ELEC3850 - Embedded Systems 1

STM32 I/O
Interrupts

Brenton Schulz

University of Newcastle

September 15, 2020

Summary

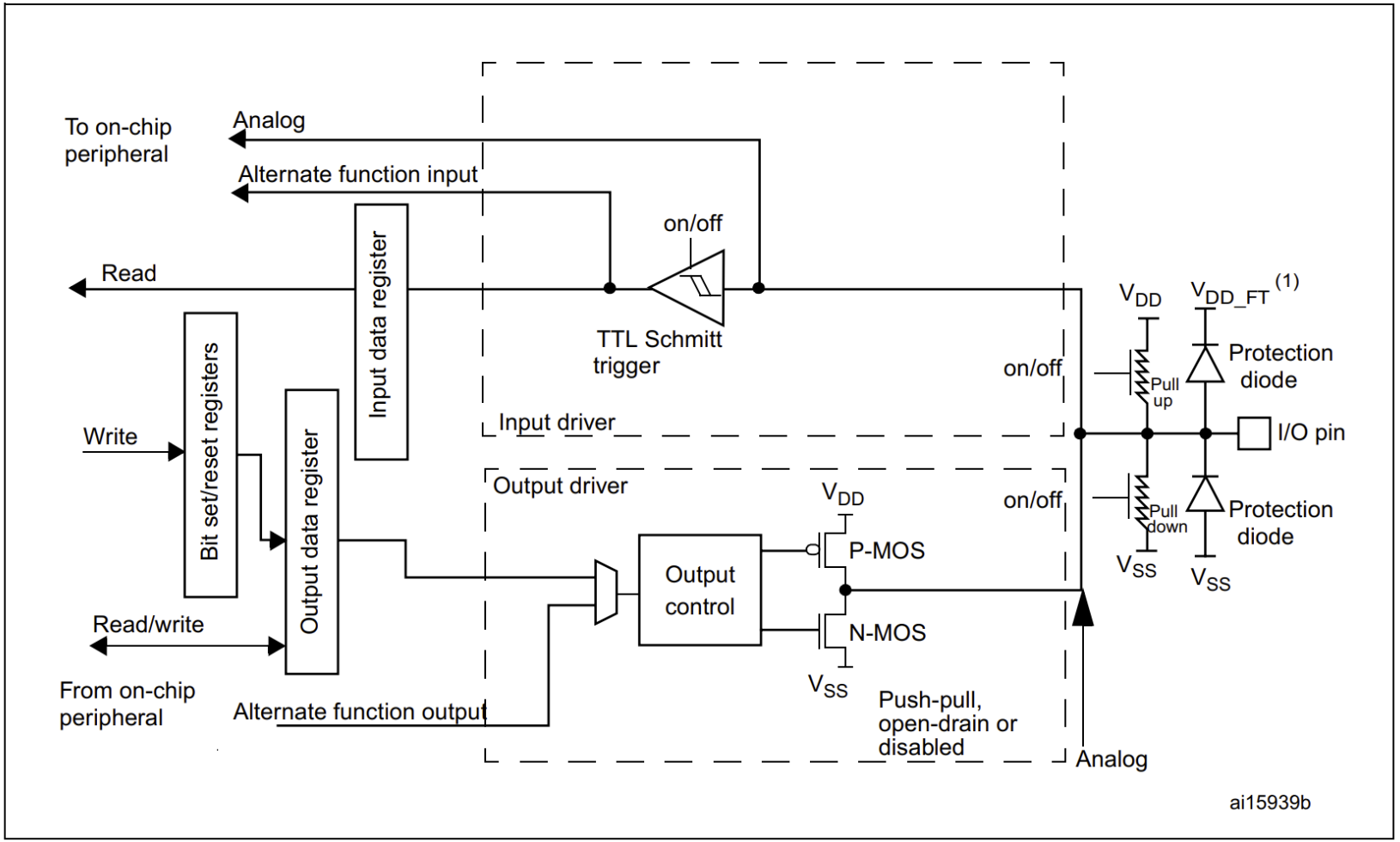
- Software driven GPIO
- Peripherals
  - SPI
  - I2C
  - UART
- Interrupts

Levels of Understanding

- Fundamental electronics
  - Transistors drive pins
  - What is push-pull Vs. open drain?
  - What are "pull-ups"?
- Datasheet / reference manual
  - Low-level configuration registers drive the GPIO circuit
- CubeMX
  - How does the datasheet translate to CubeMX?
  - In general: CubeMX assumes you have read the reference manual
- HAL
  - Using the HAL with confidence requires an understanding over everything above

GPIO Hardware

Figure 25. Basic structure of a five-volt tolerant I/O port bit



1. V_{DD_FT} is a potential specific to five-volt tolerant I/Os and different from V_{DD}.

GPIO Hardware

- STM32 pins can be configured as:
  - Digital outputs
    - Push-pull
    - Open drain
  - Digital inputs
    - With or without pull-up
    - With or without pull-down
  - Alternate Function I/Os
    - Outputs can also push-pull or open drain
    - AF inputs are analog when they drive internal ADCs
- STM32 outputs also have output bandwidth control
- NB: Outputs can have pull up/down enabled
  - This wastes a small amount of power
  - Useful if a pin swaps between input and output

GPIO Control Bits

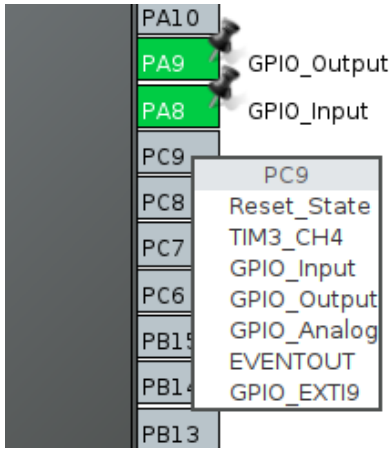
Table 35. Port bit configuration table(1)

MODER(i) [1:0]	OTYPER(i)	OSPEEDR(i) [B:A]	PUPDR(i) [1:0]		I/O configuration	
	0		0	0	GP output	PP

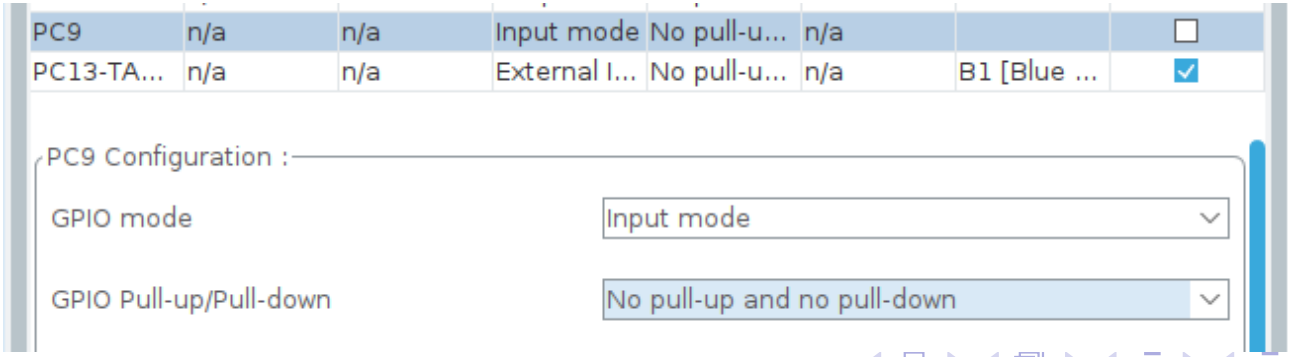
- Control bits:
  - MODER - Input/output control
  - OTYPER - Push-pull or open drain
  - OSPEEDR - Output bandwidth limiting
  - PUPDR - Pull up/down enable bits
- These control bits are packed into 32-bit GPIO registers:
  - GPIOx_MODER
  - GPIOx_OTYPER
  - GPIOx_OSPEEDR
  - GPIOx_PUPDR

CubeMX

- To configure GPIOs in CubeMX click pins and select a mode:



- Pull mode is then found in the left panel:



HAL Translation

- Open STM32CubeIDE, perform example configuration
- Observe:
  - `GPIO_TypeDef` - compare to GPIO registers in reference manual
  - `HAL_GPIO_*` functions in `stm32f4xx_hal_gpio.h`
  - HAL `GPIO_PinState` datatype

GPIO Interrupts

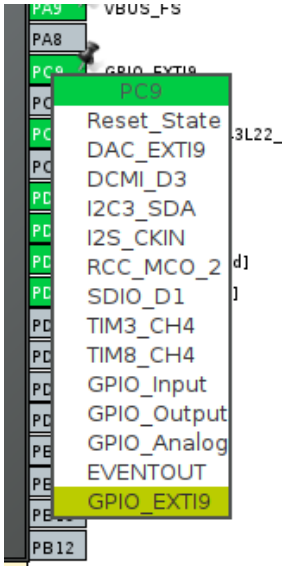
- STM32s can have interrupts triggered by GPIO state changes
  - Rising
  - Falling
  - Both
- There are 16 GPIO external interrupt (`EXTI`) sources on the STM32F407
  - Each can be triggered off 1 GPIO pin as-per Figure 42, p382 of RM0090
  - tl;dr: `PORT`xN triggers `EXTI`N
    - eg: PORTB2 can trigger EXTI2

GPIO Interrupts

- The possible interrupt vectors are:
  - EXTI0
  - EXTI1
  - EXTI2
  - EXTI3
  - EXTI4
  - EXTI9_5
  - EXTI10_15
- ie: GPIO pins 0-4 have unique interrupts, the others trigger a "grouped" interrupt
  - The ISR needs to determine which pin triggered the interrupt

GPIO Interrupts

- Example configuration: EXTI on PC9 on an STM32F407
  - Configure PC9 for GPIO_EXTI9



  - Select trigger mode in GPIO configuration panel
- Write `void EXTI9_5_IRQHandler(void)`
- Get ISR names from `startup_xxx.s`

GPIO Interrupts - Demonstration

- Observe Nucleo-F103RB project
- Note it includes `EXTI15_10_IRQHandler()`
- This function, in turn, calls `HAL_GPIO_EXTI_IRQHandler(GPIO_PIN_13)`
- If the `*_IRQHandler()` function does not exist you need to write it
  - It is the function *actually called* by the NVIC when the interrupt triggers
  - The HAL includes other interrupt functions with various names - these are **NOT** interrupt service routines called by the NVIC

GPIO Interrupts

- Crucial Note 1: EXTIs interrupts are not enabled by default!
- Enable with `HAL_NVIC_EnableIRQ(EXTI15_10_IRQn)`

GPIO Interrupts - Demonstration

- Crucial note 2: EXTIs are not cleared by hardware!
  - Software must clear the appropriate interrupt flag by writing a 1 to the correct bit in `EXTI_PI`
  - Recommended to use `__HAL_GPIO_EXTI_CLEAR_IT(GPIO_PIN);`
- Crucial note 3: When using shared EXTI interrupts use `__HAL_GPIO_EXTI_GET_IT(GPIO_PIN)` to test which EXTI triggered the ISR
- `stm32f1xx_hal_gpio.h` must be included for both the macro and `GPIO_PIN` definitions