000 - 0x4001 53FF	SPI5		
0x4001 4800 - 0x4001 4BFF	TIM11	APB2	第 481 页的第 16.6.11 节: TIM10/11/13/14 寄存器映射
0x4001 4400 - 0x4001 47FF	TIM10		第 472 页的第 16.5.14 节: TIM9/12 寄存器映射
0x4001 4000 - 0x4001 43FF	TIM9		
0x4001 3C00 - 0x4001 3FFF	EXTI	APB2	第 247 页的第 10.3.7 节: EXTI 寄存器映射
0x4001 3800 - 0x4001 3BFF	SYSCFG		第 199 页的第 8.2.9 节: SYSCFG 寄存器映射
0x4001 3400 - 0x4001 37FF	SPI4	APB2	第 769 页的第 27.5.10 节: SPI 寄存器映射
0x4001 3000 - 0x4001 33FF	SPI1	APB2	第 769 页的第 27.5.10 节: SPI 寄存器映射
0x4001 2C00 - 0x4001 2FFF	SDIO		第 819 页的第 28.9.16 节: SDIO 寄存器映射
0x4001 2000 - 0x4001 23FF	ADC1 - ADC2 - ADC3		第 286 页的第 11.13.18 节: ADC 寄存器映射
0x4001 1400 - 0x4001 17FF	USART6	APB2	第 720 页的第 26.6.8 节: USART 寄存器映射
0x4001 1000 - 0x4001 13FF	USART1		
0x4001 0400 - 0x4001 07FF	TIM8		
0x4001 0000 - 0x4001 03FF	TIM1		第 390 页的第 14.4.21 节: TIM1 和 TIM8 寄存器映射

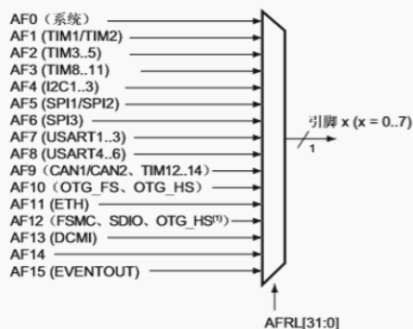
表 32. GPIO 寄存器映射和复位值

偏移	寄存器	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0x00	GPIOA_MODER	MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]		MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[1:0]		0
	Reset value	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	GPIOx_OTYPER (where x = A..I)	Reserved																	OT15	OT14	OT13	OT12	OT11	OT10	OT9	OT8	OT7	OT6	OT5	OT4	OT3	OT2	OT1	OT0
0x08	GPIOx_OSPEEDER (where x = A..I/ except B)	OSPEEDR15[1:0]		OSPEEDR14[1:0]		OSPEEDR13[1:0]		OSPEEDR12[1:0]		OSPEEDR11[1:0]		OSPEEDR10[1:0]		OSPEEDR9[1:0]		OSPEEDR8[1:0]		OSPEEDR7[1:0]		OSPEEDR6[1:0]		OSPEEDR5[1:0]		OSPEEDR4[1:0]		OSPEEDR3[1:0]		OSPEEDR2[1:0]		OSPEEDR1[1:0]		OSPEEDR0[1:0]		0
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	GPIOB_OSPEEDER	OSPEEDR15[1:0]		OSPEEDR14[1:0]		OSPEEDR13[1:0]		OSPEEDR12[1:0]		OSPEEDR11[1:0]		OSPEEDR10[1:0]		OSPEEDR9[1:0]		OSPEEDR8[1:0]		OSPEEDR7[1:0]		OSPEEDR6[1:0]		OSPEEDR5[1:0]		OSPEEDR4[1:0]		OSPEEDR3[1:0]		OSPEEDR2[1:0]		OSPEEDR1[1:0]		OSPEEDR0[1:0]		0
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
0x0C	GPIOA_PUPDR	PUPDR15[1:0]		PUPDR14[1:0]		PUPDR13[1:0]		PUPDR12[1:0]		PUPDR11[1:0]		PUPDR10[1:0]		PUPDR9[1:0]		PUPDR8[1:0]		PUPDR7[1:0]		PUPDR6[1:0]		PUPDR5[1:0]		PUPDR4[1:0]		PUPDR3[1:0]		PUPDR2[1:0]		PUPDR1[1:0]		PUPDR0[1:0]		0
	Reset value	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	GPIOB_PUPDR	PUPDR15[1:0]		PUPDR14[1:0]		PUPDR13[1:0]		PUPDR12[1:0]		PUPDR11[1:0]		PUPDR10[1:0]		PUPDR9[1:0]		PUPDR8[1:0]		PUPDR7[1:0]		PUPDR6[1:0]		PUPDR5[1:0]		PUPDR4[1:0]		PUPDR3[1:0]		PUPDR2[1:0]		PUPDR1[1:0]		PUPDR0[1:0]		0
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0x0C	GPIOx_PUPDR (where x = C..I)	PUPDR15[1:0]		PUPDR14[1:0]		PUPDR13[1:0]		PUPDR12[1:0]		PUPDR11[1:0]		PUPDR10[1:0]		PUPDR9[1:0]		PUPDR8[1:0]		PUPDR7[1:0]		PUPDR6[1:0]		PUPDR5[1:0]		PUPDR4[1:0]		PUPDR3[1:0]		PUPDR2[1:0]		PUPDR1[1:0]		PUPDR0[1:0]		0
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	GPIOx_IDR (where x = A..I)	Reserved																	IDR15	IDR14	IDR13	IDR12	IDR11	IDR10	IDR9	IDR8	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0
	Reset value																		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x14	GPIOx_ODR (where x = A..I)	Reserved																	ODR15	ODR14	ODR13	ODR12	ODR11	ODR10	ODR9	ODR8	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0
	Reset value																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	GPIOx_BSRR (where x = A..I)	BSR15	BSR14	BSR13	BSR12	BSR11	BSR10	BSR9	BSR8	BSR7	BSR6	BSR5	BSR4	BSR3	BSR2	BSR1	BSR0	BSR15	BSR14	BSR13	BSR12	BSR11	BSR10	BSR9	BSR8	BSR7	BSR6	BSR5	BSR4	BSR3	BSR2	BSR1	BSR0	
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

端口位复用功能寄存器

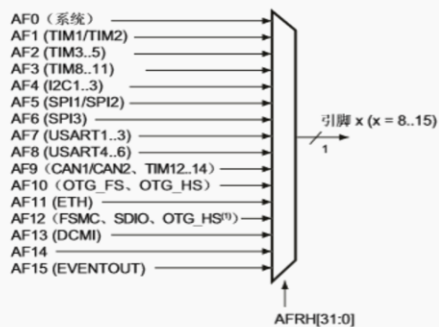
GPIOx_AFRL[31:0]

对于引脚 0 到引脚 7，GPIOx_AFRL[31:0] 寄存器会选择专用的复用功能



GPIOx_AFRH[31:0]

对于引脚 8 到引脚 15，GPIOx_AFRH[31:0] 寄存器会选择专用的复用功能



“端口复用器”的作用：最大限度的利用端口资源

相关函数

输入、输出、复用、模拟；输出包括：开漏、推挽

```
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_OUT;
GPIO_InitStructure.GPIO_OType = GPIO_OType_PP;
```

设置端口及引脚（函数）

```
void GPIO_SetBits(GPIO_TypeDef* GPIOx, uint16_t GPIO_Pin);  
void GPIO_ResetBits(GPIO_TypeDef* GPIOx, uint16_t GPIO_Pin);
```

AF 复用 GPIO 引脚

```
GPIO_PinAFConfig(GPIOB, GPIO_PinSource10, GPIO_AF_USART3); // PB10 复用为 USART3  
GPIO_PinAFConfig(GPIOB, GPIO_PinSource11, GPIO_AF_USART3); // PB11 复用为 USART3  
GPIO_PinAFConfig(GPIOB, GPIO_PinSource10, GPIO_AF_SPI2); // PB10 复用为 SPI2  
GPIO_PinAFConfig(GPIOB, GPIO_PinSource14, GPIO_AF_SPI2); // PB14 复用为 SPI2  
GPIO_PinAFConfig(GPIOB, GPIO_PinSource15, GPIO_AF_SPI2); // PB15 复用为 SPI2
```

作业

写出以下寄存器的32位地址：GPIOI_MODER、GPIOH_OTYPER、GPIOG_PUPDR。

答：在中文参考手册P53（查找基地址），在P192/193中（查找偏移地址）

GPIOI_MODER：GPIOI的基地址 + MODER的偏移地址=0x4002 2000 + 00 =0x4002 2000

GPIOH_OTYPER：GPIOH的基地址 + OTYPER的偏移地址=0x4002 1C00 + 04 =0x4002 1C04

GPIOG_PUPDR：GPIOG的基地址 + PUPDR的偏移地址=0x4002 1800 + 0C =0x4002 180C

请查到并写出以下片内外设或接口的相关寄存器端口地址：RCC_XX、U(S)ARTx_XX、EXTI_XX、TIMx_XX、RTC_XX、IWDG_XX、ADCx_XX、DACx_X、SPIx、I2Cx_XX。（不用写出）会找到它们各自的若干个寄存器的偏移地址、会推算出32位地址。

RCC_XX： 0x4002 3800\
USART6_XX： 0x4001 1400\
EXTI_XX： 0x4001 3C00\
TIM4_XX： 0x4000 0800\
RTC_XX： 0x4000 2800\
IWDG_XX： 0x4000 3000\
ADC1_XX： 0x4001 2000\
SPI4_XX： 0x4001 3400\
I2C1_XX： 0x4000 5400

GPIO有哪些相关寄存器，其中GPIOx_MODER/ GPIOx_AFRL的作用分别是什么？

GPIOx_MODER；GPIOx_OTYPER；GPIOx_OSPEEDER；GPIOx_OSPEEDER；
GPIOx_PUPDR；GPIOx_IDR；GPIOx_ODR；GPIOx_BSRR；GPIOx_LCKR；GPIOx_AFRL；
GPIOx_AFRH。

“STM32F407ZGT6.PDF数据手册”P56-60及P44 ~ P55 , 找出以下信号 , 占用哪些个GPIO线 ?

- 1) USART3_TX、 USART3_RX、 USART3_CK分别占用 : PD8/PD9/PD10; 或PC10/PC11/PC12;或PB10/PB11/PB12;\
- 2) TIM14_CH1占用 : PA7或PF9;\
- 3) I2C2_SDA、 I2C2_SCL 分别占用 : PB11/PB10; 或PH5/PH4; 或PF0/PF1;\
- 4) SPI1_SCK、 SPI1_MISO、 SPI1_MOSI分别占用 : PB3/PB4/PB5; 或PA5/ PA6/ PA7\
- 5) ADCx_IN3、 DAC1_OUT 分别占用 : PA3/ PA4