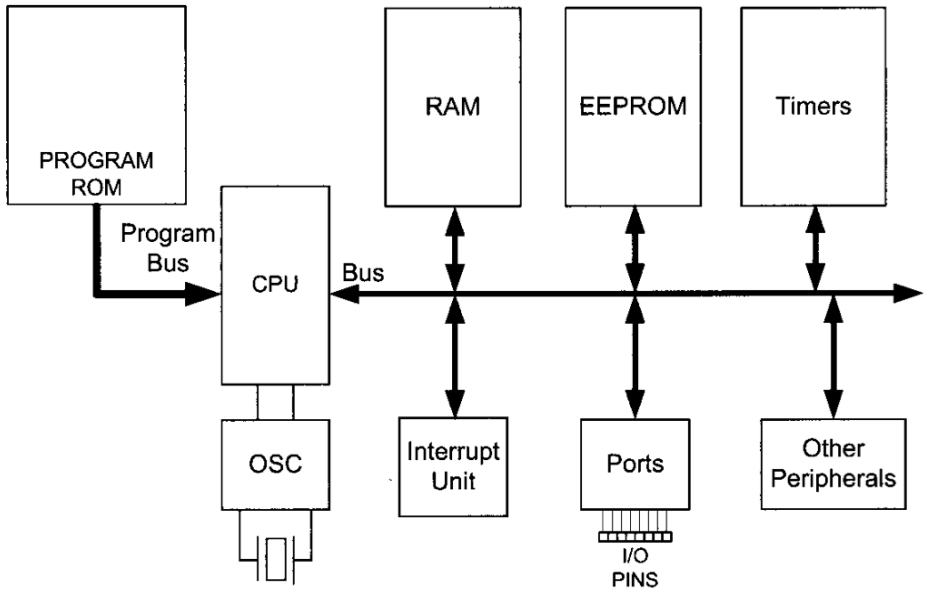
Outline

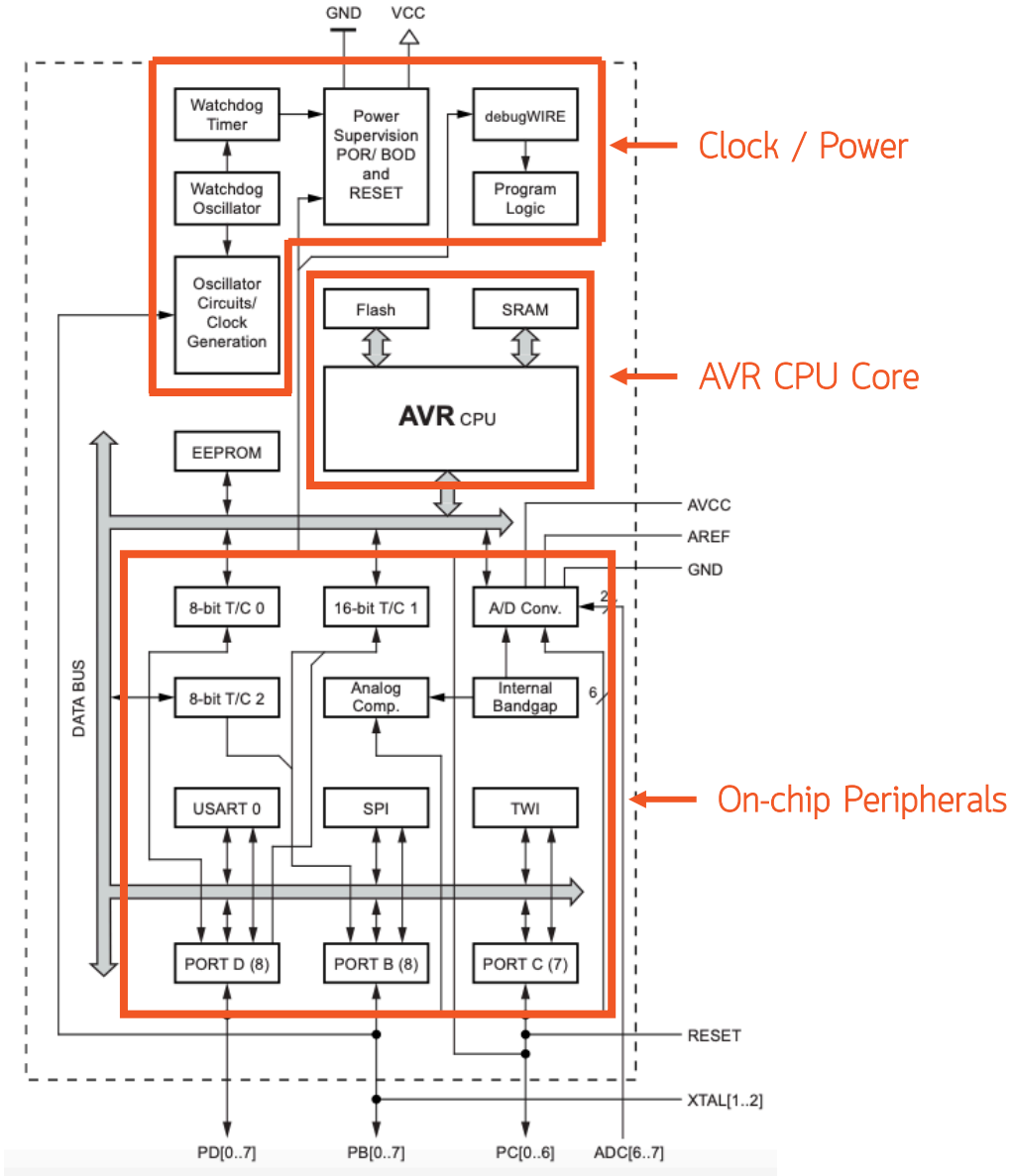
- AVR Architecture
- AVR I/O Ports
  - Chapter 14 (I/O Ports) in ATMega328P datasheet
- AVR Hardware Design
  - Minimal hardware connection
  - AVR Fuse bits
  - AVR Programming Interface

AVR I/O Ports

Recall AVR Architecture



Simplified AVR Architecture

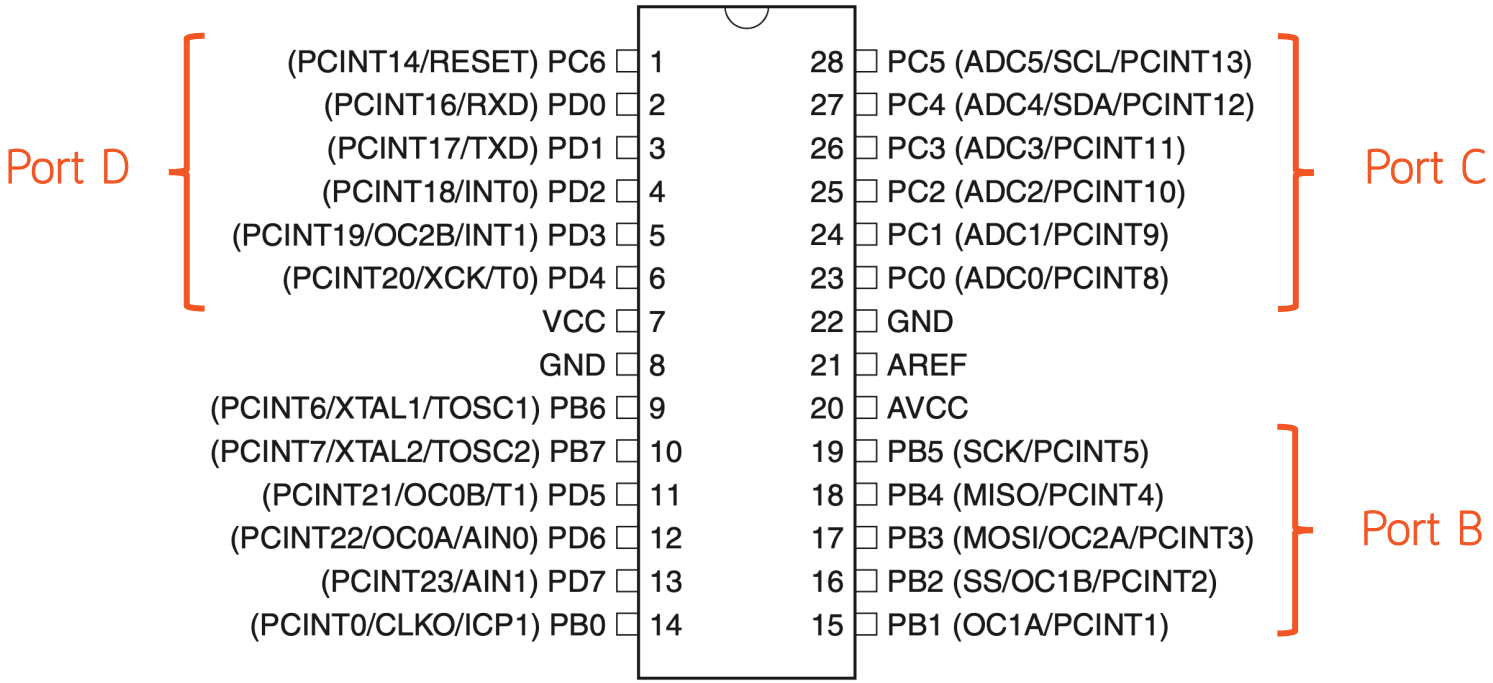


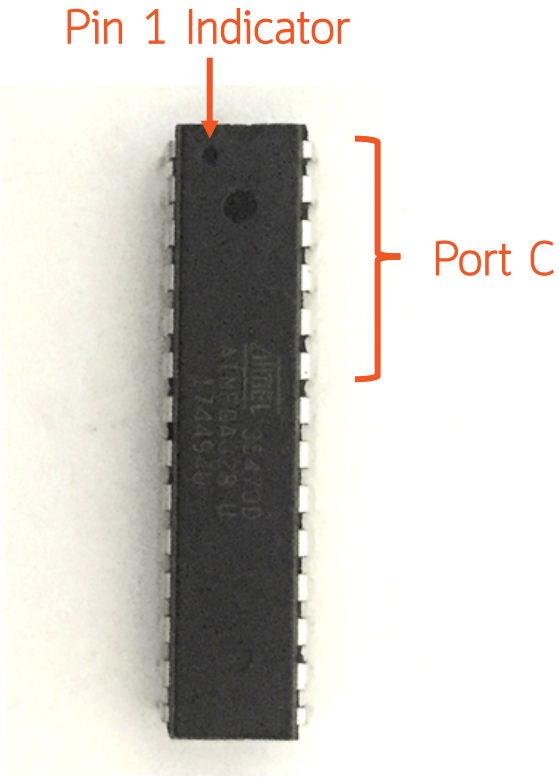
ATMega328P Architecture

AVR I/O Ports: General Features

- Pin direction (input/output), drive value (high/low for output pins), internal pull-up resistor (for input pins) can be changed individually
- Symmetrical drive characteristics with both high sink and source capability
  - 20 mA when Vcc = 5V or 10mA when Vcc = 3v
- Arrange in group of 8 bits that can be simultaneously configured

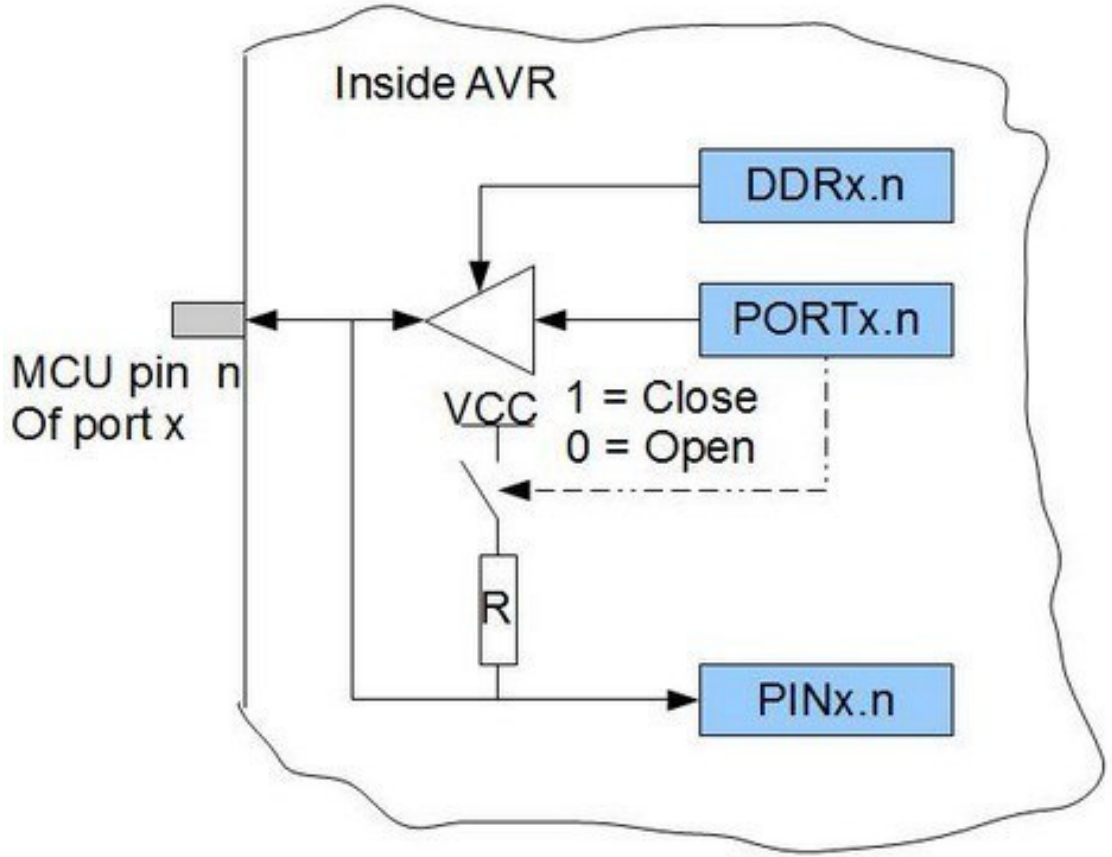
AVR I/O Ports: Physical Pin Location





- Each port group consists of 8 pins as register are 8 bits wide
- Some port group may have less than 8 pins

AVR I/O Ports: Simplify Circuit Diagram



Each port pin is controlled by three register bits:
DDRxn, PORTxn, and PINxn
e.g. PC5 is controller by DDRC5, PORTC5, PINC5

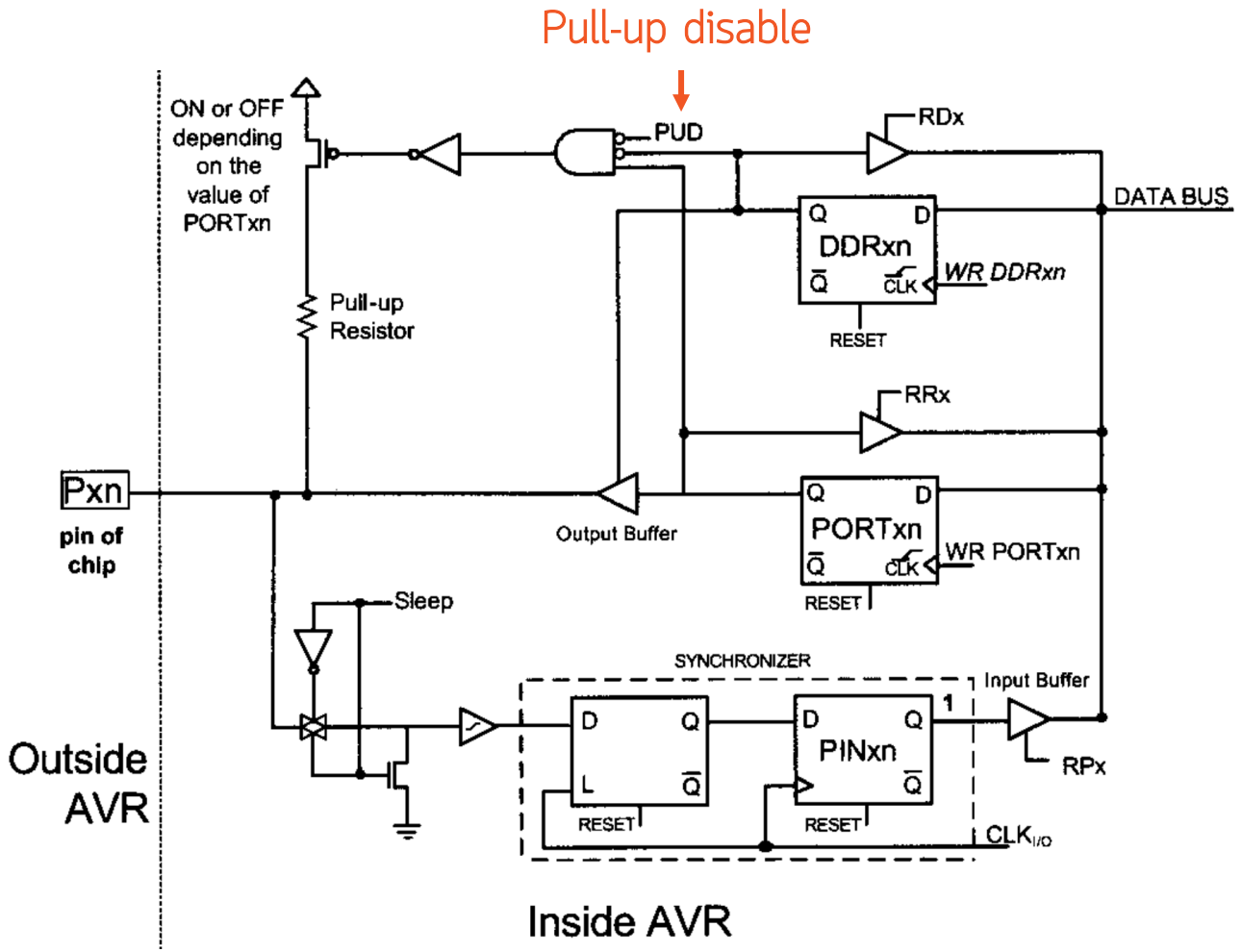
DDRx.n = 1 => Output
= 0 => Input (Tri-state)
PORTx.n = 1 => Output High / Enable pull-up
= 0 => Output Low / Disable pull-up
PINx.n = current logic level of the pin

Source: https://embedds.com/controlling-avr-io-ports-with-avr-gcc/

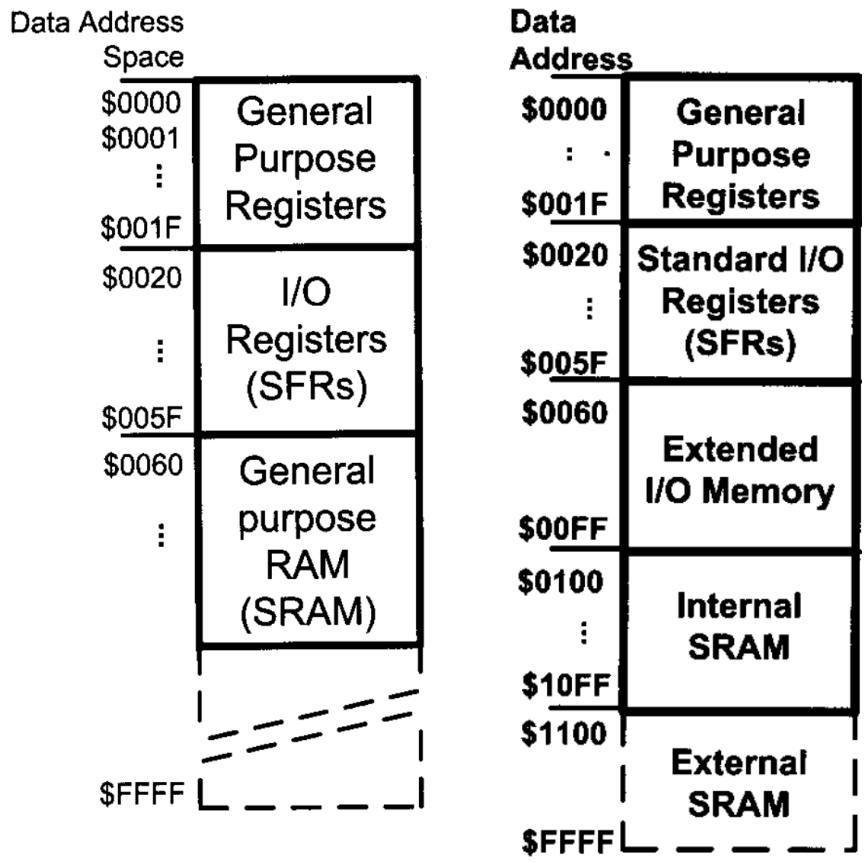
AVR I/O Ports: Register Bits Summary

DDxn	PORTxn	PUD (in MCUCR)	I/O	Pull-up	Comment
0	0	X	Input	No	Tri-state (Hi-Z)
0	1	0	Input	Yes	Pxn will source current if ext. pulled low.
0	1	1	Input	No	Tri-state (Hi-Z)
1	0	X	Output	No	Output Low (Sink)
1	1	X	Output	No	Output High (Source)

AVR I/O Ports: More Realistic diagram

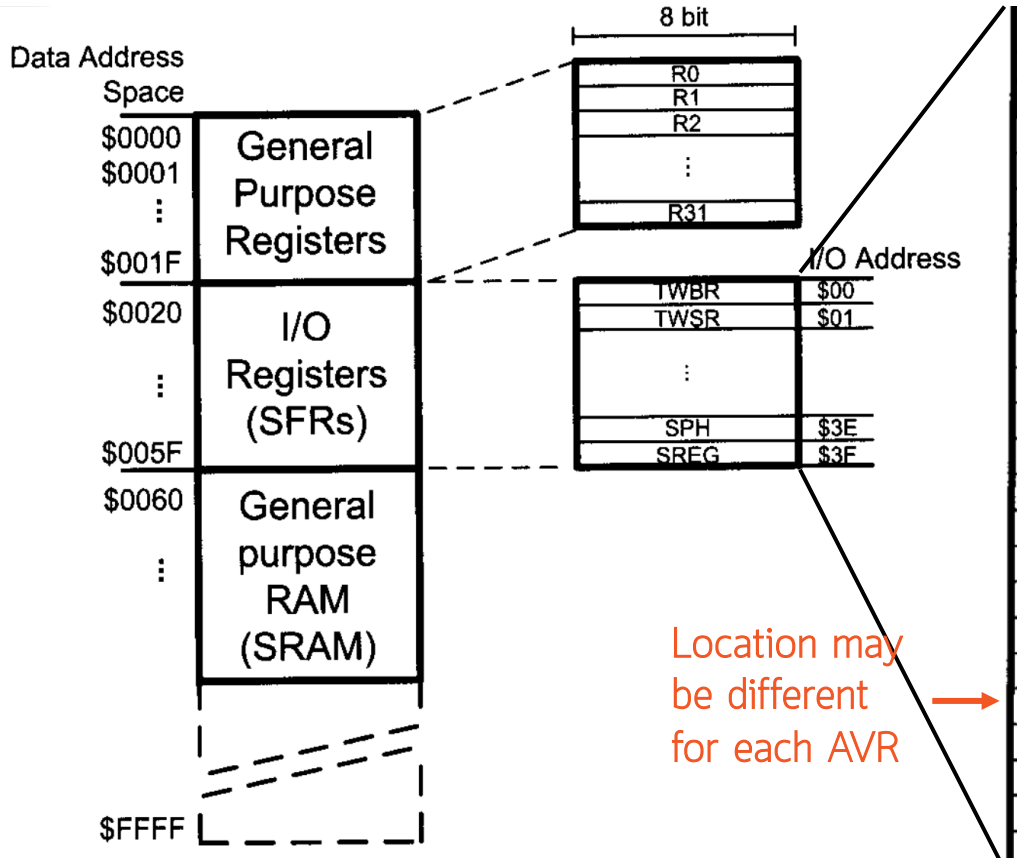


AVR I/O Ports: I/O Memory Location



- AVR data memory consists of 3 or 4 sections including general purpose registers, I/O registers, extended I/O memory, general purpose RAM
- Most recent AVRs have extended I/O memory due to the number of internal peripherals
- DDRX, PORTX and PINX are located in standard I/O or extended I/O memory

AVR I/O Ports: I/O Memory Location



Location may be different for each AVR

Address		Name
Mem.	I/O	
$20	$00	TWBR
$21	$01	TWSR
$22	$02	TWAR
$23	$03	TWDR
$24	$04	ADCL
$25	$05	ADCH
$26	$06	ADCSRA
$27	$07	ADMUX
$28	$08	ACSR
$29	$09	UBRRL
$2A	$0A	UCSRB
$2B	$0B	UCSRA
$2C	$0C	UDR
$2D	$0D	SPCR
$2E	$0E	SPSR
$2F	$0F	SPDR
$30	$10	PIND
$31	$11	DDRD
$32	$12	PORTD
$33	$13	PINC
$34	$14	DDRC
$35	$15	PORTC

Address		Name
Mem.	I/O	
$36	$16	PINB
$37	$17	DDRB
$38	$18	PORTB
$39	$19	PINA
$3A	$1A	DDRA
$3B	$1B	PORTA
$3C	$1C	EECR
$3D	$1D	EEDR
$3E	$1E	EEARL
$3F	$1F	EEARH
$40	$20	UBRRC
		UBRRH
$41	$21	WDTCR
$42	$22	ASSR
$43	$23	OCR2
$44	$24	TCNT2
$45	$25	TCCR2
$46	$26	ICR1L
$47	$27	ICR1H
$48	$28	OCR1BL
$49	$29	OCR1BH
$4A	$2A	OCR1AL

Address		Name
Mem.	I/O	
$4B	$2B	OCR1AH
$4C	$2C	TCNT1L
$4D	$2D	TCNT1H
$4E	$2E	TCCR1B
$4F	$2F	TCCR1A
$50	$30	SFIOR
$51	$31	OCDR
		OSCCAL
$52	$32	TCNT0
$53	$33	TCCR0
$54	$34	MCUCSR
$55	$35	MCUCR
$56	$36	TWCR
$57	$37	SPMCR
$58	$38	TIFR
$59	$39	TIMSK
$5A	$3A	GIFR
$5B	$3B	GICR
$5C	$3C	OCR0
$5D	$3D	SPL
$5E	$3E	SPH
$5F	$3F	SREG

AVR I/O Ports: Register Summary

MCUCR – MCU Control Register

Bit	7	6	5	4	3	2	1	0	
0x35 (0x55)	–	BODS[1]	BODSE[1]	PUD	–	–	IVSEL	IVCE	MCUCR
Read/Write	R	R/W	R/W	R/W	R	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

PORTB – The Port B Data Register

Bit	7	6	5	4	3	2	1	0	
0x05 (0x25)	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	PORTB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

DDRB – The Port B Data Direction Register

Bit	7	6	5	4	3	2	1	0	
0x04 (0x24)	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	DDRB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

PINB – The Port B Input Pins Address[1]

Bit	7	6	5	4	3	2	1	0	
0x03 (0x23)	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	PINB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	

AVR I/O Ports: I/O Programming

- Recall from previous slide that I/O registers location differs for each AVR
- The AVR GCC compiler knows where the registers are so we can write the following code to set pin PB0 and PB2 to output HIGH

```
DDRB = 0b00000101;
PORTB = 0b00000101;
```

How can we config 1 pin without affect other pins?

AVR I/O Ports: I/O Programming

- Each bit can individually be set using bitwise OR operator, clear using bitwise AND operator and toggle using XOR operator
- PBn constant can be used to refer to a bit in a register

```
MCUCR = 0b00010000;    =>  MCUCR |= (1 << PUD);
                                    0b00010000

DDRB = 0b00000101;     =>  DDRB |= _BV(PB0) | _BV(PB2);
                                   0b00000001 | 0b00000100
```

AVR I/O Ports: Electrical characteristic

29.1 Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on any Pin except $\overline{\text{RESET}}$ with respect to Ground	-0.5V to V_{CC}+0.5V
Voltage on $\overline{\text{RESET}}$ with respect to Ground	-0.5V to +13.0V
Maximum Operating Voltage	6.0V
DC Current per I/O Pin	40.0mA
DC Current V_{CC} and GND Pins	200.0mA

Table C-5: Fan-out for AVR Ports

Pin	Fan-out
IOL	20 mA
IOH	–20 mA
IIL	–1 µA
IIH	1 µA

Note: Negative current is defined as current sourced by the pin.

Figure 31-306. ATmega328: I/O Pin Output Voltage vs. Sink Current (V_{CC} = 5 V)

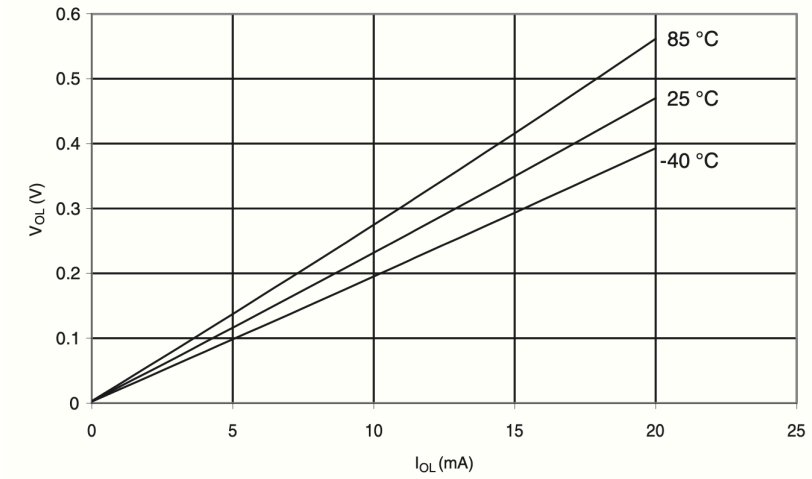
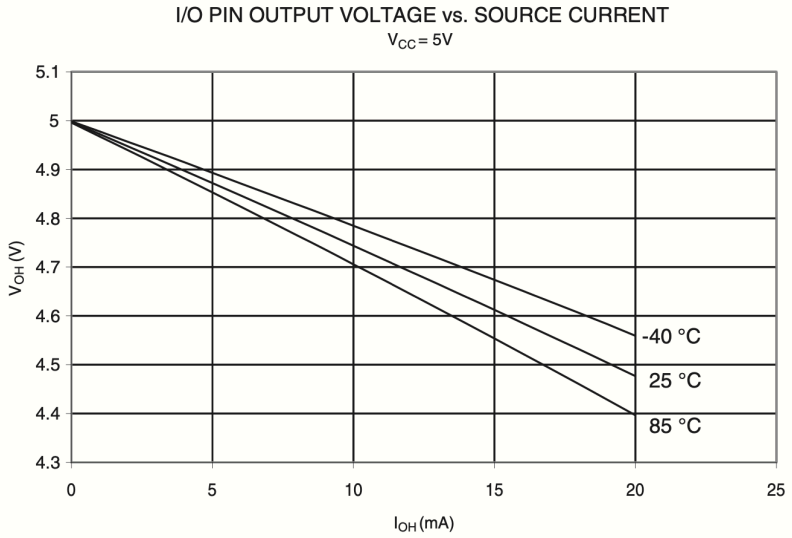


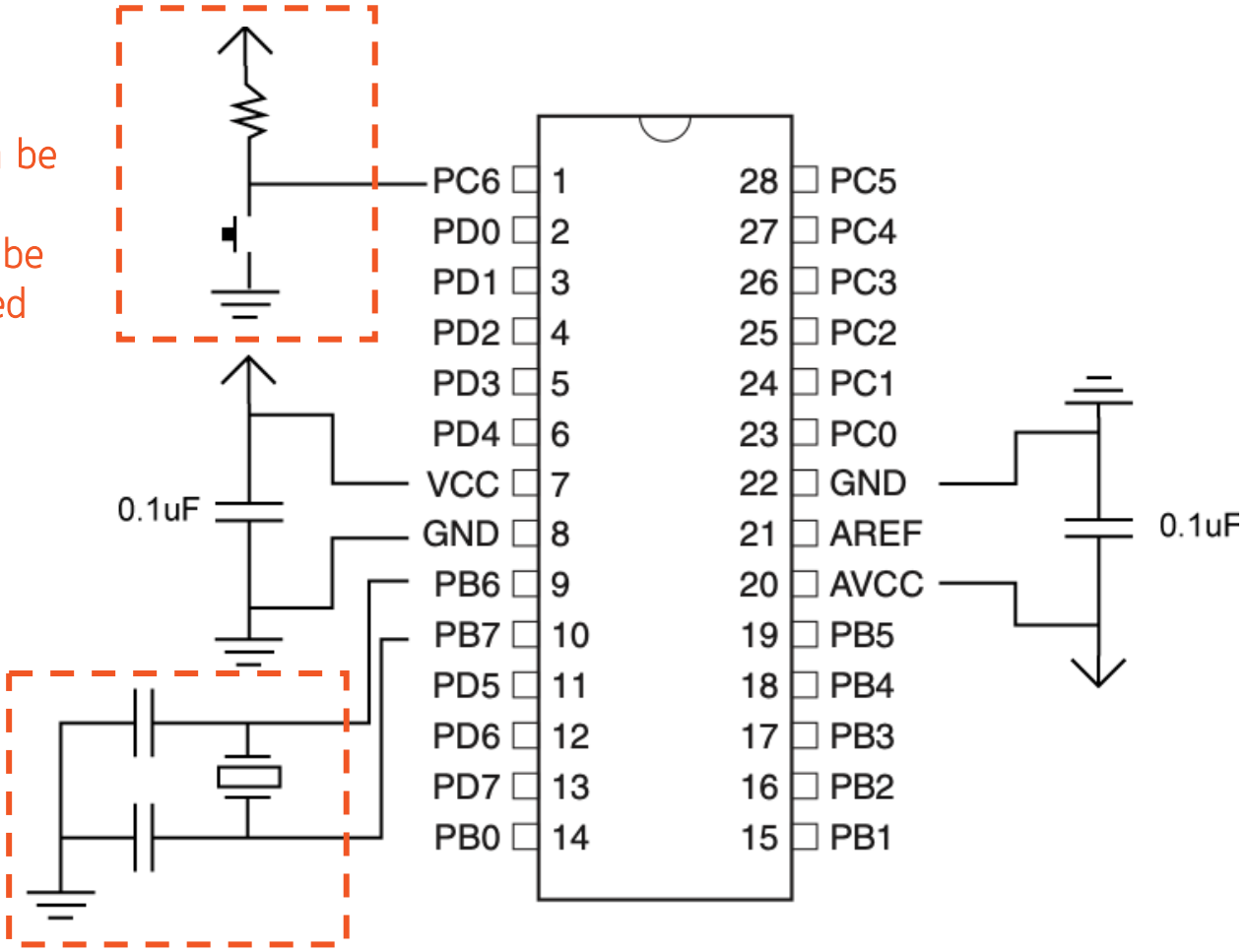
Figure 31-308. ATmega328: I/O Pin Output Voltage vs. Source Current (V_{CC} = 5 V)



AVR Hardware Design

Basic AVR Hardware Connection

- Internal resistor can be used
- Push button can be omited if board doesn't need to be manually resetted



Internal RC Osciallator can be used

AVR Fuse Bits

- ATMega328P has three Fuse bytes: Low, High and Extended for chip-level setup e.g.
  - Clock source
  - System clock pre-scalar
  - Brown-out Detector trigger level
  - Enable/disable watchdog timer, reset pin, programming method
- Fuses are read as logical zero "0" if they are programmed and "1" if they are unprogrammed
- https://www.engbedded.com/fusecalc/

AVR Fuse Bits: Extended Fuse Byte

Table 28-6. Extended Fuse Byte for ATmega328/328P

Extended Fuse Byte	Bit No	Description	Default Value
–	7	–	1
–	6	–	1
–	5	–	1
–	4	–	1
–	3	–	1
BODLEVEL2[(1)]	2	Brown-out Detector trigger level	1 (unprogrammed)
BODLEVEL1[(1)]	1	Brown-out Detector trigger level	1 (unprogrammed)
BODLEVEL0[(1)]	0	Brown-out Detector trigger level	1 (unprogrammed)

Table 29-12. BODLEVEL Fuse Coding[(1)(2)]

BODLEVEL 2:0 Fuses	Min. V_{BOT}	Typ V_{BOT}	Max V_{BOT}	Units
111	BOD Disabled			
110	1.7	1.8	2.0	V
101	2.5	2.7	2.9	
100	4.1	4.3	4.5	
011	Reserved			
010				
001				
000				

AVR Fuse Bits: High/Low Fuse Byte

Table 28-8. Fuse High Byte for **ATmega328/328P**

High Fuse Byte	Bit No	Description	Default Value
RSTDISBL[1]	7	External Reset Disable	1 (unprogrammed)
DWEN	6	debugWIRE Enable	1 (unprogrammed)
SPIEN[2]	5	Enable Serial Program and Data Downloading	0 (programmed, SPI programming enabled)
WDTON[3]	4	Watchdog Timer Always On	1 (unprogrammed)
EESAVE	3	EEPROM memory is preserved through the Chip Erase	1 (unprogrammed), EEPROM not reserved
BOOTSZ1	2	Select Boot Size (see Table 27-7 on page 284, Table 27-10 on page 285 and Table 27-13 on page 286 for details)	0 (programmed)[4]
BOOTSZ0	1	Select Boot Size (see Table 27-7 on page 284, Table 27-10 on page 285 and Table 27-13 on page 286 for details)	0 (programmed)[4]
BOOTRST	0	Select Reset Vector	1 (unprogrammed)

Table 28-9. Fuse Low Byte

Low Fuse Byte	Bit No	Description	Default Value
CKDIV8[4]	7	Divide clock by 8	0 (programmed)
CKOUT[3]	6	Clock output	1 (unprogrammed)
SUT1	5	Select start-up time	1 (unprogrammed)[1]
SUT0	4	Select start-up time	0 (programmed)[1]
CKSEL3	3	Select Clock source	0 (programmed)[2]
CKSEL2	2	Select Clock source	0 (programmed)[2]
CKSEL1	1	Select Clock source	1 (unprogrammed)[2]
CKSEL0	0	Select Clock source	0 (programmed)[2]

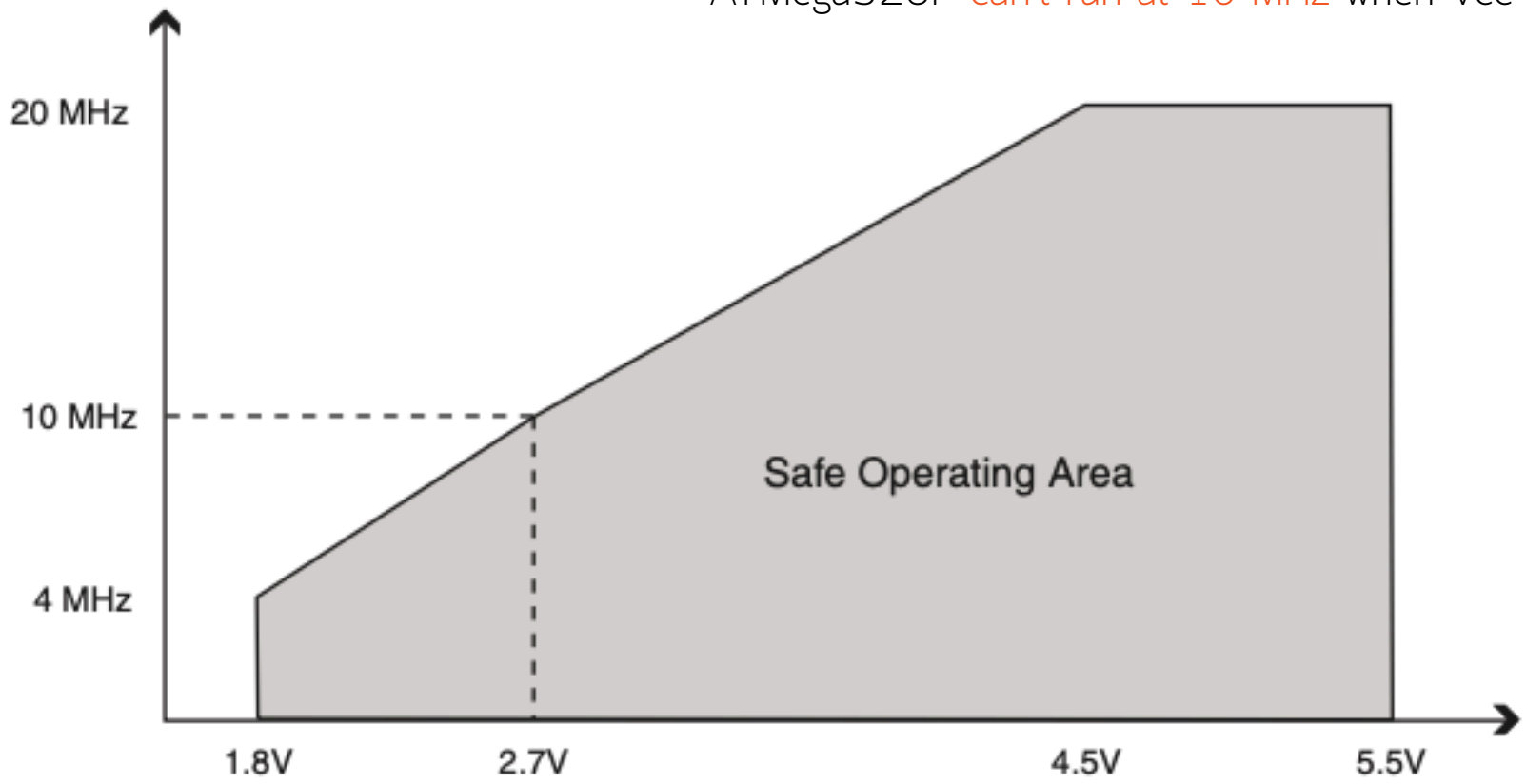
Table 9-1. Device Clocking Options Select[1]

Device Clocking Option	CKSEL3...0
Low Power Crystal Oscillator	1111 - 1000
Full Swing Crystal Oscillator	0111 - 0110
Low Frequency Crystal Oscillator	0101 - 0100
Internal 128kHz RC Oscillator	0011
Calibrated Internal RC Oscillator	0010
External Clock	0000
Reserved	0001

Note: 1. For all fuses "1" means unprogrammed while "0" means programmed.

AVR Max Freq. vs VCC Consideration

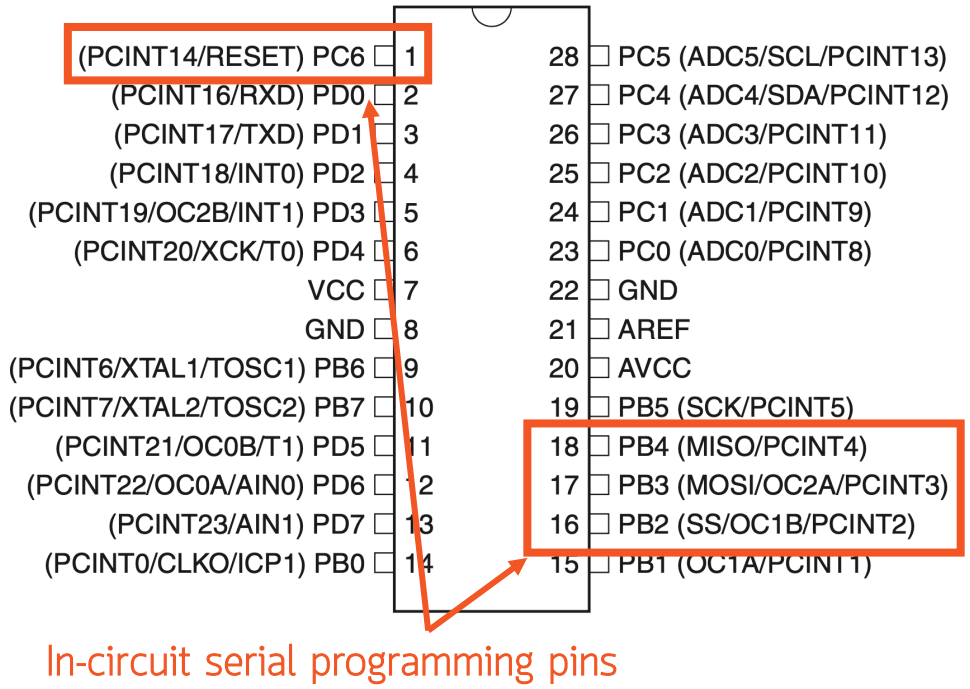
Figure 29-1. Maximum Frequency vs. V_{CC}



ATMega328P can't run at 16 MHz when Vcc = 3.3V !!!

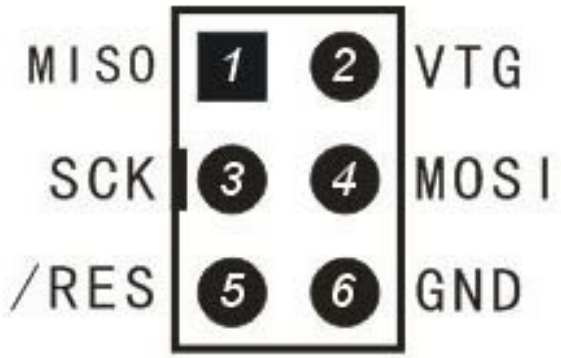
AVR Serial Programming Connection

- Most AVR can be programmed through the SPI interface except the most recent tinyAVR 0- and 1-series, and megaAVR 0-series



In-circuit serial programming pins

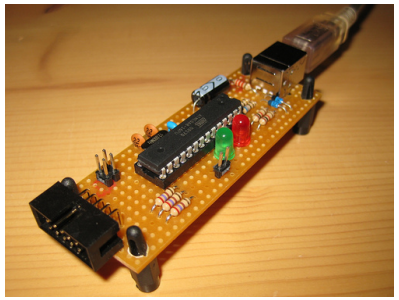
6-pin vs 10-pin
AVR ICSP Connector Layout



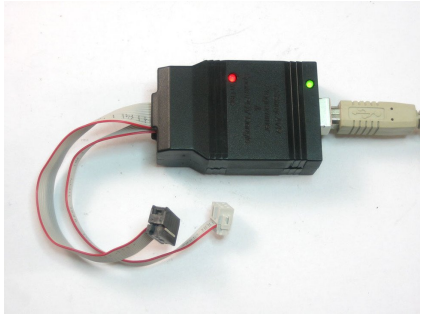
Some AVR Programmers

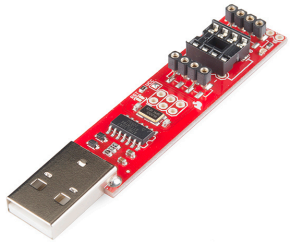
1. Opensource Programmer

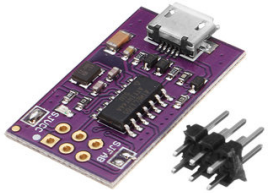
- USBasp



- USBtinyISP







2. Proprietary Programmer

- Atmel ICE

- Microchip Snap