General Purpose Input/Output (GPIO)

Zonghua Gu

Fall 2025

Acknowledgement: some slides taken from instructor resources of Daniel Lewis textbook

Raspberry Pi GPIO pins

Raspberry Pi (Rev1)

3V3	1	2	5V
GPIOØ	3	4	5V
GPIO1	5	6	GND
GPIO4	7	8	GPIO14
GND	9	10	GPIO15
GPIO17	11	12	GPIO18
GPIO27	13	14	GND
GPIO22	15	16	GPIO23
3V3	17	18	GPIO24
GPIO10	19	20	GND
GPIO9	21	22	GPIO25
GPIO11	23	24	GPIO8
GND	25	26	GPIO7

Raspberry Pi (Rev 2)

3V3	1	2	5V
GPIO2	3	4	5V
GPIO3	5	6	GND
GPIO4	7	8	GPIO14
GND	9	10	GPIO15
GPIO17	11	12	GPIO18
GPIO27	13	14	GND
GPIO22	15	16	GPIO23
3V3	17	18	GPIO24
GPIO10	19	20	GND
GPIO9	21	22	GPIO25
GPIO11	23	24	GPIO8
GND	25	26	GPIO7

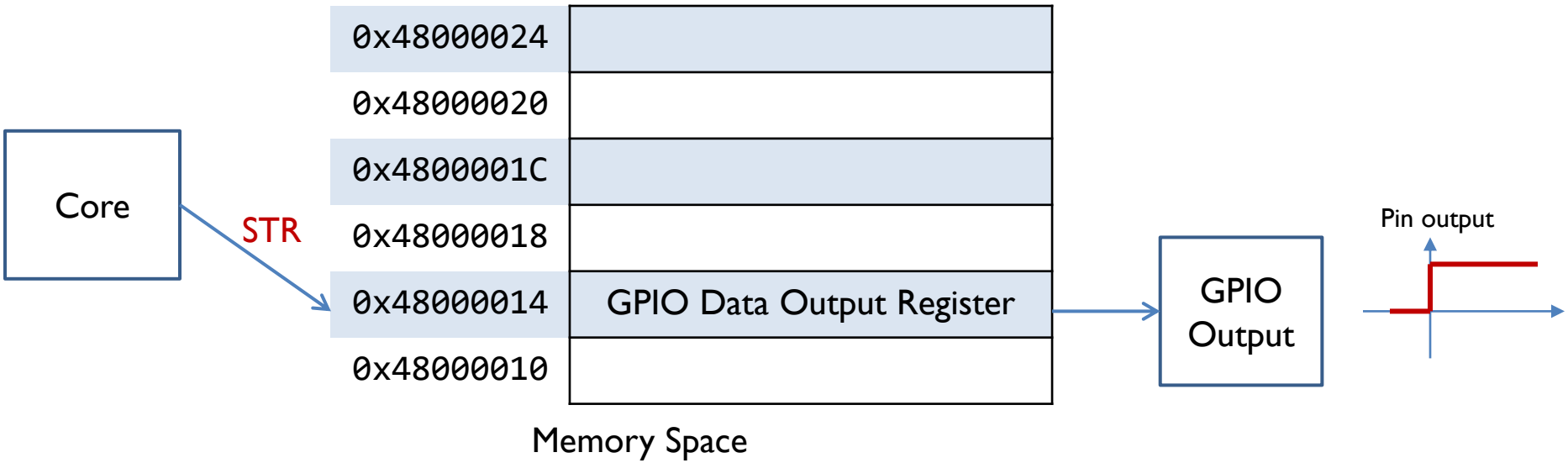
Raspberry Pi B+, 2, 3 & Zero

3V3	1	2	5V
GPIO2	3	4	5V
GPIO3	5	6	GND
GPIO4	7	8	GPIO14
GND	9	10	GPIO15
GPIO17	11	12	GPIO18
GPIO27	13	14	GND
GPIO22	15	16	GPIO23
3V3	17	18	GPIO24
GPIO10	19	20	GND
GPIO9	21	22	GPIO25
GPIO11	23	24	GPIO8
GND	25	26	GPIO7
DNC	27	28	DNC
GPIO5	29	30	GND
GPIO6	31	32	GPIO12
GPIO13	33	34	GND
GPIO19	35	36	GPIO16
GPIO26	37	38	GPIO20
GND	39	40	GPIO21

Key
+
Ground
UART
I2C
SPI
GPIO
Pin Number

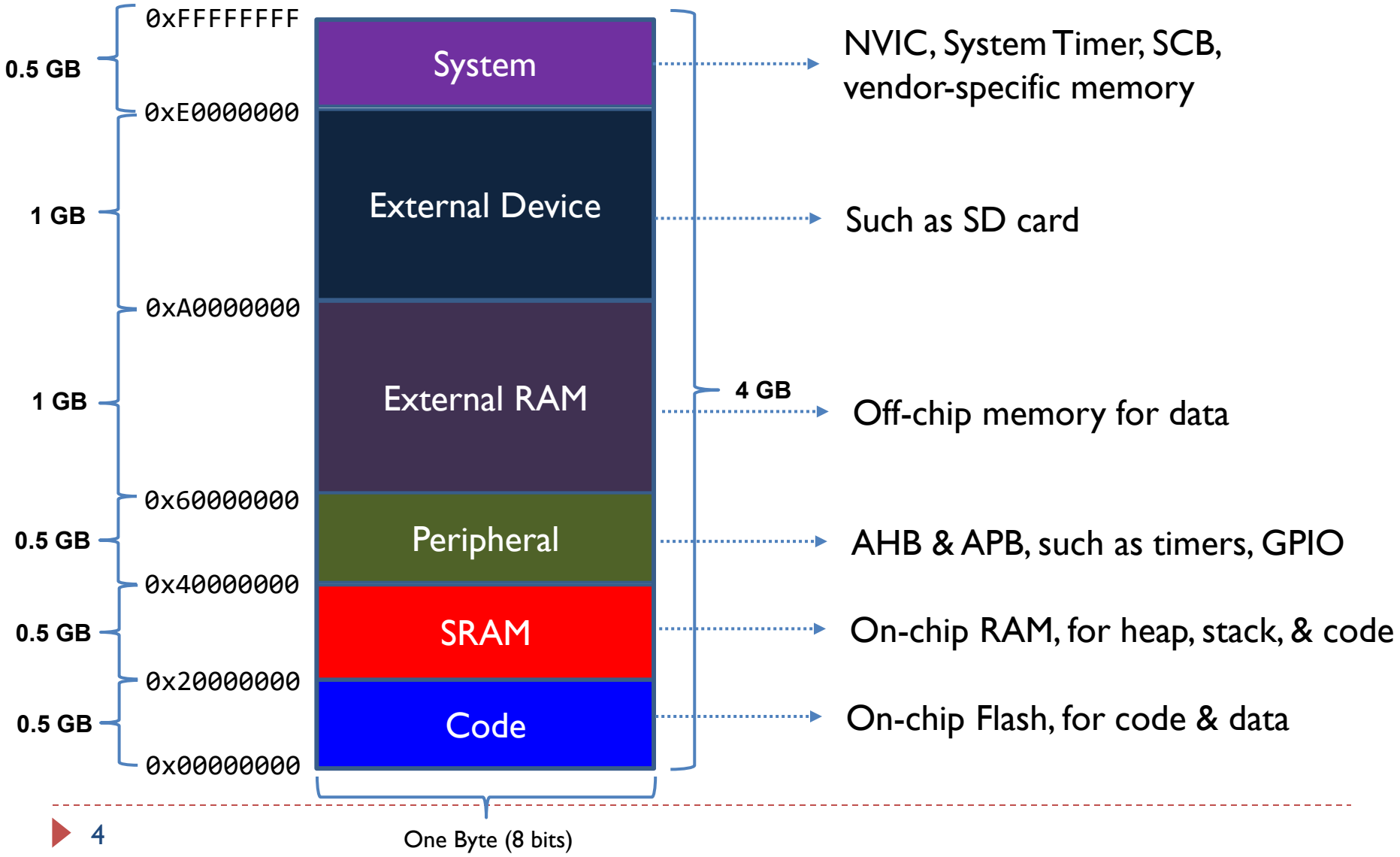
Interfacing Peripherals

- **Port-mapped I/O**
  - Use special CPU instructions: `Special_instruction Reg, Port`
- **Memory-mapped I/O**
  - A simpler and more convenient way to interface I/O devices
  - Each device registers is assigned to a memory address in the address space of the microprocessor
  - Use native CPU load/store instructions: `LDR/STR Reg, [Reg, #imm]`

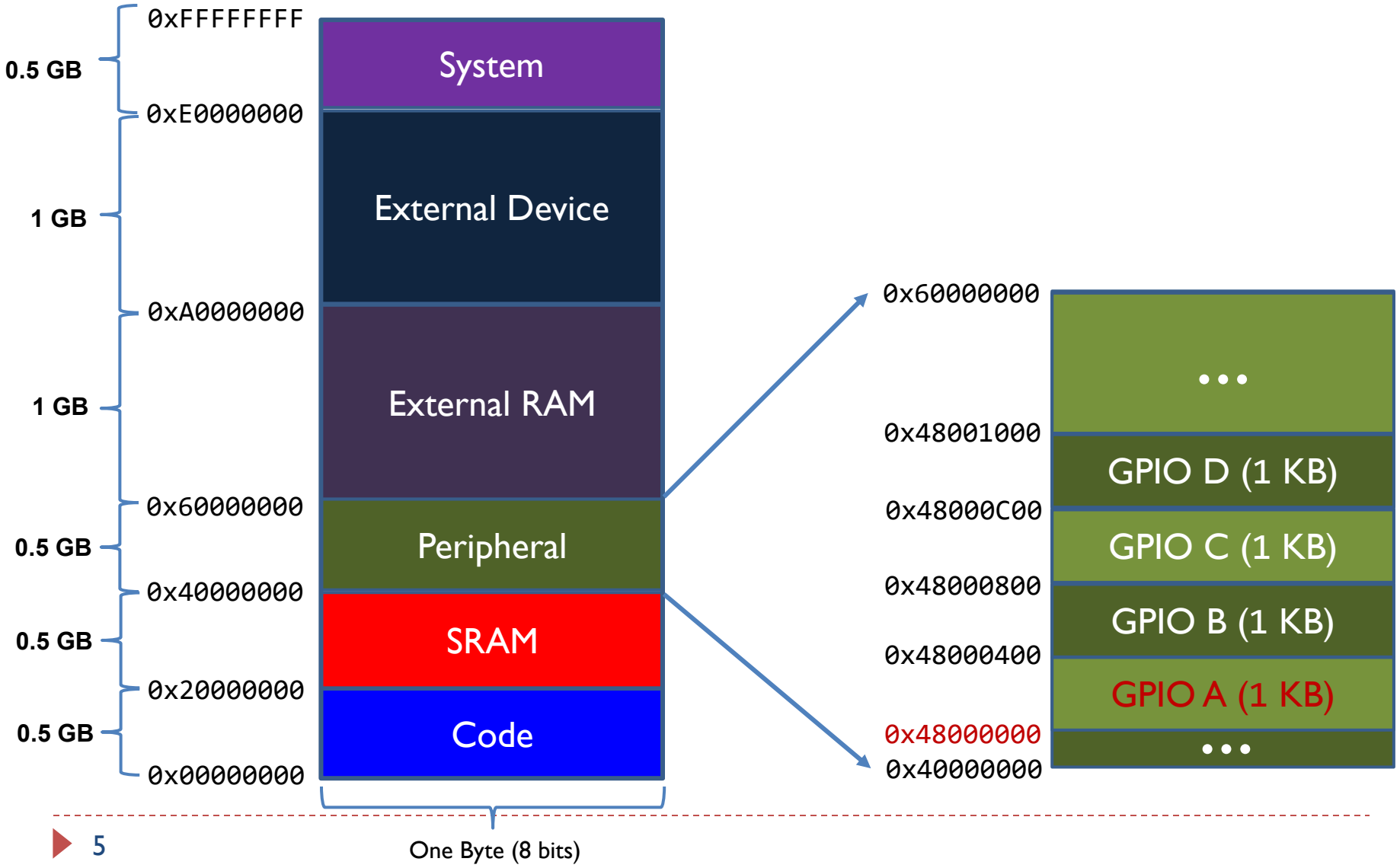


ARM Cortex-M microprocessors use memory-mapped I/O.

Memory Map of Cortex-M4 (Pi uses Cortex-A53)



Memory Map of STM32L4

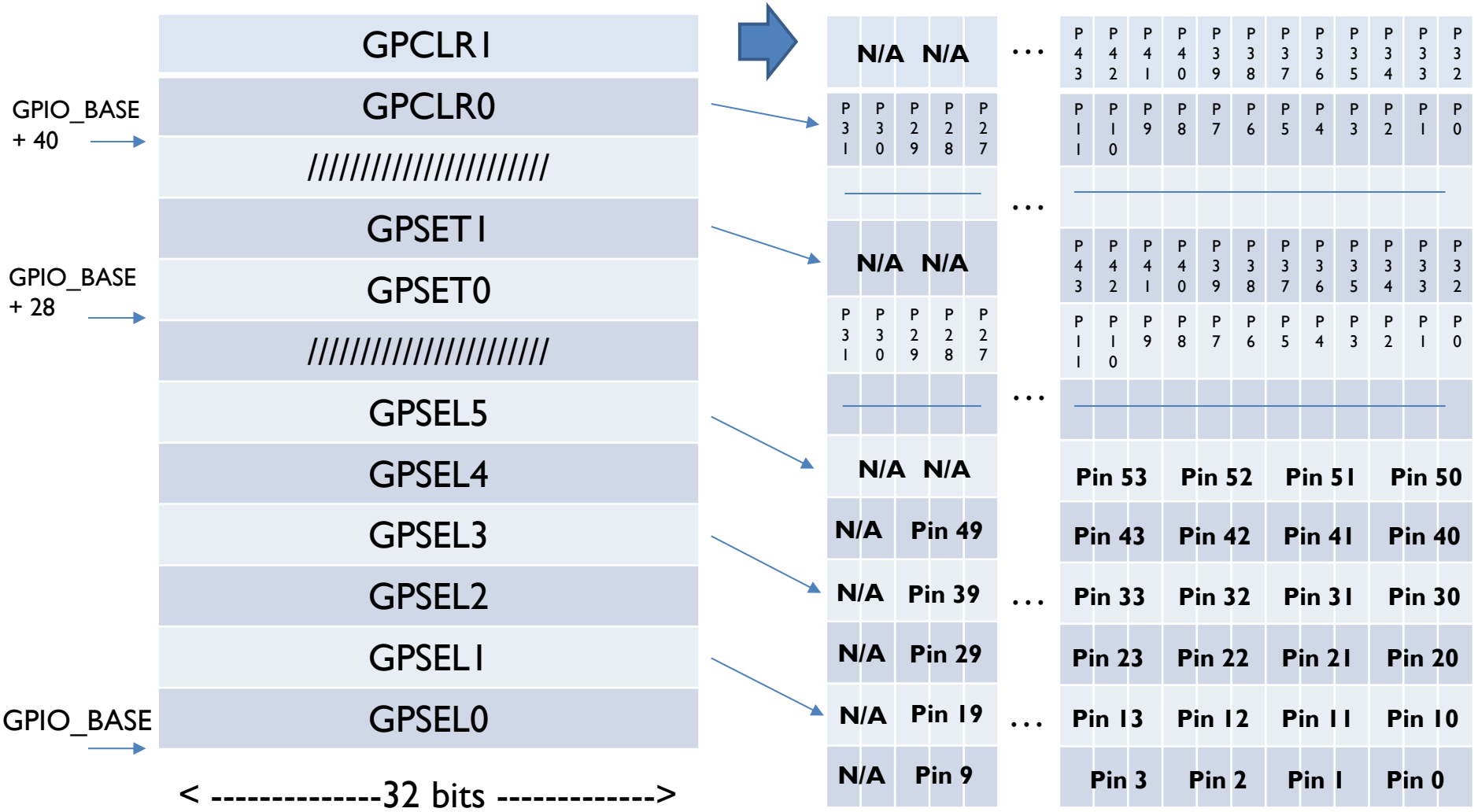


Steps to using GPIO pins

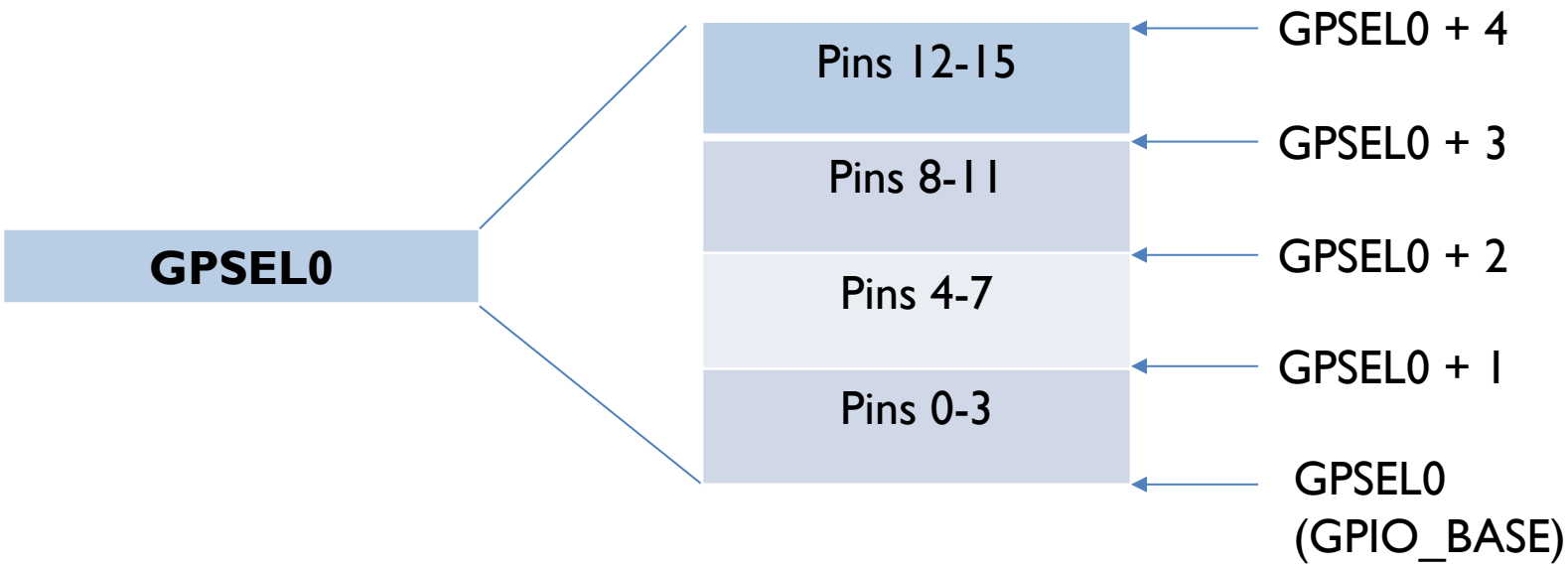
- 1) Configure the pin as an input or an output
  - A) Get the address of the appropriate GPSEL0 register
  - B) ORR the current value with the value you want

- 1) Set or clear the pin
  - A) Get the address of GPSET or GPCLR register
  - B) ORR the current value with the value you want

Memory-mapped GPIO Cortex-A53 (Pi)



Thinking about memory addresses

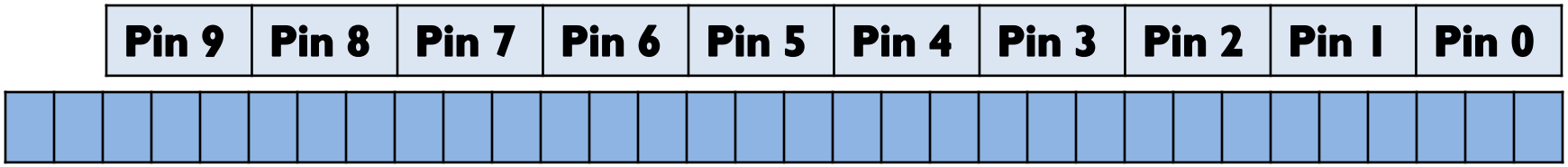


GPSEL Registers on ARM-Cortex A53 (Pi)

No	Name	Off	Pins
0	GPIO Function Select 0 (GPSEL0)	#0	0-9
1	GPIO Function Select 1 (GPSEL1)	#4	10-19
2	GPIO Function Select 2 (GPSEL2)	#8	20-29
3	GPIO Function Select 3 (GPSEL3)	#12	30-39
4	GPIO Function Select 4 (GPSEL4)	#16	40-49
5	GPIO Function Select 5 (GPSEL5)	#20	50-53

Figure 25b. GPIO registers and pin control.

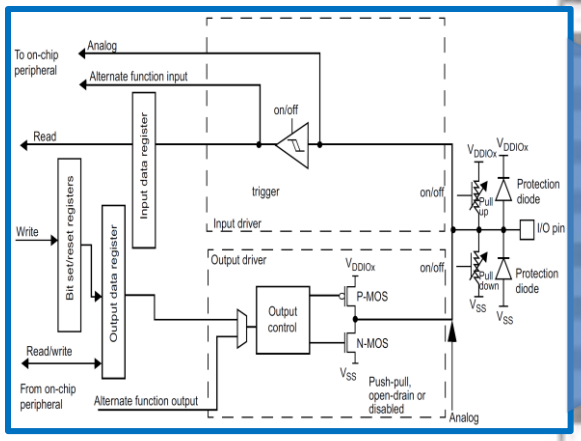
GPSEL0 Register

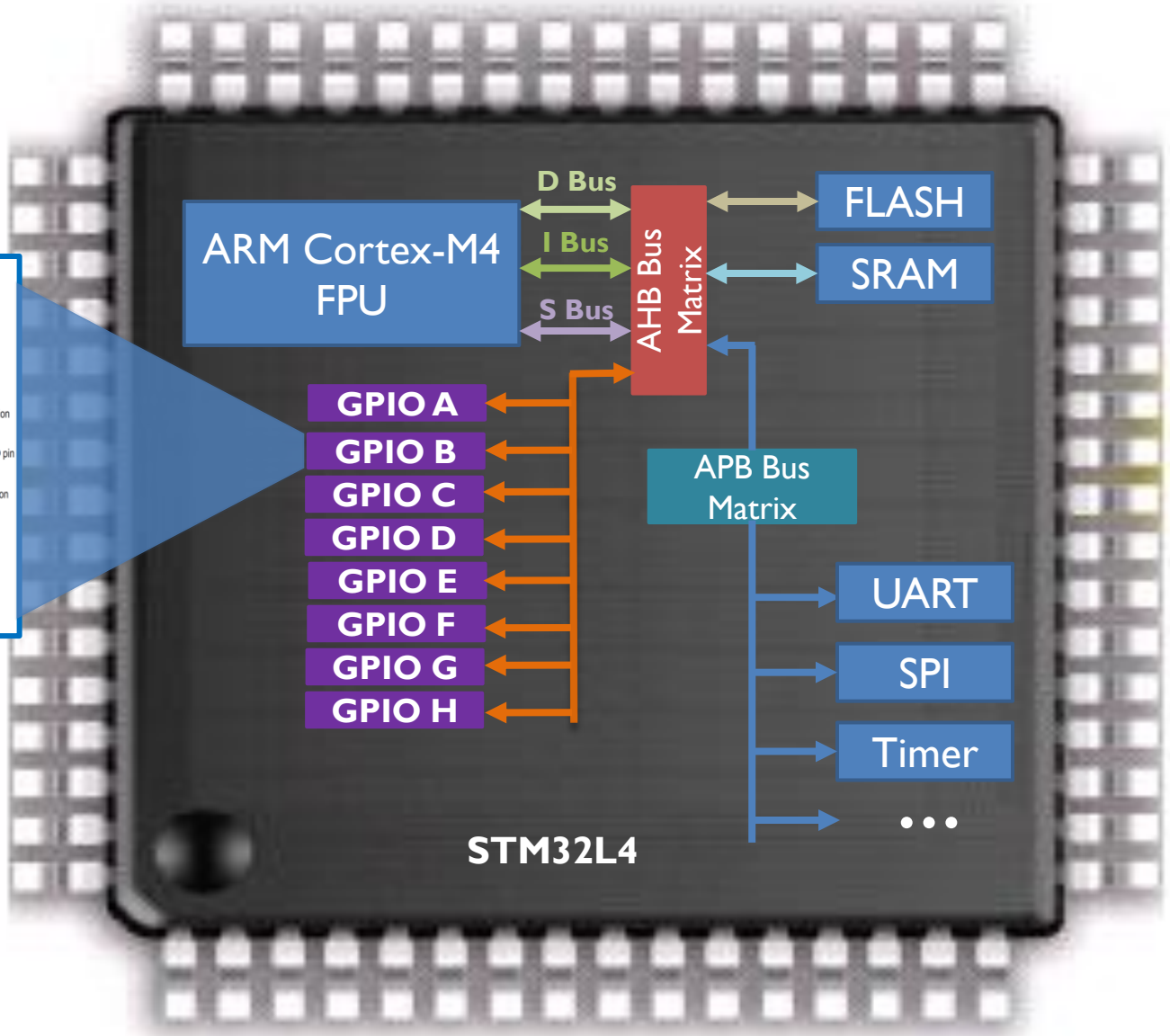


How would we set pin 2 as an output in assembly?

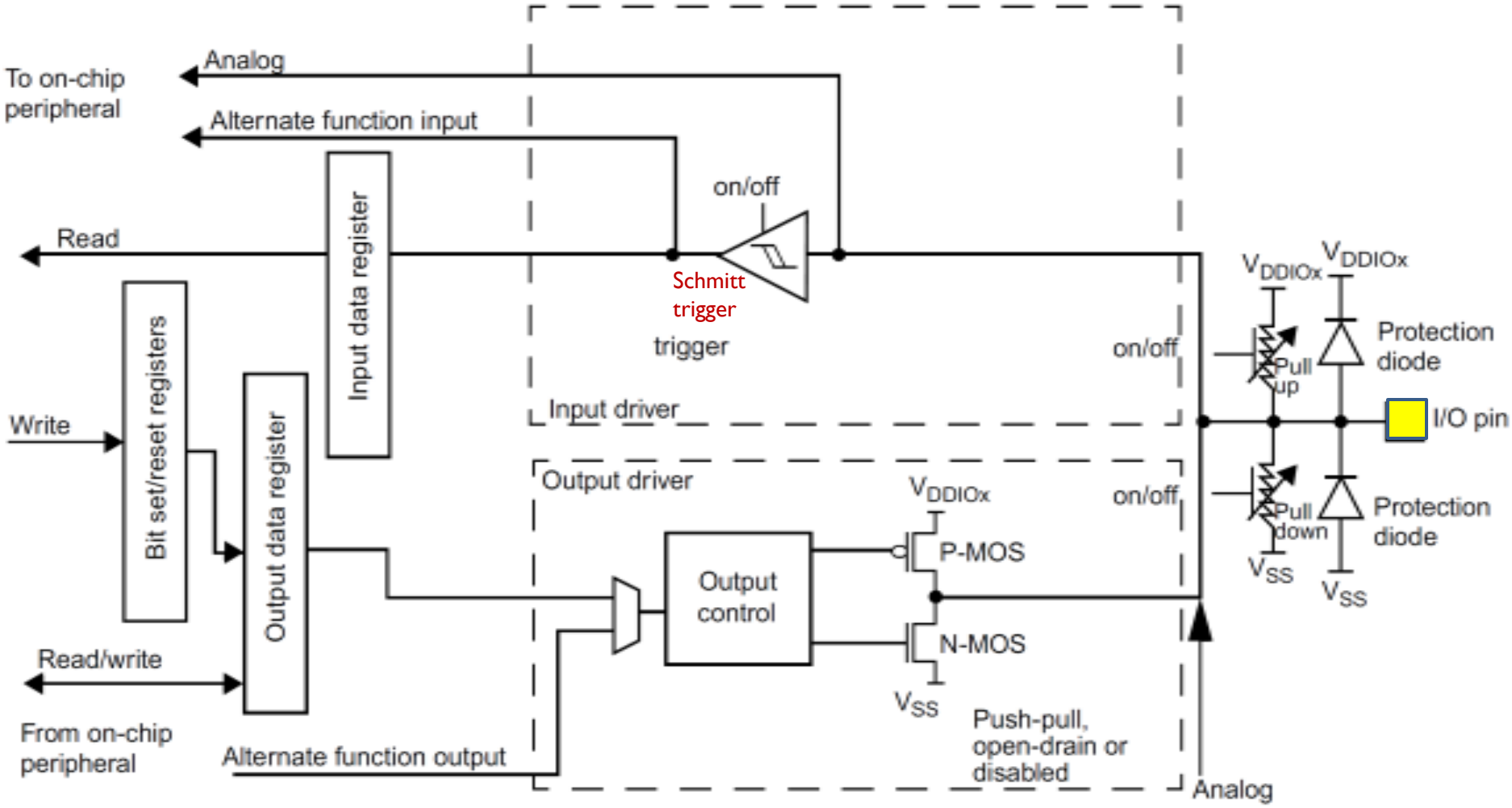
```
LDR R1, =GPSEL0
LDR R2, [R1]
MOV R3, #1<<6
ORR R2, R3
STR R2, [R1]
```

General Purpose Input/Output (GPIO)



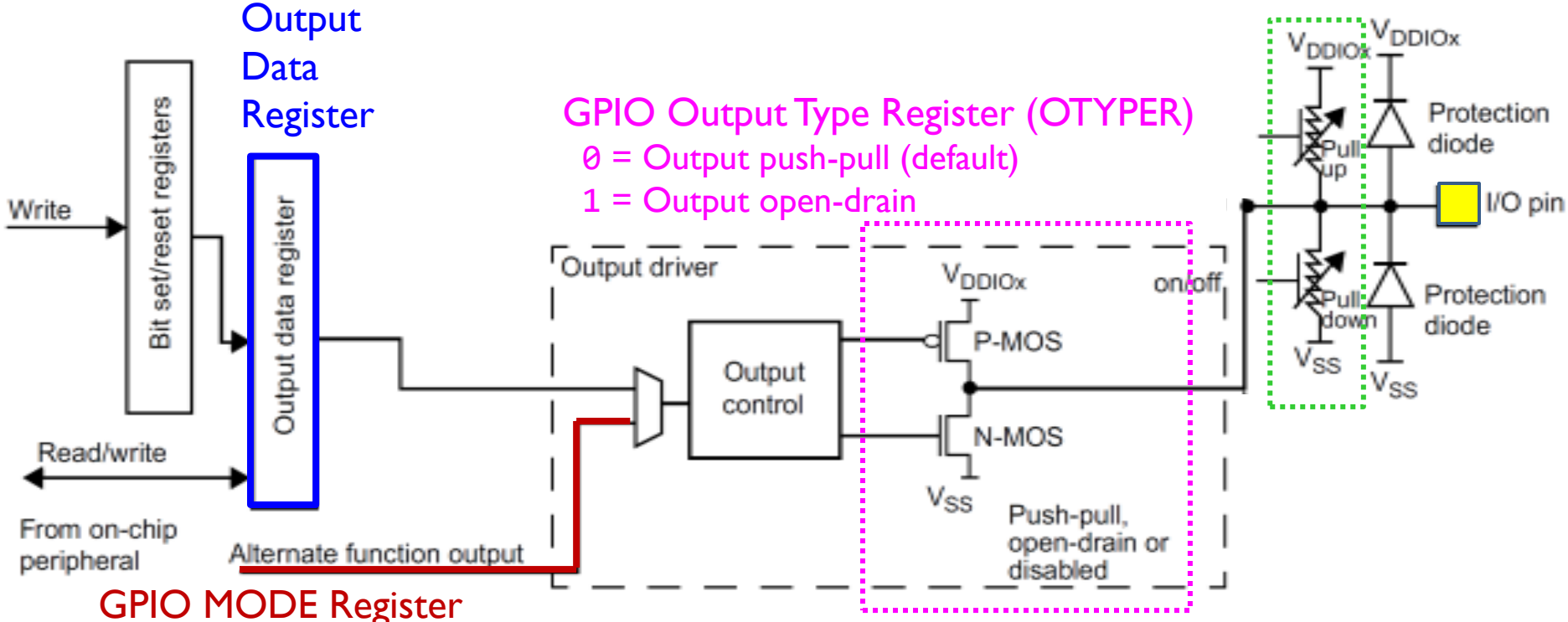


Basic Structure of an I/O Port Bit
Input and Output



Basic Structure of an I/O Port Bit: Output

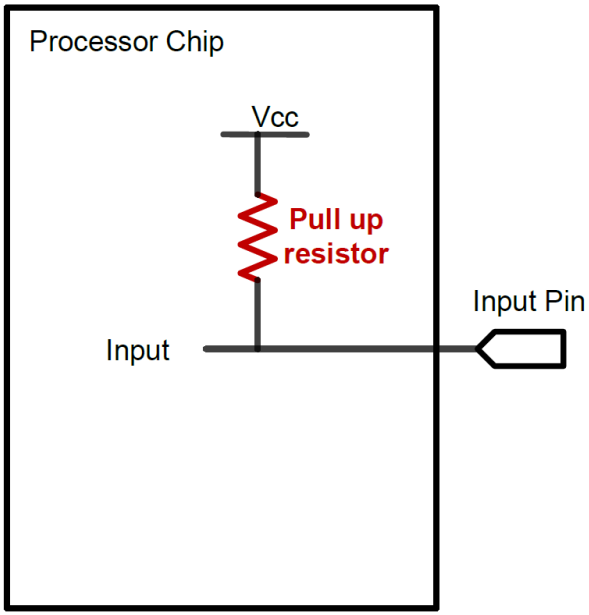
GPIO Pull-up/Pull-down Register (PUPDR)
00 = No pull-up, pull-down 01 = Pull-up
10 = Pull-down 11 = Reserved



GPIO MODE Register
00 = Input, 01 = Output,
10 = AF, 11 = Analog (default)

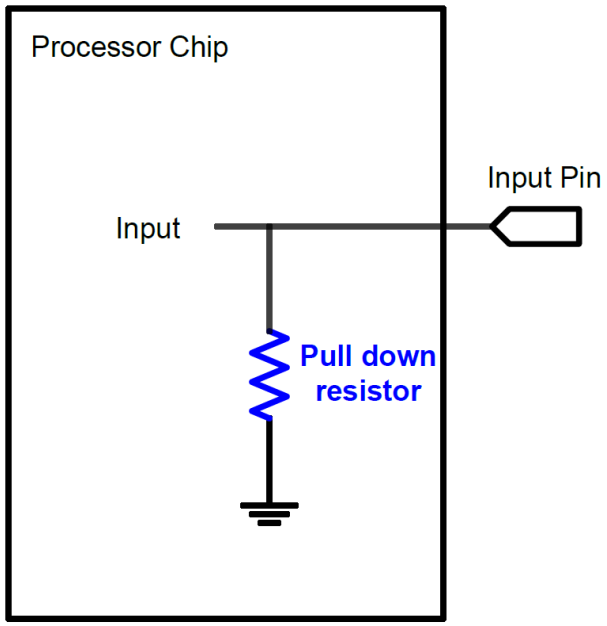
GPIO Input: Pull Up and Pull Down

- A digital input can have three states: High, Low, and High-Impedance (also called floating, tri-stated, HiZ)



Pull-Up

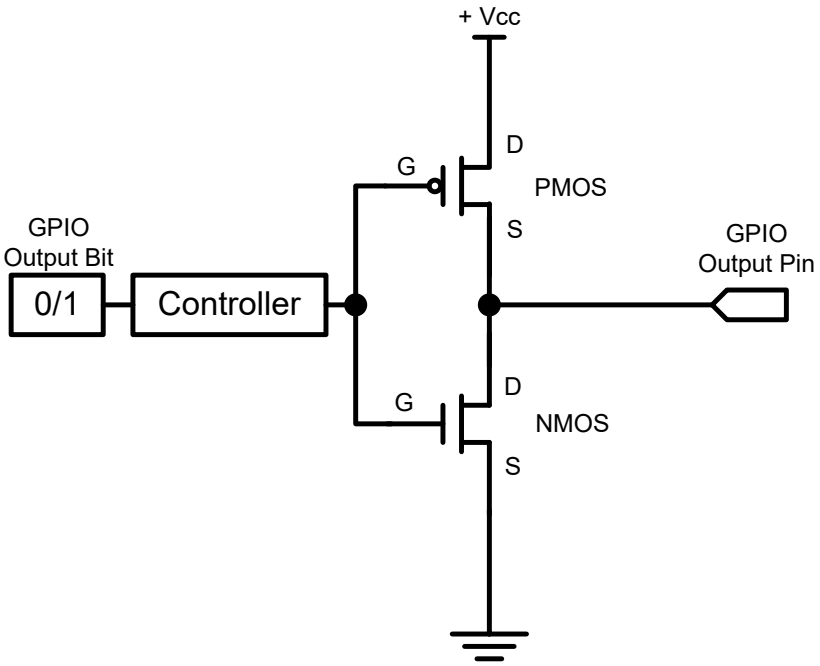
If external input is HiZ, the input is read as a valid HIGH.

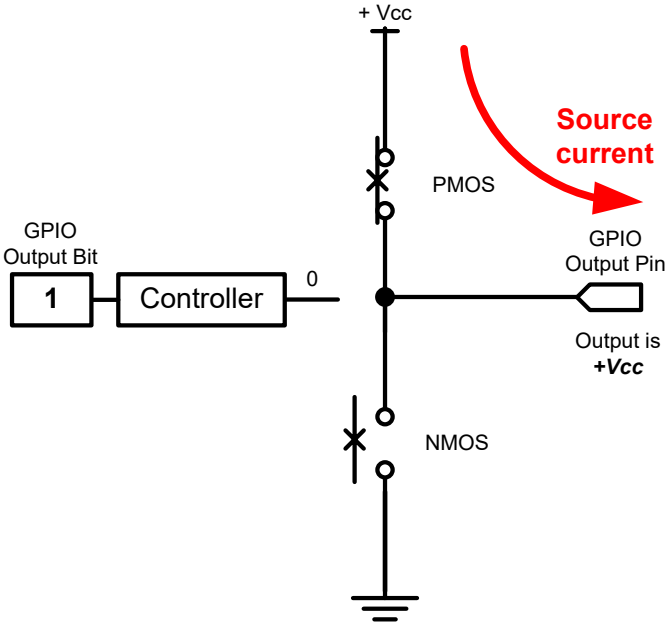


Pull-Down

If external input is HiZ, the input is read as a valid LOW.

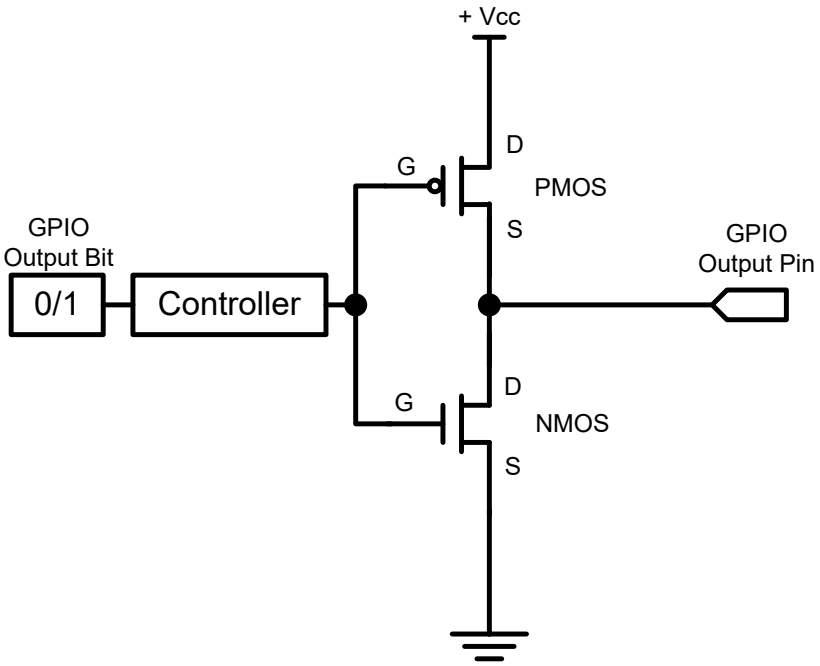
GPIO Output: Push-Pull

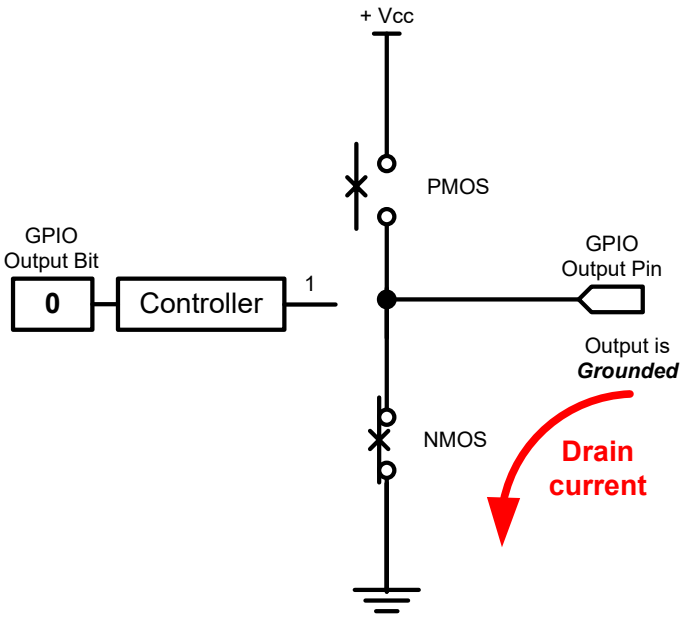




GPIO Output = 1
Source current to external circuit

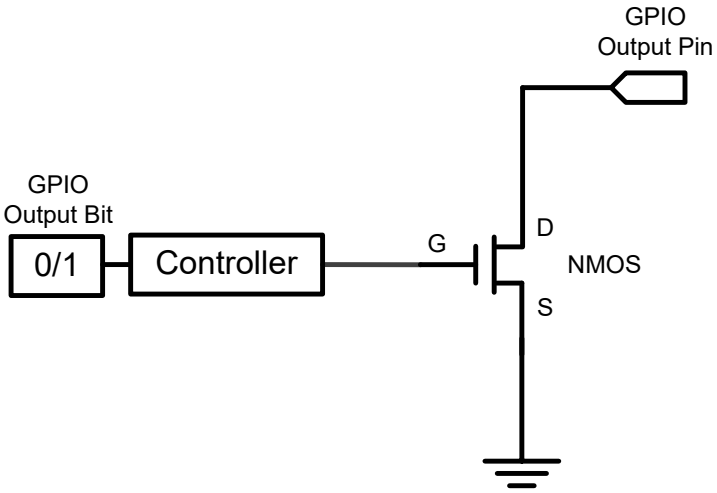
GPIO Output: Push-Pull

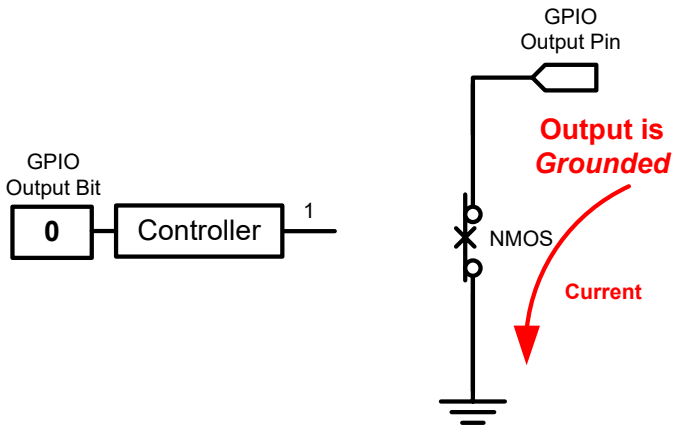




GPIO Output = 0
Drain current from external circuit

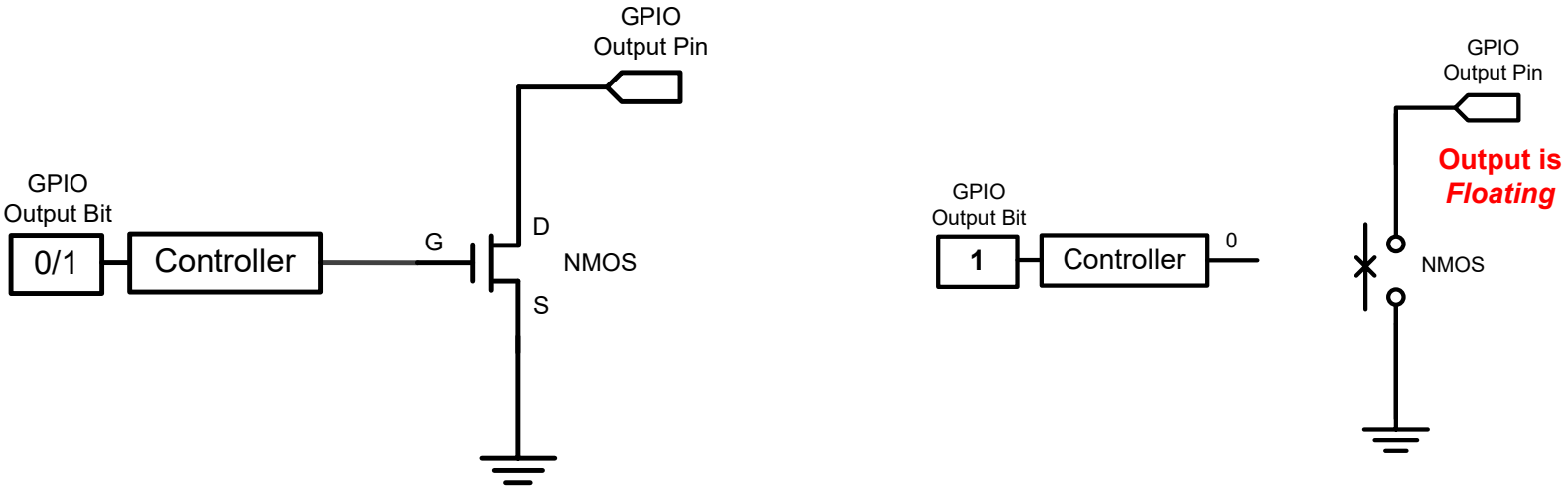
GPIO Output: Open-Drain





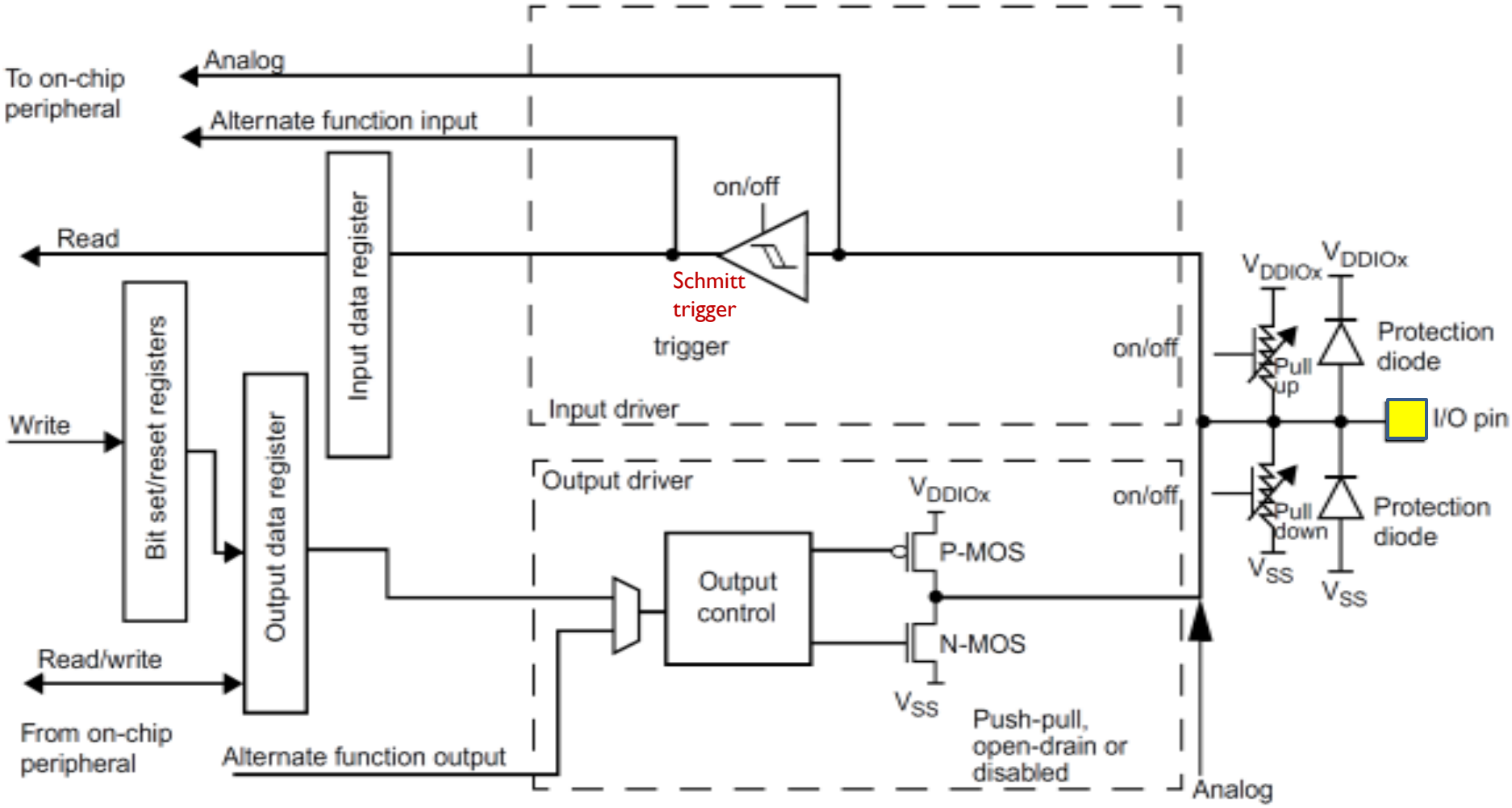
GPIO Output = 0
Drain current from external circuit

GPIO Output: Open-Drain

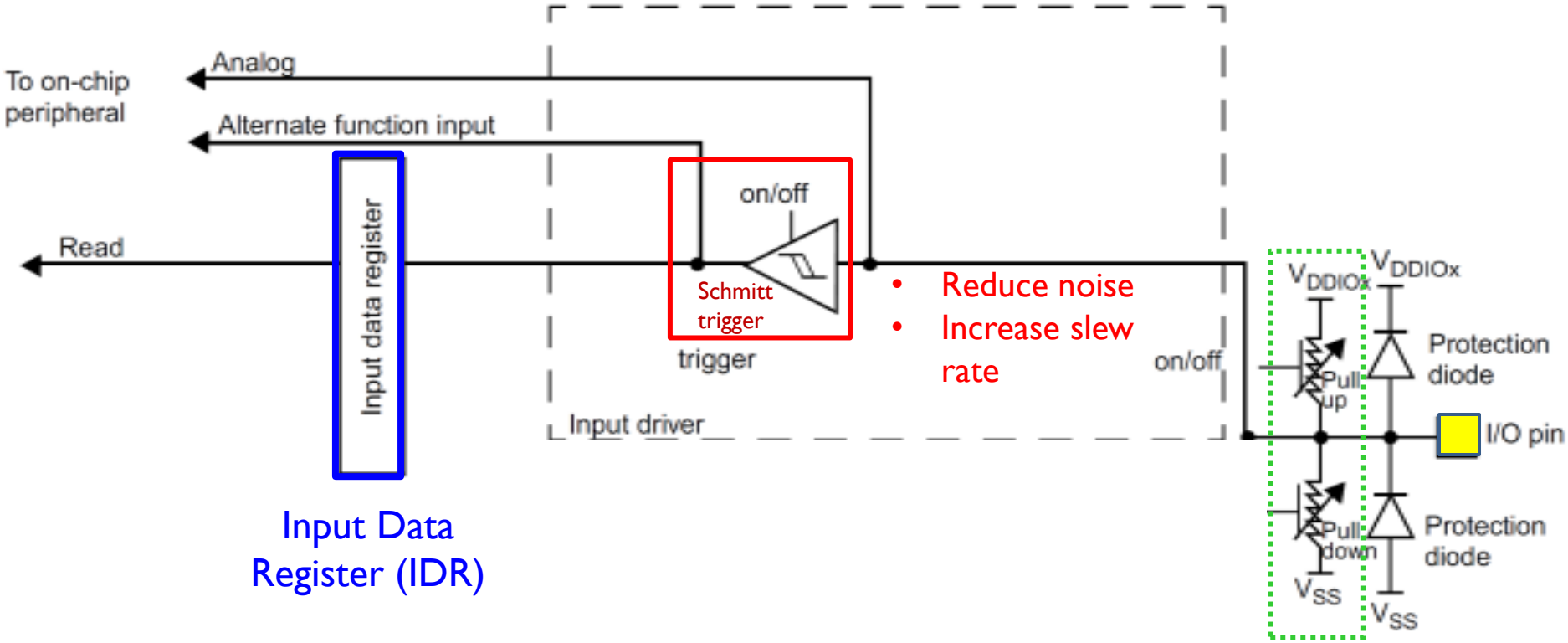


Output = 1
GPIO Pin has high-impedance to external circuit

Basic Structure of an I/O Port Bit
Input and Output



Basic Structure of an I/O Port Bit: Input



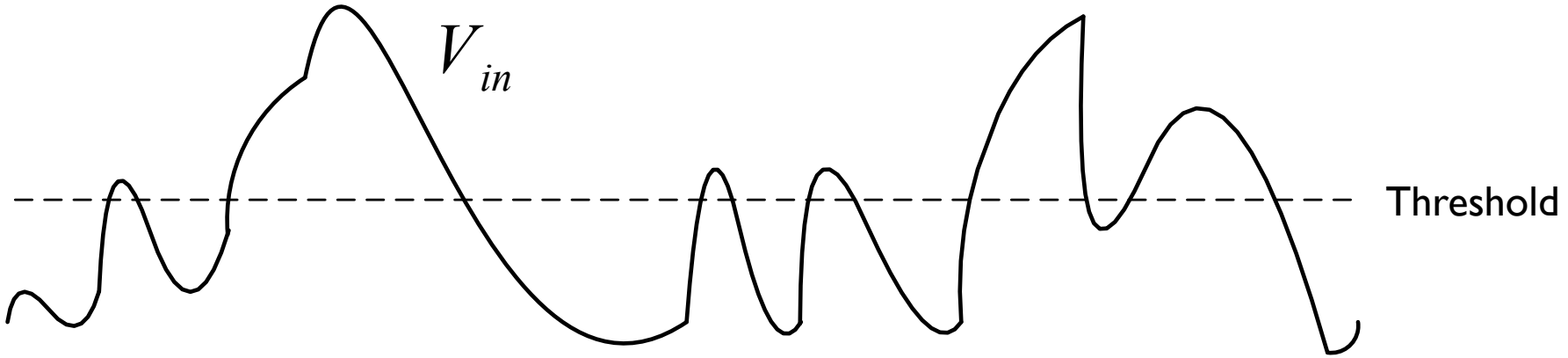
Input Data Register (IDR)

GPIO Pull-up/Pull-down Register (PUPDR)

00 = No pull-up, pull-down 01 = Pull-up

10 = Pull-down 11 = Reserved

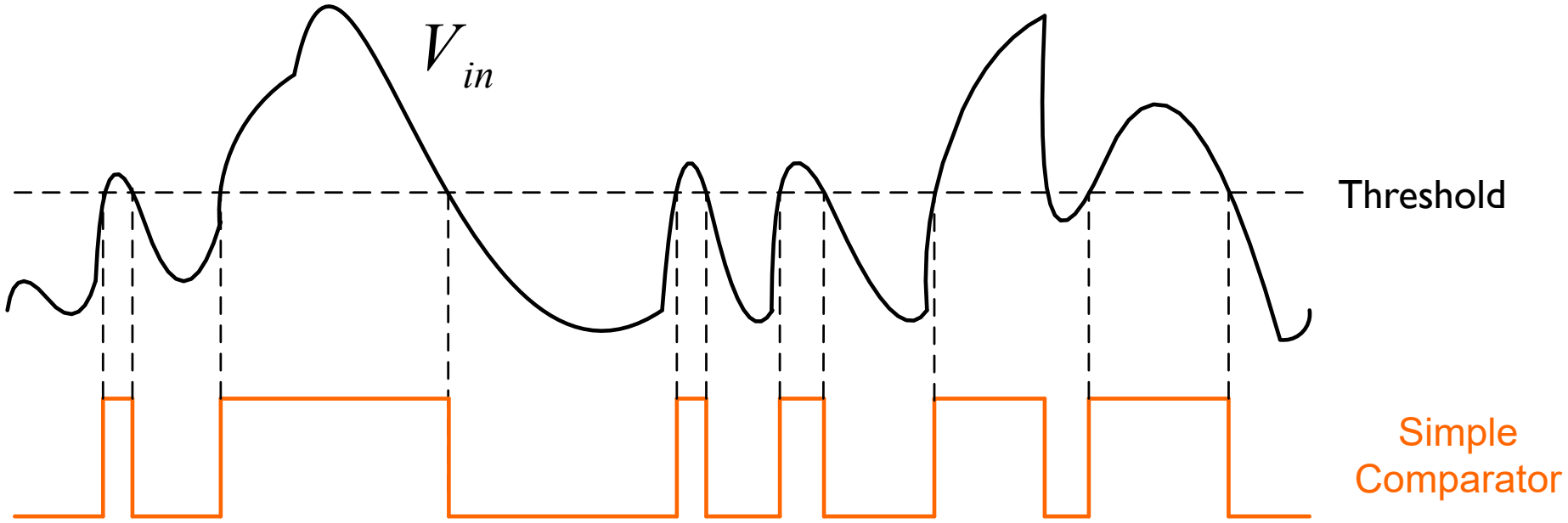
Schmitt Trigger



Analog signals

- Noisy
- Rise and fall slowly (small slew rate)

Schmitt Trigger



Schmitt Trigger