SoC外设

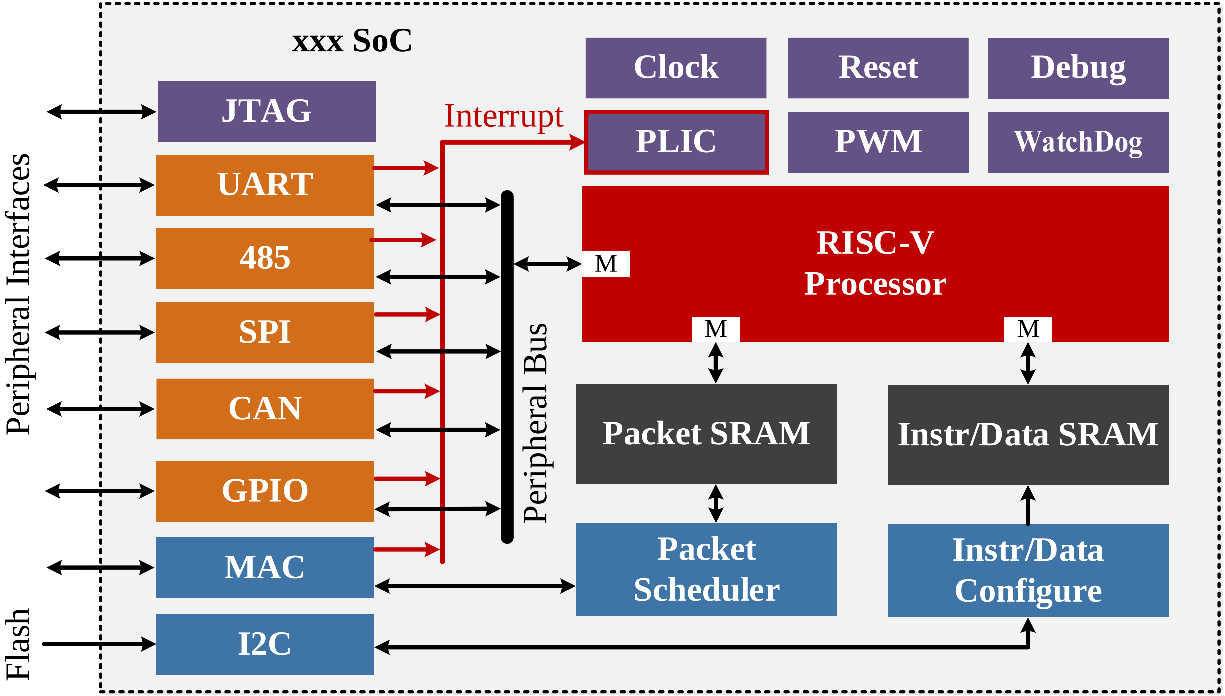
厉俊男

主要内容

- SoC接口设计
  - SoC外设接口组成
  - SoC外设接口定义
  - SoC外设寄存器定义
- 外设中断设计
  - RISC-V中断规范

SoC外设接口基本组成

- ■ 接口类型
  - □ 数据接口：CPU通过数据接口读写外设
  - □ 控制接口：外设通过控制接口向CPU发送中断请求
  - □ 调试接口：用户同通过调试接口监测CPU运行状态



SoC外设接口工作原理（1/2）

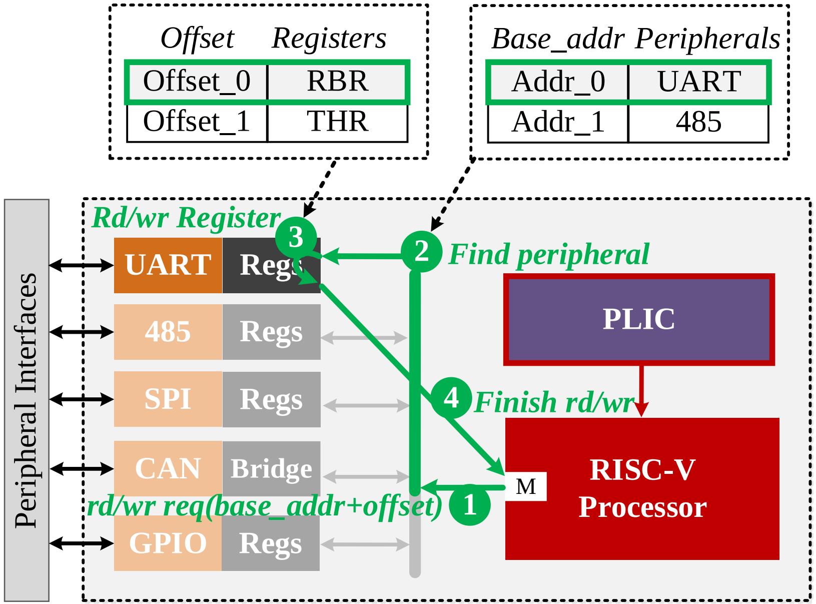
- 读/写数据接口处理流程

  ① CPU发起读/写请求，携带待读/写存储单元的地址信息（外设基地址+寄存器偏移）

  ② 数据总线通过比对基地址获取待访问外设，如UART

  ③ 外设通过查询寄存器偏移地址信息获得待访问的寄存器，如UART_RBR（Receive Buffer Register，接收缓存寄存器）

  ④ 完成写操作，或将读取的寄存器内容返回CPU



Offset	*Registers*
Offset_0	RBR
Offset_1	THR

Base_addr	*Peripherals*
Addr_0	UART
Addr_1	485

SoC外设接口工作原理（2/2）

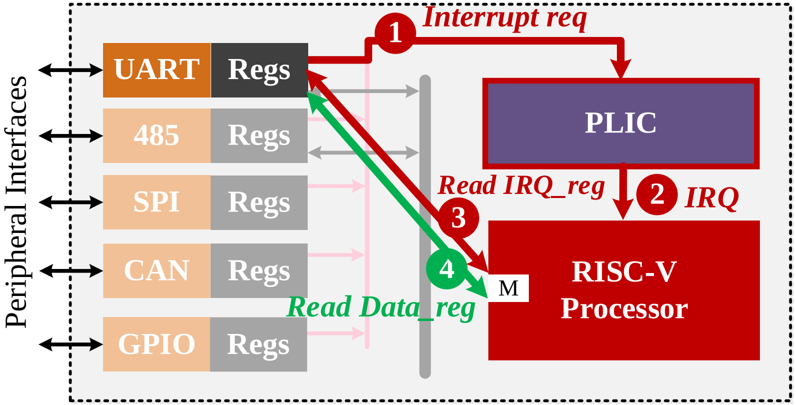
■ 中断请求与处理流程

① 外设触发中断条件，向中断控制器PLIC发起中断请求，例如UART接收到数据

② PLIC选出优先级最高的中断请求，通告CPU

③ CPU通过根据中断外设标识，通过数据数据总线读取外设的中断类型寄存器，获取触发的中断类型

④ 根据中断类型执行相应的中断程序，例如读取已接收的数据。



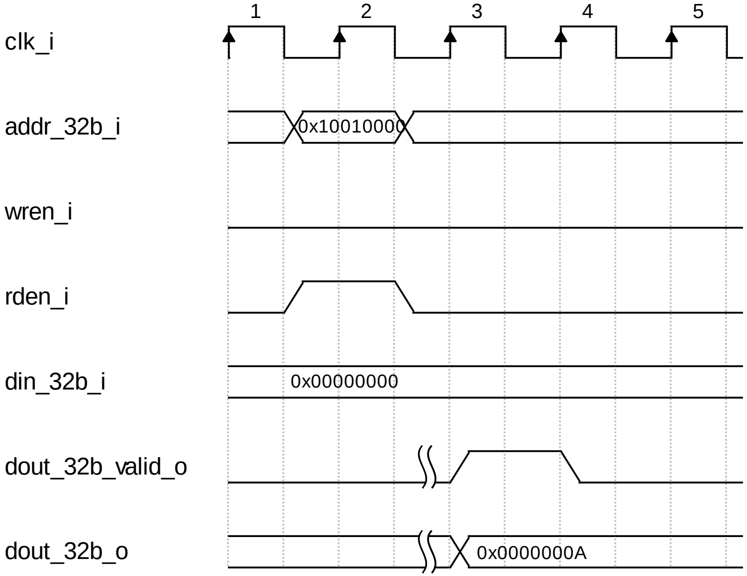
SoC外设数据接口（1/2）

- 数据接口定义，采用类Xilinx SRAM访问接口

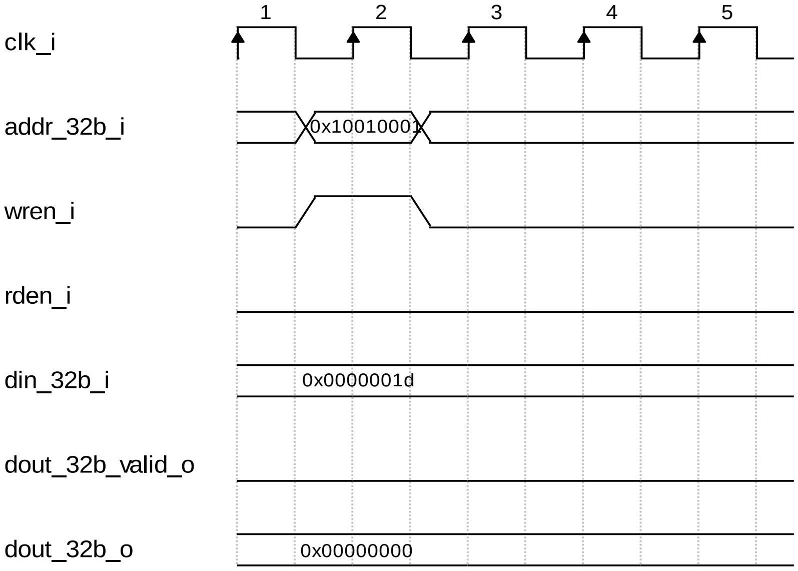
信号名称	宽度	方向	功能
clk_i	1	Input	访存时钟信号
rst_n_i	1	Input	复位所有寄存器，低使能
addr_32b_i	1	Input	访存地址，高16b用于标识外设信息，低16b用于标识外设内部寄存器信息
we_i	1	Input	写使能，高有效
en_i	1	Input	读使能，高有效
din_32b_i	1	Input	写数据
dout_valid_o	11	Output	读有效
dout_32b_o	32	Output	外设返回读取寄存器内容

SoC外设数据接口（2/2）

- 数据接口时序，与Xilinx SRAM接口类似



读时序



写时序

外设寄存器定义——UART

寄存器名称	地址 （偏移）	功能描述
UART_RBR	0x0	接收缓存寄存器（低8b有效），只读 如果接收缓存寄存器为空，则返0xFFFF_FFFF
UART_THR	0x4	发送缓存寄存器（低8b有效），只写
UART_DLM & UART_DLL	0x8	波特率除数（高&低 8b），具体数值与工作频率相关，例如50MHz工作频率，波特率为115200，则需要配置的波特率除数为16’d434
UART_IER	0x10	中断使能寄存器，支持读/写 [31:3]：保留 [2]： ‘1’表示打开校验错误触发中断功能 [1]： ‘1’表示打开TX-FIFO为空触发中断功能 [0]： ‘1’表示打开RX-FIFO超过阈值触发中断功能
UART_IIR	0x14	中断类型寄存器，只读 [31:4]：保留 [3:0]： ‘1’表示RX-FIFO超过阈值中断类型 [3:0]： ‘2’表示TX-FIFO为空中断类型 [3:0]： ‘3’表示校验错误中断类型

寄存器名称	地址 （偏移）	功能描述
UART_FCR	0x18	FIFO控制寄存器，只写 [31:2]：保留 [1:0]： RX-FIFO阈值，’0’表示阈值为1B， ’1’表示阈值为2B，’2’表示阈值为4B， ’3’表示阈值为8B，
UART_LCR	0x1c	线路控制寄存器，支持读/写 [31:8]：保留 [7]： 将该域置1时，表示允许配置波特率除数 [5:4]： 校验方式，0表示奇校验，1表示偶校验，2表示校验0，3表示校验1 [3]： 校验位是否打开标志，1表示打开 [2]： 尾位宽，0表示1b，1表示2b [1:0]：UART每次传输的位数，’0’表示阈值为5b，’1’表示阈值为6b，’2’表示阈值为7b，’3’表示阈值为8b，默认为8b
UART_LSR	0x20	线路状态寄存器，只读 [31:3]：保留 [2]： TX-FIFO是否为空，1为空 [1]： 是否遇到校验码错误，1为遇到 [0]： RX-FIFO是否为空，1为空

外设寄存器定义——CAN（BasicCAN）

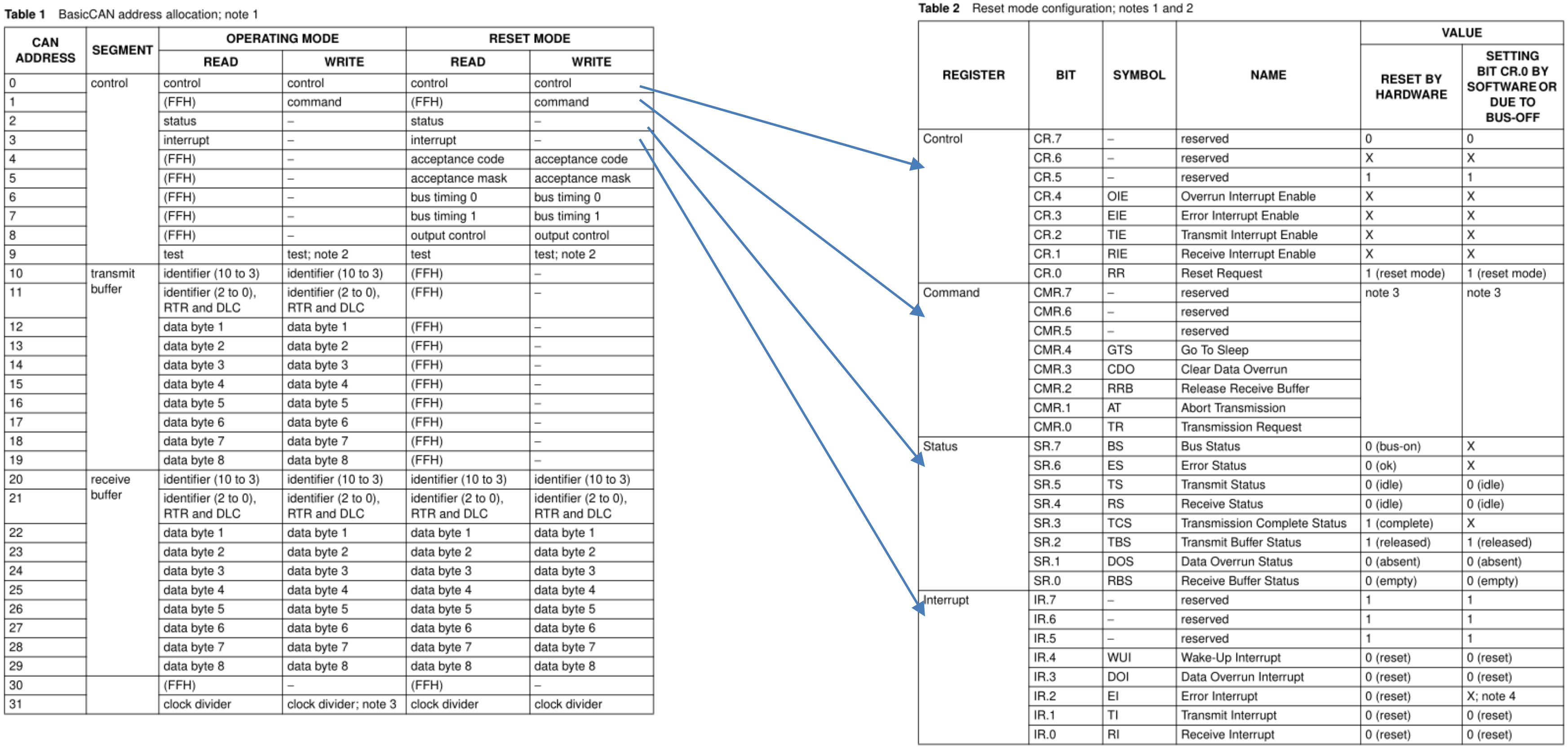


Table 1 BasicCAN address allocation; note 1

CAN ADDRESS	SEGMENT	OPERATING MODE		RESET MODE	
		READ	WRITE	READ	WRITE
0	control	control	control	control	control
1		(FFH)	command	(FFH)	command
2		status	–	status	–
3		interrupt	–	interrupt	–
4		(FFH)	–	acceptance code	acceptance code
5		(FFH)	–	acceptance mask	acceptance mask
6		(FFH)	–	bus timing 0	bus timing 0
7		(FFH)	–	bus timing 1	bus timing 1
8		(FFH)	–	output control	output control
9		test	test; note 2	test	test; note 2
10	transmit buffer	identifier (10 to 3)	identifier (10 to 3)	(FFH)	–
11		identifier (2 to 0), RTR and DLC	identifier (2 to 0), RTR and DLC	(FFH)	–
12		data byte 1	data byte 1	(FFH)	–
13		data byte 2	data byte 2	(FFH)	–
14		data byte 3	data byte 3	(FFH)	–
15		data byte 4	data byte 4	(FFH)	–
16		data byte 5	data byte 5	(FFH)	–
17		data byte 6	data byte 6	(FFH)	–
18		data byte 7	data byte 7	(FFH)	–
19		data byte 8	data byte 8	(FFH)	–
20	receive buffer	identifier (10 to 3)	identifier (10 to 3)	identifier (10 to 3)	identifier (10 to 3)
21		identifier (2 to 0), RTR and DLC	identifier (2 to 0), RTR and DLC	identifier (2 to 0), RTR and DLC	identifier (2 to 0), RTR and DLC
22		data byte 1	data byte 1	data byte 1	data byte 1
23		data byte 2	data byte 2	data byte 2	data byte 2
24		data byte 3	data byte 3	data byte 3	data byte 3
25		data byte 4	data byte 4	data byte 4	data byte 4
26		data byte 5	data byte 5	data byte 5	data byte 5
27		data byte 6	data byte 6	data byte 6	data byte 6
28		data byte 7	data byte 7	data byte 7	data byte 7
29		data byte 8	data byte 8	data byte 8	data byte 8
30		(FFH)	–	(FFH)	–
31		clock divider	clock divider; note 3	clock divider	clock divider

Table 2 Reset mode configuration; notes 1 and 2

REGISTER	BIT	SYMBOL	NAME	VALUE	
				RESET BY HARDWARE	SETTING BIT CR.0 BY SOFTWARE OR DUE TO BUS-OFF
Control	CR.7	–	reserved	0	0
	CR.6	–	reserved	X	X
	CR.5	–	reserved	1	1
	CR.4	OIE	Overrun Interrupt Enable	X	X
	CR.3	EIE	Error Interrupt Enable	X	X
	CR.2	TIE	Transmit Interrupt Enable	X	X
	CR.1	RIE	Receive Interrupt Enable	X	X
	CR.0	RR	Reset Request	1 (reset mode)	1 (reset mode)
Command	CMR.7	–	reserved	note 3	note 3
	CMR.6	–	reserved		
	CMR.5	–	reserved		
	CMR.4	GTS	Go To Sleep		
	CMR.3	CDO	Clear Data Overrun		
	CMR.2	RRB	Release Receive Buffer		
	CMR.1	AT	Abort Transmission		
	CMR.0	TR	Transmission Request		
Status	SR.7	BS	Bus Status	0 (bus-on)	X
	SR.6	ES	Error Status	0 (ok)	X
	SR.5	TS	Transmit Status	0 (idle)	0 (idle)
	SR.4	RS	Receive Status	0 (idle)	0 (idle)
	SR.3	TCS	Transmission Complete Status	1 (complete)	X
	SR.2	TBS	Transmit Buffer Status	1 (released)	1 (released)
	SR.1	DOS	Data Overrun Status	0 (absent)	0 (absent)
	SR.0	RBS	Receive Buffer Status	0 (empty)	0 (empty)
Interrupt	IR.7	–	reserved	1	1
	IR.6	–	reserved	1	1
	IR.5	–	reserved	1	1
	IR.4	WUI	Wake-Up Interrupt	0 (reset)	0 (reset)
	IR.3	DOI	Data Overrun Interrupt	0 (reset)	0 (reset)
	IR.2	EI	Error Interrupt	0 (reset)	X; note 4
	IR.1	TI	Transmit Interrupt	0 (reset)	0 (reset)
	IR.0	RI	Receive Interrupt	0 (reset)	0 (reset)

外设寄存器定义——CAN（PeliCAN）

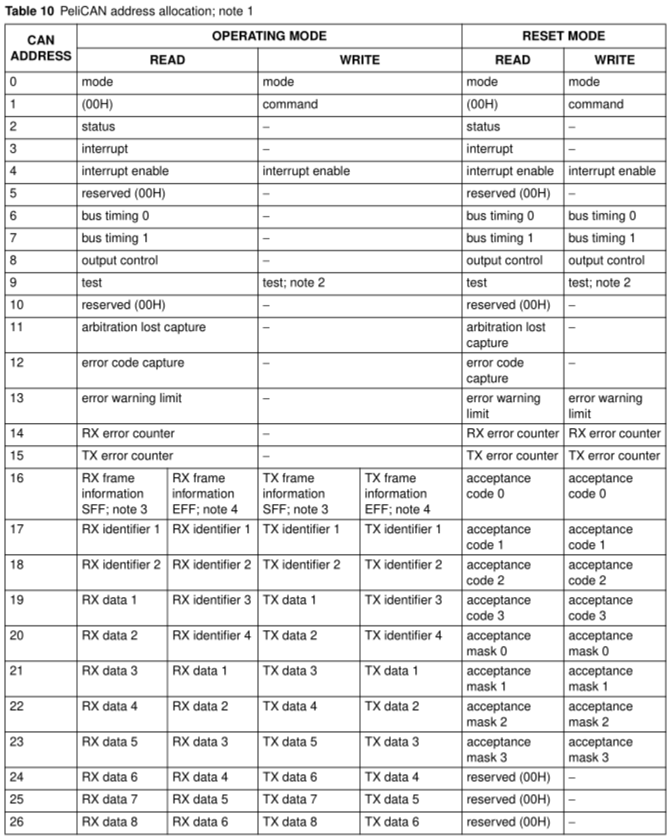
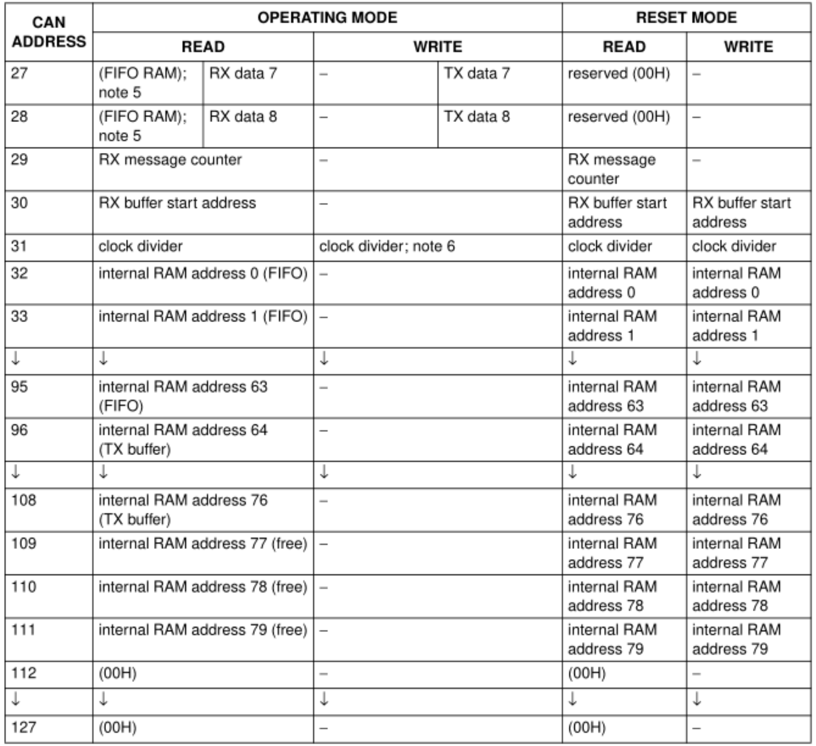


Table 10 PeliCAN address allocation; note 1

CAN ADDRESS	OPERATING MODE				RESET MODE	
	READ		WRITE		READ	WRITE
0	mode		mode		mode	mode
1	(00H)		command		(00H)	command
2	status		–		status	–
3	interrupt		–		interrupt	–
4	interrupt enable		interrupt enable		interrupt enable	interrupt enable
5	reserved (00H)		–		reserved (00H)	–
6	bus timing 0		–		bus timing 0	bus timing 0
7	bus timing 1		–		bus timing 1	bus timing 1
8	output control		–		output control	output control
9	test		test; note 2		test	test; note 2
10	reserved (00H)		–		reserved (00H)	–
11	arbitration lost capture		–		arbitration lost capture	–
12	error code capture		–		error code capture	–
13	error warning limit		–		error warning limit	error warning limit
14	RX error counter		–		RX error counter	RX error counter
15	TX error counter		–		TX error counter	TX error counter
16	RX frame information SFF; note 3	RX frame information EFF; note 4	TX frame information SFF; note 3	TX frame information EFF; note 4	acceptance code 0	acceptance code 0
17	RX identifier 1	RX identifier 1	TX identifier 1	TX identifier 1	acceptance code 1	acceptance code 1
18	RX identifier 2	RX identifier 2	TX identifier 2	TX identifier 2	acceptance code 2	acceptance code 2
19	RX data 1	RX identifier 3	TX data 1	TX identifier 3	acceptance code 3	acceptance code 3
20	RX data 2	RX identifier 4	TX data 2	TX identifier 4	acceptance mask 0	acceptance mask 0
21	RX data 3	RX data 1	TX data 3	TX data 1	acceptance mask 1	acceptance mask 1
22	RX data 4	RX data 2	TX data 4	TX data 2	acceptance mask 2	acceptance mask 2
23	RX data 5	RX data 3	TX data 5	TX data 3	acceptance mask 3	acceptance mask 3
24	RX data 6	RX data 4	TX data 6	TX data 4	reserved (00H)	–
25	RX data 7	RX data 5	TX data 7	TX data 5	reserved (00H)	–
26	RX data 8	RX data 6	TX data 8	TX data 6	reserved (00H)	–



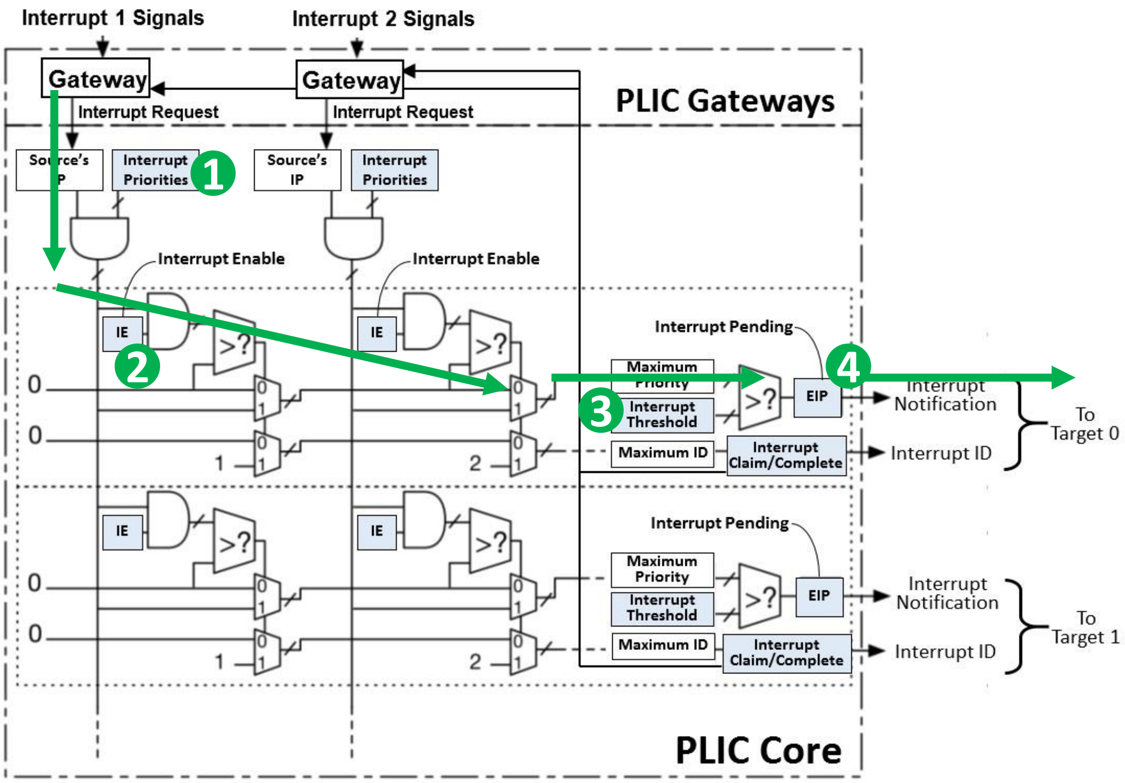
CAN ADDRESS	OPERATING MODE				RESET MODE	
	READ		WRITE		READ	WRITE
27	(FIFO RAM); note 5	RX data 7	–	TX data 7	reserved (00H)	–
28	(FIFO RAM); note 5	RX data 8	–	TX data 8	reserved (00H)	–
29	RX message counter		–		RX message counter	–
30	RX buffer start address		–		RX buffer start address	RX buffer start address
31	clock divider		clock divider; note 6		clock divider	clock divider
32	internal RAM address 0 (FIFO)		–		internal RAM address 0	internal RAM address 0
33	internal RAM address 1 (FIFO)		–		internal RAM address 1	internal RAM address 1
↓	↓		↓		↓	↓
95	internal RAM address 63 (FIFO)		–		internal RAM address 63	internal RAM address 63
96	internal RAM address 64 (TX buffer)		–		internal RAM address 64	internal RAM address 64
↓	↓		↓		↓	↓
108	internal RAM address 76 (TX buffer)		–		internal RAM address 76	internal RAM address 76
109	internal RAM address 77 (free)		–		internal RAM address 77	internal RAM address 77
110	internal RAM address 78 (free)		–		internal RAM address 78	internal RAM address 78
111	internal RAM address 79 (free)		–		internal RAM address 79	internal RAM address 79
112	(00H)		–		(00H)	–
↓	↓		↓		↓	↓
127	(00H)		–		(00H)	–

主要内容

- SoC接口设计
  - SoC外设接口组成
  - SoC外设接口定义
  - SoC外设寄存器定义
- 外设中断设计
  - RISC-V中断规范

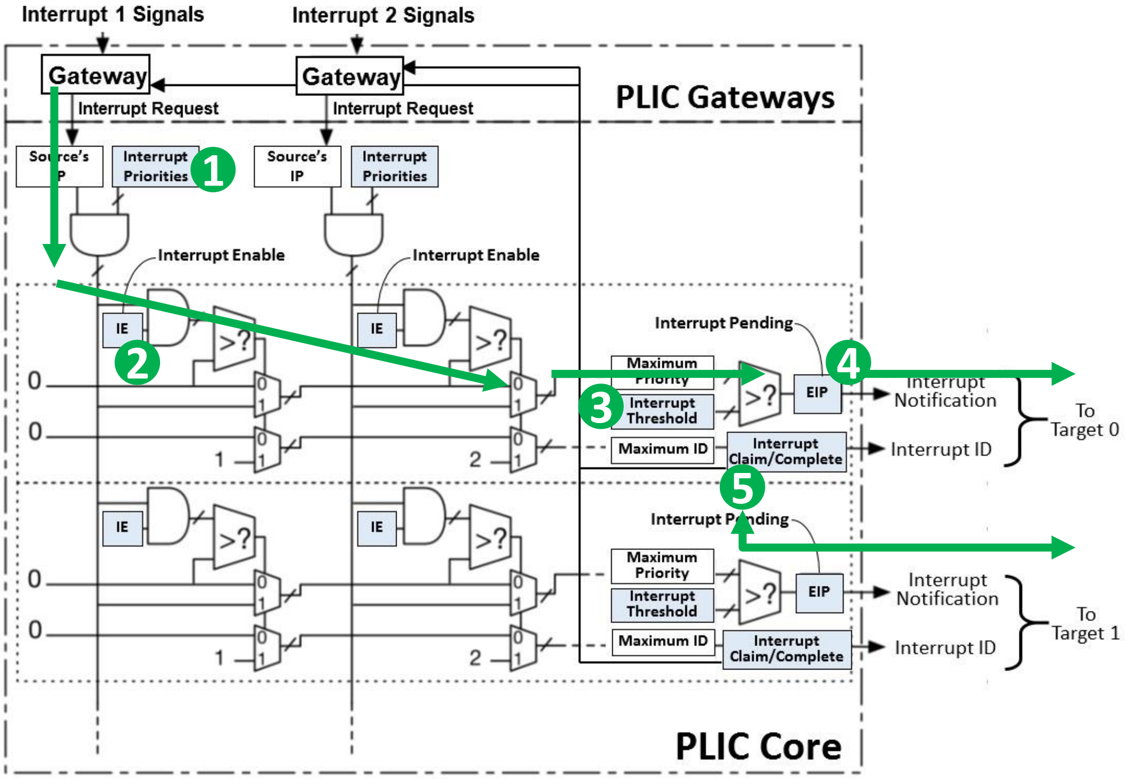
RISC-V中断规范

- 6个寄存器
  - ① Interrupt priority register（中断优先级寄存器）
  - ② enables register（中断使能寄存器）
  - ③ Priority thresholds register（优先级阈值寄存器）
  - ④ Interrupt pending bits register（中断挂起寄存器）
  - ⑤ Interrupt claim register（中断声明寄存器）
  - ⑥ Interrupt completion register（中断完成寄存器）



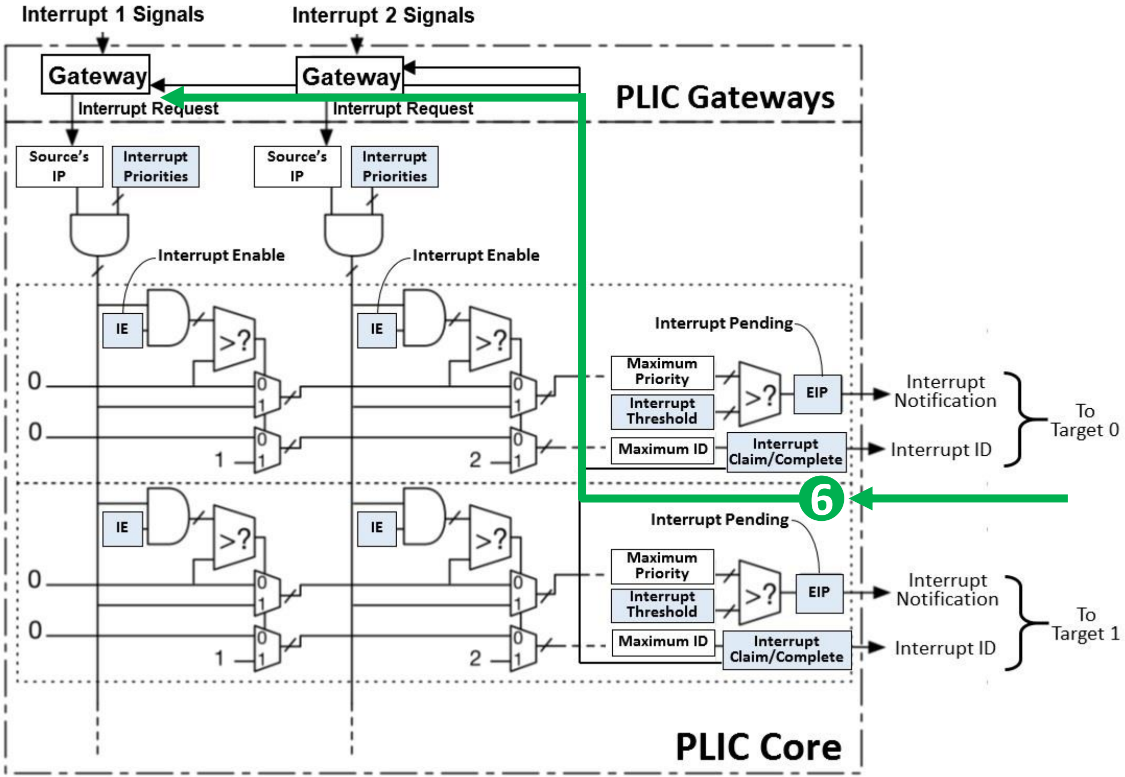
RISC-V中断规范

- 6个寄存器
  ① Interrupt priority register（中断优先级寄存器）
  ② enables register（中断使能寄存器）
  ③ Priority thresholds register（优先级阈值寄存器）
  ④ Interrupt pending bits register（中断挂起寄存器）
  ⑤ Interrupt claim register（中断声明寄存器）
  ⑥ Interrupt completion register（中断完成寄存器）



RISC-V中断规范

- 6个寄存器
  ① Interrupt priority register（中断优先级寄存器）
  ② enables register（中断使能寄存器）
  ③ Priority thresholds register（优先级阈值寄存器）
  ④ Interrupt pending bits register（中断挂起寄存器）
  ⑤ Interrupt claim register（中断声明寄存器）
  ⑥ Interrupt completion register（中断完成寄存器）



谢谢！