

AT32F423 GPIO Application Note

前言

这篇应用笔记介绍AT32F423的GPIO功能及固件驱动程序API的配置和使用，并对BSP例程的软件设计加以说明，同时演示使用方法并展示实验效果，供用户参考。

AT32F423xx的通用功能I/O (GPIO)提供了一系列与外部环境通讯的接口，可用于MCU与其他嵌入式设备之间通过数字或模拟方式的通讯。AT32F423系列的GPIO还提供了丰富I/O复用功能，能够使得多个外设可以同时工作，并且保证每个引脚在某一时刻只会连接到一个外设，从而避免了外设冲突的产生。

注：本应用笔记对应的代码是基于雅特力提供的V2.x.x 板级支持包（BSP）而开发，对于其他版本BSP，需要注意使用上的区别。

参考资料：

- AT32F423_Firmware_Library_V2.x.x\project\at_start_f423\examples\gpio
- RM_AT32F423 系列技术手册文档的通用和复用功能 I/O（GPIO 和 IOMUX）章节
- Datasheet_AT32F423 系列数据手册

支持型号列表：

支持型号	AT32F423xx
------	------------

目录

1	GPIO 特性.....	5
2	GPIO	6
2.1	GPIO toggle.....	8
2.2	IO 引脚的 5V or 3.3V 容忍	8
2.2.1	标准 3.3V 容忍引脚（TC）	8
2.2.2	带模拟功能 5 V 容忍引脚（FTa）	8
2.2.3	带 20 mA 吸入能力 5 V 容忍引脚（FTf）	9
2.2.4	5V 容忍引脚（FT）	9
3	IOMUX	10
3.1	I/O 复用功能输入/输出	10
3.2	特殊 I/O	20
3.2.1	调试复用引脚.....	20
3.2.2	振荡器复用引脚	20
3.2.3	电池供电域下的引脚.....	20
4	GPIO 固件驱动程序 API	22
4.1	输出模式.....	22
4.2	输入模式.....	22
4.3	模拟模式.....	22
4.4	复用模式.....	23
4.4.1	USART I/O 复用模式配置	23
4.4.2	TMR I/O 复用模式配置	24
4.4.3	I2C I/O 复用模式配置	24
5	版本历史	26

表目录

表 1. GPIO 配置表.....	7
表 2. TC 引脚示例	8
表 3. FTa 引脚示例	9
表 4. FTf 引脚示例.....	9
表 5. FT 引脚示例.....	9
表 6. 通过 GPIOA_MUX 寄存器配置端口 A 的复用功能	10
表 7. 通过 GPIOB_MUX 寄存器配置端口 B 的复用功能	12
表 8. 通过 GPIOC_MUX 寄存器配置端口 B 的复用功能.....	14
表 9. 通过 GPIOD_MUX 寄存器配置端口 D 的复用功能.....	16
表 10. 通过 GPIOE_MUX 寄存器配置端口 E 的复用功能	18
表 11. 通过 GPIOF_MUX 寄存器配置端口 F 的复用功能.....	20
表 12. 文档版本历史	26

图目录

图 1. GPIO 基本结构	6
图 2. I/O 翻转速度	8

1 GPIO 特性

AT32F423支持多达86个双向I/O引脚，这些引脚分为6组，分别为PA0-PA15、PB0-PB15、PC0-PC15、PD0-PD15、PE0-PE15、PF0-PF2 PF6 PF8-PF1 0、每个引脚都可以实现与外部的通讯、控制以及数据采集的功能。

- 每个引脚都可以软件配置成浮空输入、上拉/下拉输入、模拟输入/输出、通用推挽/开漏输出、复用推挽/开漏输出。
- 每个引脚都可以映射到16个外部中断
- 几乎所有I/O口可容忍5V
- 所有I/O口均为快速I/O，寄存器存取速度最高f_{AHB}
- 每个引脚都有独立的弱上拉/下拉功能。
- 每个引脚都可以软件配置输出驱动能力。
- 每个引脚的外设功能可以通过一个特定的操作锁定，以避免意外的写入I/O寄存器
- GPIO设置/清除寄存器（GPIOx_SCR）和GPIO位清除寄存器（GPIOx_CLR）为GPIO输出数据寄存器（GPIOx_ODT）提供位访问能力

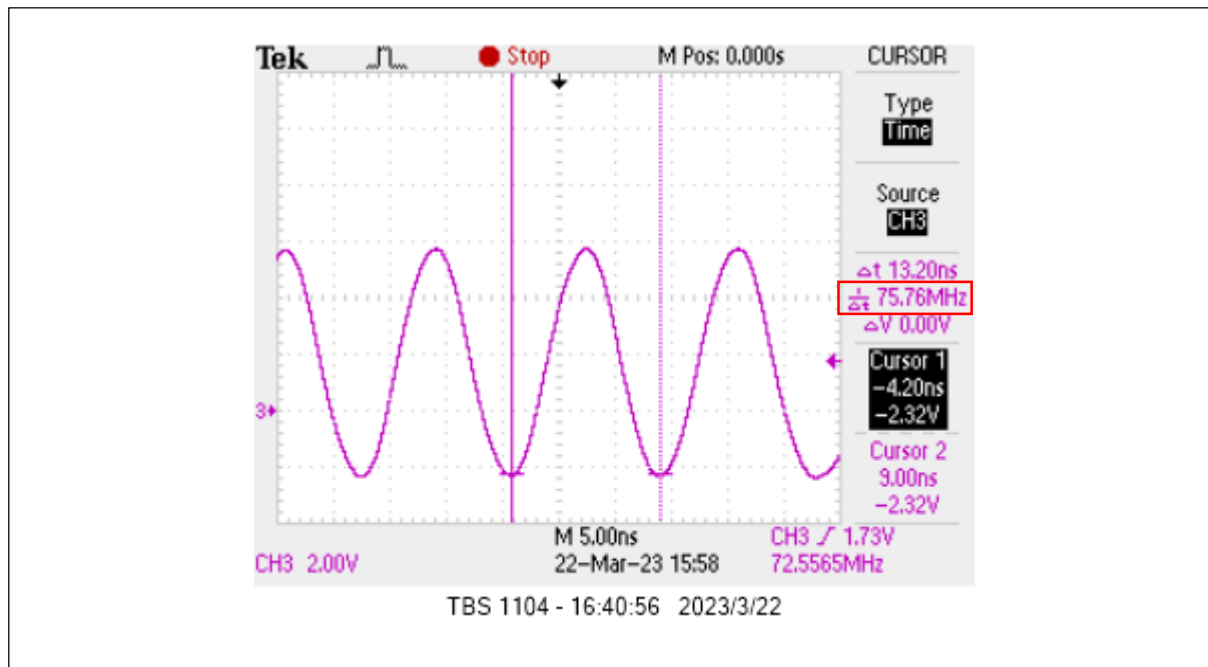
表 1. GPIO 配置表

CFGR[1: 0]	OMODE	ODRVR[1: 0]		PULL[1: 0]		I/O 配置	
01	0			0	0	通用输出	PP
	0			0	1	通用输出	PP+PU
	0			1	0	通用输出	PP+PD
	0			1	1	保留	
	1			0	0	通用输出	OD
	1			0	1	通用输出	OD+PU
	1			1	0	通用输出	OD+PD
	1			1	1	保留(通用输出 OD)	
10	0	ODRV[1: 0]		0	0	复用输出	PP
	0			0	1	复用输出	PP+PU
	0			1	0	复用输出	PP+PD
	0			1	1	保留	
	1			0	0	复用输出	OD
	1			0	1	复用输出	OD+PU
	1			1	0	复用输出	OD+PD
	1			1	1	保留	
00	x	x	x	0	0	输入	浮空
	x	x	x	0	1	输入	PU
	x	x	x	1	0	输入	PD
	x	x	x	1	1	保留(浮空输入)	
11	x	x	x	0	0	输入/输出	模拟
	x	x	x	0	1	保留	
	x	x	x	1	0		
	x	x	x	1	1		

2.1 GPIO toggle

AT32F423提供的I/O口均为快速I/O，寄存器存取速度最高为f_{AHB}，所以可以看到GPIO翻转频率能够轻松达到75MHz：

图 2. I/O 翻转速度



2.2 IO 引脚的 5V or 3.3V 容忍

2.2.1 标准 3.3V 容忍引脚 (TC)

所有振荡器用到的引脚都是标准3.3V容忍引脚。

- PA11/PA12
- PC14/PC15 (LEXT_IN/ OUT)
- PF0/PF1 (HEXT_IN/ OUT)

表 2. TC 引脚示例

引脚名称	引脚类型	IO电平	复用功能	附加功能
PF0	I/O	TC	TMR1_CH1 / I2C1_SDA	HEXT_IN

2.2.2 帶模擬功能 5 V 容忍引腳 (FTa)

ADC占用端口为带模拟功能5V容忍引脚。

- PA0 – PA7
- PB0 – PB2 PB10-PB15
- PC0 – PC5
- PE7

- FTa引脚设置为输入浮空、输入上拉、或输入下拉时，具有5V电平容忍特性；设置为模拟模式时，不具5V电平容忍特性，此时输入电平必须小于VDD + 0.3V

表 3. FTa 引脚示例

引脚名称	引脚名称	IO结构	复用功能	附加功能
PA0	I/O	FTa	TMR2_CH1 / TMR2_EXT / TMR9_CH2C / I2C2_SCL / USART2_RX / USART2_CTS / USART4_TX /	ADC_IN0 / TAMP2 / WKUP1

2.2.3 带 20 mA 吸入能力 5 V 容忍引脚（FTf）

部分I2C占用端口为带20 mA吸入能力的5 V容忍引脚，用以支持I2C的增强快速模式。

- PD12 – PD13

表 4. FTf 引脚示例

引脚名称	引脚名称	IO结构	复用功能	附加功能
PD12	I/O	FTf	TMR4_CH1 / I2C2_SCL / USART3_RTS_DE / USART8_CK_DE / XMC_A17	—

2.2.4 5V 容忍引脚（FT）

其余的GPIO都为5V容忍引脚。

表 5. FT 引脚示例

引脚名称	引脚名称	IO结构	复用功能	附加功能
PA8	I/O	FT	CLKOUT / TMR1_CH1 / TM9_BRK / I2C3_SCL / USART1_CK / USART2_TX / USART7_RX / OTGFS1_SOF	—

3 IOMUX

3.1 I/O 复用功能输入/输出

- 大多数外设共享同一个GPIO引脚（比如PA0，可作为TMR1_EXT / USART2_CTS / I2C2_SCL / USART4_TX..）
- 而对某个具体的GPIO引脚，在任意时刻只有一个外设能够与之相连
- 某些外设功能还可以重映射到其他引脚，从而使得能同时使用的外设数量更多

选择每个端口线的有效复用功能之一是由两个寄存器来决定的，分别是 GPIOx_MUXL 和 GPIOx_MUXH 复用功能寄存器。可根据应用的需求用这两寄存器连接复用功能模块到其他引脚。

表 6. 通过 GPIOA_MUX 寄存器配置端口 A 的复用功能

引脚名	MUX0	MUX1	MUX2	MUX3	MUX4	MUX5	MUX6	MUX7
PA0		TMR2_CH1 TMR2_EXT		TMR9_CH2C	I2C2_SCL		USART2_RX	USART2_CTS
PA1		TMR2_CH2		TMR9_CH1C	I2C2_SDA	I2C1_SMBA	SPI3_CS/I2S3_WS	USART2_RTS_ DE
PA2		TMR2_CH3		TMR9_CH1				USART2_TX
PA3		TMR2_CH4		TMR9_CH2		I2S2_MCK		USART2_RX
PA4					I2C1_SCL	SPI1_CS/I2S1_ WS	SPI3_CS/I2S3_WS	USART2_CK
PA5		TMR2_CH1 TMR2_EXT				SPI1_SCK/I2S1_ _CK	USART3_CK	USART3_RX
PA6		TMR1_BRK	TMR3_CH1			SPI1_MISO/I2S 1_MCK	I2S2_MCK	USART3_CTS
PA7		TMR1_CH1C	TMR3_CH2		I2C3_SCL	SPI1_MOSI/I2S 1_SD		USART3_TX
PA8	CLKOUT	TMR1_CH1		TMR9_BRK	I2C3_SCL			USART1_CK
PA9	CLKOUT	TMR1_CH2			I2C3_SMBA	SPI2_SCK/I2S2 _CK		USART1_TX
PA10	ERTC_REFIN	TMR1_CH3				SPI2_MOSI/I2S 2_SD		USART1_RX
PA11		TMR1_CH4			I2C2_SCL	SPI2_CS/I2S2_ WS	I2C1_SMBA	USART1_CTS
PA12		TMR1_EXT			I2C2_SDA	SPI2_MISO/I2S 2_MCK		USART1_RTS_ DE
PA13	JTMS SWDIO	IR_OUT			I2C1_SDA	I2S_SDEXT	SPI3_MISO/I2S3_MC K	
PA14	JTCK SWCLK				I2C1_SMBA		SPI3_MOSI/I2S3_SD	
PA15	JTDI	TMR2_CH1 TMR2_EXT				SPI1_CS/I2S1_ WS	SPI3_CS/I2S3_WS	USART1_TX

引脚名	MUX8	MUX9	MUX10	MUX11	MUX12	MUX13	MUX14	MUX15
PA0	USART4_TX							EVENTOUT
PA1	USART4_RX							EVENTOUT
PA2		CAN2_RX			XMC_D4			EVENTOUT
PA3		CAN2_TX			XMC_D5			EVENTOUT
PA4	USART6_TX	TMR14_CH1	OTGFS_OE		XMC_D6			EVENTOUT
PA5	USART6_RX	TMR13_CH1C			XMC_D7			EVENTOUT
PA6	USART3_RX	TMR13_CH1						EVENTOUT
PA7		TMR14_CH1						EVENTOUT
PA8	USART2_TX	USART7_RX	OTGFS_SOF					EVENTOUT
PA9	I2C1_SCL	TMR14_BRK	OTGFS_VBUS					EVENTOUT
PA10	I2C1_SDA		OTGFS_ID					EVENTOUT
PA11	USART6_TX	CAN1_RX						EVENTOUT
PA12	USART6_RX	CAN1_TX						EVENTOUT
PA13			OTGFS_OE					EVENTOUT
PA14	USART2_TX							EVENTOUT
PA15	USART2_RX	USART7_TX	USART4_RTS_DE		XMC_NE2			EVENTOUT

表 7. 通过 GPIOB_MUX 寄存器配置端口 B 的复用功能

引脚名	MUX0	MUX1	MUX2	MUX3	MUX4	MUX5	MUX6	MUX7
PB0		TMR1_CH2C	TMR3_CH3			SPI1_MISO / I2S1_MCK	SPI3_MOSI/I2S3_SD	USART2_RX
PB1		TMR1_CH3C	TMR3_CH4			SPI1_MOSI / I2S1_SD	SPI2_SCK/I2S2_CK	USART2_CK
PB2		TMR2_CH4	TMR3_EXT		I2C3_SMBA		SPI3_MOSI/I2S3_SD	
PB3	JTDO SWO	TMR2_CH2			I2C2_SDA	SPI1_SCK/I2S1_CK	SPI3_SCK/I2S3_CK	USART1_RX
PB4	JNTRST		TMR3_CH1	TMR11_BRK	I2C3_SDA	SPI1_MISO/I2S1_MCK	SPI3_MISO/I2S3_MCK	USART1_CTS
PB5			TMR3_CH2	TMR10_BRK	I2C3_SMBA	SPI1_MOSI/I2S1_SD	SPI3_MOSI/I2S3_SD	USART1_CK
PB6			TMR4_CH1	TMR10_CH1C	I2C1_SCL	I2S1_MCK	SPI3_CS / I2S3_WS	USART1_TX
PB7			TMR4_CH2	TMR11_CH1C	I2C1_SDA		SPI3_SCK / I2S3_CK	USART1_RX
PB8		TMR2_CH1 TMR2_EXT	TMR4_CH3	TMR10_CH1	I2C1_SCL		SPI3_MISO / I2S3_MCK	USART1_TX
PB9	IR_OUT	TMR2_CH2	TMR4_CH4	TMR11_CH1	I2C1_SDA	SPI2_CS/I2S2_WS	SPI3_MOSI / I2S3_SD	I2C2_SDA
PB10		TMR2_CH3			I2C2_SCL	SPI2_SCK/I2S2_CK	I2S3_MCK	USART3_TX
PB11		TMR2_CH4			I2C2_SDA			USART3_RX
PB12		TMR1_BRK		TMR12_BRK	I2C2_SMBA	SPI2_CS/I2S2_WS	SPI3_SCK/I2S3_CK	
PB13	CLKOUT	TMR1_CH1C		TMR12_CH1C	I2C3_SMBA	SPI2_SCK/I2S2_CK		I2C3_SCL
PB14		TMR1_CH2C			I2C3_SDA	SPI2_MISO/I2S2_MCK	I2S_SDEXT	USART3_RTS _DE
PB15	ERTC_REFIN	TMR1_CH3C		TMR12_CH1C	I2C3_SCL	SPI2_MOSI/I2S2_SD		

引脚名	MUX8	MUX9	MUX10	MUX11	MUX12	MUX13	MUX14	MUX15
PB0	USART3_CK							EVENTOUT
PB1	USART3_RTS _DE	TMR14_CH1						EVENTOUT
PB2		TMR14_CH1C						EVENTOUT
PB3	USART1_RTS _DE	USART7_RX	USART5_TX					EVENTOUT
PB4	I2S_SDEXT	USART7_TX	USART5_RX					EVENTOUT
PB5	USART5_RX	CAN2_RX	USART5_RTS_DE					EVENTOUT
PB6	USART5_TX	CAN2_TX	USART4_CK					EVENTOUT
PB7	USART4_CTS				XMC_NADV			EVENTOUT
PB8	USART5_RX	CAN1_RX						EVENTOUT
PB9	USART5_TX	CAN1_TX	I2S1_MCK					EVENTOUT
PB10					XMC_NOE			EVENTOUT
PB11		TMR13_BRK						EVENTOUT
PB12	USART3_CK	CAN2_RX			XMC_D13			EVENTOUT
PB13	USART3_CTS	CAN2_TX						EVENTOUT
PB14		TMR12_CH1			XMC_D0			EVENTOUT
PB15		TMR12_CH2						EVENTOUT

表 8. 通过 GPIOC_MUX 寄存器配置端口 B 的复用功能

引脚名	MUX0	MUX1	MUX2	MUX3	MUX4	MUX5	MUX6	MUX7
PC0					I2C3_SCL			I2C1_SCL
PC1					I2C3_SDA	SPI3_MOSI/I2S3_SD	SPI2_MOSI/I2S2_SD	I2C1_SDA
PC2						SPI2_MISO/I2S2_MCK	I2S_SDEXT	
PC3						SPI2_MOSI/I2S2_SD		
PC4				TMR9_CH1		I2S1_MCK		USART3_TX
PC5				TMR9_CH2	I2C1_SMBA			USART3_RX
PC6		TMR1_CH1	TMR3_CH1		I2C1_SCL	I2S2_MCK		
PC7		TMR1_CH2	TMR3_CH2		I2C1_SDA	SPI2_SCK/I2S2_CK	I2S3_MCK	
PC8		TMR1_CH3	TMR3_CH3					USART8_TX
PC9	CLKOUT	TMR1_CH4	TMR3_CH4		I2C3_SDA			USART8_RX
PC10							SPI3_SCK/I2S3_CK	USART3_TX
PC11						I2S_SDEXT	SPI3_MISO/I2S3_MCK	USART3_RX
PC12				TMR11_CH1	I2C2_SDA		SPI3_MOSI/I2S3_SD	USART3_CK
PC13								
PC14								
PC15								

引脚名	MUX8	MUX9	MUX10	MUX11	MUX12	MUX13	MUX14	MUX15
PC0	USART6_TX	USART7_TX						EVENTOUT
PC1	USART6_RX	USART7_RX						EVENTOUT
PC2	USART8_TX				XMC_NWE			EVENTOUT
PC3	USART8_RX				XMC_A0			EVENTOUT
PC4		TMR13_CH1			XMC_NE4			EVENTOUT
PC5		TMR13_CH1C			XMC_NOE			EVENTOUT
PC6	USART6_TX	USART7_TX			XMC_D1			EVENTOUT
PC7	USART6_RX	USART7_RX			XMC_NADV			EVENTOUT
PC8	USART6_CK							EVENTOUT
PC9	I2C1_SDA		OTGFS_OE					EVENTOUT
PC10	USART4_TX							EVENTOUT
PC11	USART4_RX				XMC_D2			EVENTOUT
PC12	USART4_CK		USART5_TX		XMC_D3			EVENTOUT
PC13								EVENTOUT
PC14								EVENTOUT
PC15								EVENTOUT

表 9. 通过 GPIOD_MUX 寄存器配置端口 D 的复用功能

引脚名	MUX0	MUX1	MUX2	MUX3	MUX4	MUX5	MUX6	MUX7
PD0							SPI3_MOSI/I2S3_SD	SPI2_CS/I2S2_WS
PD1							SPI2_SCK/I2S2_CK	SPI2_CS/I2S2_WS
PD2			TMR3_EXT					USART3_RTS_DE
PD3						SPI2_SCK/I2S2_CK	SPI2_MISO/I2S2_MCK	USART2_CTS
PD4							SPI2_MOSI/I2S2_SD	USART2_RTS_DE
PD5								USART2_TX
PD6						SPI3_MOSI/I2S3_SD		USART2_RX
PD7								USART2_CK
PD8								USART3_TX
PD9								USART3_RX
PD10								USART3_CK
PD11					I2C2_SMBA			USART3_CTS
PD12			TMR4_CH1		I2C2_SCL			USART3_RTS_DE
PD13			TMR4_CH2		I2C2_SDA			
PD14			TMR4_CH3		I2C3_SCL			
PD15			TMR4_CH4		I2C3_SDA			

引脚名	MUX8	MUX9	MUX10	MUX11	MUX12	MUX13	MUX14	MUX15
PD0	USART4_RX	CAN1_RX			XMC_D2			EVENTOUT
PD1	USART4_TX	CAN1_TX			XMC_D3			EVENTOUT
PD2	USART5_RX				XMC_NWE			EVENTOUT
PD3					XMC_CLK			EVENTOUT
PD4					XMC_NOE			EVENTOUT
PD5					XMC_NWE			EVENTOUT
PD6					XMC_NWAIT			EVENTOUT
PD7					XMC_NE1			EVENTOUT
PD8		TMR12_CH2C			XMC_D13			EVENTOUT
PD9					XMC_D14			EVENTOUT
PD10	USART4_TX				XMC_D15			EVENTOUT
PD11					XMC_A16			EVENTOUT
PD12	USART8_CK_ RTS_DE				XMC_A17			EVENTOUT
PD13	USART8_TX				XMC_A18			EVENTOUT
PD14	USART8_RX				XMC_D0			EVENTOUT
PD15		USART7_CK_RTS_ DE			XMC_D1			EVENTOUT

表 10. 通过 GPIOE_MUX 寄存器配置端口 E 的复用功能

引脚名	MUX0	MUX1	MUX2	MUX3	MUX4	MUX5	MUX6	MUX7
PE0			TMR4_EXT					
PE1		TMR1_CH2C						
PE2			TMR3_EXT	TMR9_BRK				
PE3			TMR3_CH1	TMR9_CH2C				
PE4			TMR3_CH2	TMR9_CH1C				
PE5			TMR3_CH3	TMR9_CH1				
PE6			TMR3_CH4	TMR9_CH2				
PE7		TMR1_EXT						
PE8		TMR1_CH1C						
PE9		TMR1_CH1						
PE10		TMR1_CH2C						
PE11		TMR1_CH2						
PE12		TMR1_CH3C				SPI1_CS/I2S1_WS		
PE13		TMR1_CH3				SPI1_SCK/I2S1_CK		
PE14		TMR1_CH4				SPI1_MISO/I2S1_MCK		
PE15		TMR1_BRK				SPI1_MOSI/I2S1_SD		

引脚名	MUX8	MUX9	MUX10	MUX11	MUX12	MUX13	MUX14	MUX15
PE0	USART8_RX	TMR13_CH1			XMC_LB			EVENTOUT
PE1	USART8_TX	TMR14_CH1			XMC_UB			EVENTOUT
PE2		TMR14_CH1C			XMC_A23			EVENTOUT
PE3		TMR14_BRK			XMC_A19			EVENTOUT
PE4					XMC_A20			EVENTOUT
PE5					XMC_A21			EVENTOUT
PE6					XMC_A22			EVENTOUT
PE7	USART5_CK	USART7_RX			XMC_D4			EVENTOUT
PE8	USART4_TX	USART7_TX			XMC_D5			EVENTOUT
PE9	USART4_RX				XMC_D6			EVENTOUT
PE10	USART5_TX				XMC_D7			EVENTOUT
PE11	USART5_RX				XMC_D8			EVENTOUT
PE12					XMC_D9			EVENTOUT
PE13					XMC_D10			EVENTOUT
PE14					XMC_D11			EVENTOUT
PE15					XMC_D12			EVENTOUT

表 11. 通过 GPIOF_MUX 寄存器配置端口 F 的复用功能

引脚名	MUX0	MUX1	MUX2	MUX3	MUX4	MUX5	MUX6	MUX7
PF0		TMR1_CH1			I2C1_SDA			
PF1		TMR1_CH2C			I2C1_SCL	SPI2_CS / I2S2_WS		
PF2						SPI2_SCK/ I2S2_CK		
PF6		TMR2_CH1			I2C2_SCL			
PF8		TMR2_CH2			I2C2_SDA			
PF9			TMR4_CH1					
PF10			TMR4_CH2					

引脚名	MUX8	MUX9	MUX10	MUX11	MUX12	MUX13	MUX14	MUX15
PF0								EVENTOUT
PF1								EVENTOUT
PF2		USART7_CK_ RTS_DE						EVENTOUT
PF6		USART7_RX						EVENTOUT
PF8		USART7_TX						EVENTOUT
PF9	USART6_TX	TMR12_CH1						EVENTOUT
PF10	USART6_RX	TMR12_CH2						EVENTOUT

3.2 特殊 I/O

3.2.1 调试复用引脚

- 在复位时，和复位后不像其他GPIO一样处于浮空输入状态，而是处于复用模式
- PA13: SWDIO，复用上拉
- PA14: SWCLK，复用下拉

3.2.2 振荡器复用引脚

- 振荡器关闭的状态下（复位后的默认状态），相关引脚可用作GPIO
- 振荡器使能状态下，相应引脚的GPIO配置无效
- 振荡器处于bypass模式（使用外部时钟源）时，LEXT_IN/HEXT_IN为振荡器时钟输入引脚，LEXT_OUT/HEXT_OUT可做GPIO使用

3.2.3 电池供电域下的引脚

- 电池供电域下的引脚包括PC13、PC14以及PC15，电池供电域由VDD供电。
- PC13可以作为通用I/O口、TAMPER引脚、ERTC校准时钟、ERTC闹钟或秒输出，PC14和

PC15可以用于GPIO或LEXT引脚。（PC13至PC15作为I/O口的速度必须限制在2MHz以下，最大负载为30pF，而且这些I/O口绝对不能当作电流源）。

4 GPIO 固件驱动程序 API

Artery 提供的固件驱动程序包含了一系列固件函数来管理 GPIO 的下列功能：

- 初始化配置
- 读取输入端口或某个输入引脚
- 读取输出端口或某个输出引脚
- 设置或清除某个引脚的输出
- 锁定引脚
- 引脚的复用功能配置

注：所有 project 都是基于 keil 5 而建立，若用户需要在其他编译环境上使用，请参考 AT32xxx_Firmware_Library_V2.x.x\project\at_start_xxx\templates 中各种编译环境（例如 IAR6/7, keil 4/5）进行简单修改即可。

4.1 输出模式

GPIO 提供了两种不同类型的输出模式分别是，推挽输出以及开漏输出，下面是输出模式的配置示例：

```
gpio_init_struct.gpio_pins = GPIO_PINS_x;  
gpio_init_struct.gpio_mode = GPIO_MODE_OUTPUT; /* 选择输出 */  
gpio_init_struct.gpio_pull = GPIO_PULL_NONE; /* 无、上拉或下拉 */  
gpio_init_struct.gpio_out_type = GPIO_OUTPUT_PUSH_PULL; /* 选择推挽或开漏 */  
gpio_init_struct.gpio_drive_strength = GPIO_DRIVE_STRENGTH_STRONGER;  
gpio_init(GPIOy, &gpio_init_struct);
```

4.2 输入模式

GPIO 提供了三种不同类型的输入模式分别是，浮空输入、上拉输入以及下拉输入，下面是输入模式的配置示例：

```
gpio_init_struct.gpio_pins = GPIO_PINS_x;  
gpio_init_struct.gpio_mode = GPIO_MODE_INPUT; /* 选择输入 */  
gpio_init_struct.gpio_pull = GPIO_PULL_NONE; /* 无、上拉或下拉 */  
gpio_init_struct.gpio_drive_strength = GPIO_DRIVE_STRENGTH_STRONGER;  
gpio_init(GPIOy, &gpio_init_struct);
```

4.3 模拟模式

当需要使用 ADC 通道作为输入时，需要将相应的引脚配置为模拟模式，下面是模拟模式的配置示例：

```
gpio_init_struct.gpio_pins = GPIO_PINS_x;
```

```
gpio_init_struct.gpio_mode = GPIO_MODE_ANALOG; /* 选择模拟输入 */
gpio_init_struct.gpio_pull = GPIO_PULL_NONE; /* 无、上拉或下拉 */
gpio_init_struct.gpio_drive_strength = GPIO_DRIVE_STRENGTH_STRONGER;
gpio_init(GPIOy, &gpio_init_struct);
```

4.4 复用模式

1. 不论使用何种外设模式，都必须将 I/O 配置为复用功能，之后系统才能正确使用 I/O（输入或输出）。
2. I/O 引脚通过复用器连接到相应的外设，该复用器一次只允许一个外设的复用功能 (MUX) 连接到 I/O 引脚。这样便可确保共用同一个 I/O 引脚的外设之间不会发生冲突。每个 I/O 引脚都有一个复用器，该复用器具有 16 路复用功能输入/输出（MUX0 到 MUX15），可通过 `gpio_pin_mux_config()` 函数对这些引脚进行配置：
 - 复位后，所有 I/O 都会连接到系统的复用功能 0 (MUX0)
 - 通过配置 MUX1 到 MUX15 可以映射外设的复用功能
3. 除了这种灵活的 I/O 复用架构之外，各外设还具有映射到不同 I/O 引脚的复用功能，这可以针对不同器件封装优化外设 I/O 功能的数量；例如，可将 USART2_TX 引脚映射到 PA2 或 PA14 引脚上。
4. 配置过程：
 - 使用 `gpio_pin_mux_config()` 函数将引脚连接到所需的外设复用功能 (MUX)，例如配置 PA0 作为 TMR1_EXT 输入
`gpio_pin_mux_config(GPIOA, GPIO_PINS_SOURCE0, GPIO_MUX_4);`
 - 使用 `GPIO_Init()` 函数配置 I/O 引脚：
 - 通过以下方式配置复用功能模式下的所需引脚
`gpio_init_struct.gpio_mode = GPIO_MODE_MUX;`
 - 通过以下成员选择类型、上拉/下拉和驱动力
`gpio_out_type`、`gpio_pull` 和 `gpio_drive_strength` 成员

根据上述配置过程，下面将介绍几种外设的常用配置示例。

4.4.1 USART I/O 复用模式配置

```
/* 使能 GPIOA 的时钟 */
crm_periph_clock_enable(CRM_GPIOA_PERIPH_CLOCK, TRUE);

/* 将 PA9 连接到 USART1_Tx */
gpio_pin_mux_config(GPIOA, GPIO_PINS_SOURCE9, GPIO_MUX_7);

/* 将 PA10 连接到 USART1_Rx */
gpio_pin_mux_config(GPIOA, GPIO_PINS_SOURCE10, GPIO_MUX_7);

/* 将 USART1_Tx 和 USART1_Rx 配置为复用功能 */
```

```
gpio_init_struct.gpio_pins = GPIO_PINS_9 | GPIO_PINS_10;  
gpio_init_struct.gpio_mode = GPIO_MODE_MUX;  
gpio_init_struct.gpio_pull = GPIO_PULL_NONE;  
gpio_init_struct.gpio_out_type = GPIO_OUTPUT_PUSH_PULL;  
gpio_init_struct.gpio_drive_strength = GPIO_DRIVE_STRENGTH_STRONGER;  
gpio_init(GPIOA, &gpio_init_struct);
```

4.4.2 TMR I/O 复用模式配置

```
/* 使能 GPIOA 的时钟 */  
crm_periph_clock_enable(CRM_GPIOA_PERIPH_CLOCK, TRUE);  
  
/* 将 PA6 连接到 TMR1_BRK */  
gpio_pin_mux_config(GPIOA, GPIO_PINS_SOURCE6, GPIO_MUX_1);  
/* 将 PA8 连接到 TMR1_CH1 */  
gpio_pin_mux_config(GPIOA, GPIO_PINS_SOURCE8, GPIO_MUX_1);  
/* 将 PA12 连接到 TMR1_EXT */  
gpio_pin_mux_config(GPIOA, GPIO_PINS_SOURCE12, GPIO_MUX_1);  
  
/* 将以上TMR1引脚(PA6/PA8/PA12)配置为复用功能 */  
gpio_init_struct.gpio_pins = GPIO_PINS_6 | GPIO_PINS_8 | GPIO_PINS_12;  
gpio_init_struct.gpio_mode = GPIO_MODE_MUX;  
gpio_init_struct.gpio_pull = GPIO_PULL_NONE;  
gpio_init_struct.gpio_out_type = GPIO_OUTPUT_PUSH_PULL;  
gpio_init_struct.gpio_drive_strength = GPIO_DRIVE_STRENGTH_STRONGER;  
gpio_init(GPIOA, &gpio_init_struct);
```

4.4.3 I2C I/O 复用模式配置

```
/* 使能 GPIOB 的时钟 */  
crm_periph_clock_enable(CRM_GPIOB_PERIPH_CLOCK, TRUE);  
  
/* 将 PB6 连接到 I2C1_SCL */  
gpio_pin_mux_config(GPIOB, GPIO_PINS_SOURCE6, GPIO_MUX_4);  
/* 将 PB7 连接到 I2C1_SDA */  
gpio_pin_mux_config(GPIOB, GPIO_PINS_SOURCE7, GPIO_MUX_4);  
/* 将 I2C1_SCL 和 I2C1_SDA 配置为复用功能，注意I2C引脚的输出类型应为开漏输出 */
```



```
gpio_init_struct.gpio_pins = GPIO_PINS_6 | GPIO_PINS_7;  
gpio_init_struct.gpio_mode = GPIO_MODE_MUX;  
gpio_init_struct.gpio_pull = GPIO_PULL_NONE;  
gpio_init_struct.gpio_out_type = GPIO_OUTPUT_OPEN_DRAIN;  
gpio_init_struct.gpio_drive_strength = GPIO_DRIVE_STRENGTH_STRONGER;  
gpio_init(GPIOB, &gpio_init_struct);
```

5 版本历史

表 12. 文档版本历史

日期	版本	变更
2023.03.20	2.0.0	最初版本

重要通知 - 请仔细阅读

买方自行负责对本文所述雅特力产品和服务的选择和使用，雅特力概不承担与选择或使用本文所述雅特力产品和服务相关的任何责任。

无论之前是否有过任何形式的表示，本文档不以任何方式对任何知识产权进行任何明示或默示的授权或许可。如果本文档任何部分涉及任何第三方产品或服务，不应被视为雅特力授权使用此类第三方产品或服务，或许可其中的任何知识产权，或者被视为涉及以任何方式使用任何此类第三方产品或服务或其中任何知识产权的保证。

除非在雅特力的销售条款中另有说明，否则，雅特力对雅特力产品的使用和/或销售不做任何明示或默示的保证，包括但不限于有关适销性、适合特定用途(及其依据任何司法管辖区的法律的对应情况)，或侵犯任何专利、版权或其他知识产权的默示保证。

雅特力产品并非设计或专门用于下列用途的产品：(A) 对安全性有特别要求的应用，如：生命支持、主动植入设备或对产品功能安全有要求的系统；(B) 航空应用；(C) 汽车应用或汽车环境；(D) 航天应用或航天环境，且/或(E) 武器。因雅特力产品不是为前述应用设计的，而采购商擅自将其用于前述应用，即使采购商向雅特力发出了书面通知，风险由购买者单独承担，并且独力负责在此类相关使用中满足所有法律和法规要求。

经销的雅特力产品如有不同于本文档中提出的声明和/或技术特点的规定，将立即导致雅特力针对本文所述雅特力产品或服务授予的任何保证失效，并且不应以任何形式造成或扩大雅特力的任何责任。

© 2023 雅特力科技 保留所有权利