RP2040(树莓派Pico) PIO - 指令集描述

之前已经说过,PIO支持的多条指令.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JMP	0	0	0	Delay/side-set					Condition			Address				
WAIT	0	0	1	Delay/side-set					Pol	Source		Index				
IN	0	1	0	Delay/side-set					Source			Bit count				
OUT	0	1	1	Delay/side-set					Destination			Bit count				
PUSH	1	0	0	Delay/side-set					0	IfF	Blk	0	0	0	0	0
PULL	1	0	0	Delay/side-set					1	IfE	Blk	0	0	0	0	0
MOV	1	0	1	Delay/side-set					Destination			Op		Source		
IRQ	1	1	0	Delay/side-set					0	Clr	Wait	Index				
SET	1	1	1	Delay/side-set					Destination			Data				

JMP命令 => JMP (条件) [地址],条件可以是如下数值:

- 000 = 无条件(默认)
- !X = X 寄存器为0
- X-- = X 寄存器在减1后非零
- !Y = Y 寄存器为0
- Y-- = Y 寄存器在减1后非零
- X!=Y = 两者不等
- PIN = 引脚状态
- !OSRE = 输出寄存器非空

WAIT命令 => wait [状态,1/0] [gpio/pin/irq] [num],这个命令如下填充:

- 状态 = 需要等待当前标志变为1或者0.
- num = 当是gpio的时候,这里指的是实际GPIO号,当是pin的时候,这里是sm pin,当然irq的时候,这里指的是中断号.

IN命令 => in [源], [位数],从源输入数据到ISR,这个命令填充如下:

- 源可以是:PINS,X,Y,NULL(全0),ISR,OSR
- 位数:1 - 32位

OUT命令 => out [目标], [位数],从OSR输出内容到目的地,这个命令填充如下:

- 目标可以是:PINS,X,Y,NULL(全0),PINDIR,PC,ISR,OSR
- 位数:1 - 32位

PUSH/PULL 命令 => push/pull [iffull/ifempty] [block/unblock],主要是推送/抓取移位寄存器的数据,这个命令填充如下:

- 如果iffull/ifempty为1,则如果队列满(对PUSH而言)/空(对PULL而言),则不进行任何操作,默认0.
- 如果block,则阻塞等待直至有效.

MOV命令 => mov [目标] (操作) [源],其中操作是可选的,如果操作是!或者~,则数据取反,如果操作是::则按位翻转.

IRQ 命令 => irq (set/nowait/wait/clear) [中断号],具体参数如下:

- irq / irq set / irq set nowait => 不等待IRQ
- irq wait => 等待IRQ清除
- irq clear => 清除IRQ
- 中断号,范围0 ~ 7

SET 命令 => set [目的地] [值],目的地可以是PINS,PINDIR,X,Y,值范围可以是0 - 31.