I/O Module - Overview

Figure: I/O Module Block Diagram
I/O Module - Changes from ATMega

▶ More numerous, powerful and complex(!) peripherals than Mega
▶ New programming model greatly improved.
▶ New pin design increases output electrical choices.
▶ I/O ports are groups of 8 pins (A-F)
▶ Each I/O pin is controlled by 22 different registers!
▶ Pins can take on one of eight different electrical configurations.
▶ Basic functionality still set by OUT, DIR, and IN registers.
▶ OUT and DIR are augmented by direct *set*, *clear*, and *toggle* registers.
I/O Module - DIR, OUT and IN Registers

- DIRn, OUTn, and INn relationships assuming totem-pole pin:
  - "Z" indicates high-impedance driver
  - "X" indicates unknown value, driven by external logic

<table>
<thead>
<tr>
<th>DIRn</th>
<th>OUTn</th>
<th>Pn</th>
<th>INn</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Z</td>
<td>X</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>Z</td>
<td>X</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Table: DIR, OUT, and IN Relationships
I/O Module - SET, CLR and TGL Registers

- DIR and OUT can be directly manipulated using *strobe* registers
  - DIR manipulated with DIRSET, DIRCLR, DIRTGL.
  - OUT manipulated with OUTSET, OUTCLR, OUTTGL.
  - Reading the strobe registers returns value of OUT, *not* the strobe register!!
I/O Module - SET, CLR, TGL; Examples

PORTA.DIRSET = (PIN6_bm | PIN0_bm);

DIR register (before)  0b1011_0011
Bit mask            0b0100_0001
DIR register (after)  0b1111_0011

PORTA.DIRCLR = (PIN7_bm | PIN3_bm | PIN2_bm);

DIR register (before)  0b1101_1011
Bit mask            0b1000_1100
DIR register (after)  0b0101_0011

PORTA.DIRTGL = (PIN7_bm | PIN6_bm | PIN4_bm | PIN3_bm);

DIR register (before)  0b0111_0111
Bit mask            0b1101_1000
DIR register (after)  0b1010_1111
I/O Module - SET, CLR and TGL Registers

- Advantages of SET, CLR and TGL registers
  - prevents non-atomic read-modify-write sequences
  - saves memory space
  - minimizes execution time

```c
// Set bit 3 in DIR register. Don’t affect other bits.
PORTA.DIR |= PIN3_bm; // set bit three using bit mask
// non-atomic read/modify/write results
// 224: 80 91 00 06 lds r24, 0x0600
// 228: 88 60 ori r24, 0x08 ; 8
// 22a: 80 93 00 06 sts 0x0600, r24

// PORTA.DIRSET = PIN3_bm; // use OUTSET to set bit 3
// smaller, faster, atomic transaction
// 22e: 88 e0 ldi r24, 0x08 ; 8
// 230: 80 93 01 06 sts 0x0601, r24

// 5 locations in program memory versus 3
// 5 clock cycles versus 3
```
I/O Module - IN Register

- All digital pin inputs are synchronized to the I/O clock

**Figure: I/O Pin Synchronization**
Synchronization creates up to a two clock-cycle delay.
I/O Module - Other Registers

- PINnCTRL - slew rate, inverted I/O, output type, input sense
- We will discuss the following registers later:
  - Interrupt system registers:
    - INTCTRL - Interrupt Control Register
    - INT0MASK - Selects pin for interrupt
    - INT1MASK - Selects pin for interrupt
    - INTFLAGS - Signals interrupt on pin
  - REMAP - Moves SPI, USART0, TCNT0 pins within a port
- There are five more registers, we won’t use them.(!)