

Digital Inputs

Last updated 6/30/22

Digital Inputs

- Many pins on the Nucleo board are capable of acting as digital inputs
 - Digital input: reads the pin with
 - Electrical: 0.0v or 3.3v
 - Interprets the value as
 - logical: '1' or '0'
- There are limitations on voltage levels at the input
 - Max: $V_{IL} < 1.0V$ and $V_{IH} > 2.5V$

Digital Inputs

- Digital Inputs are created by creating `DigitalIn` “objects”

- Creating an object

- called `someName`
- tied to pin `somePin`

```
DigitalIn someName(somePin);
```

- `someName` now references the digital input object

- Reading a 1 or 0 on the digital input pin

```
foo = someName.read(); // reads the pin – foo is 0 or 1
```

Digital Inputs

- DigitalIn example
 - DigitalIn object named MyInput connected to pin D5

```
////////////////////////////////////
//
// digital_in_ex project
//
// created 7/8/21 by tj
// rev 0
//
////////////////////////////////////
//
// Example of using DigitalIn class
//
// This program prints out the value read by DigitalIn class
// a wire is tied to D5 and switched from gnd to vdd and back
//
////////////////////////////////////

#include "mbed.h"
#include <stdio.h>          // only needed when printing

#define T_WAIT 2000000    // in us

// Global HARDWARE Objects
// Create the digital input object tied to D5
DigitalIn MyInput(D5);

int main(void){
    setbuf(stdout, NULL); // disable buffering when printing

    // splash
    printf("\n\ndigital_in_ex\n");
    printf("Using Mbed OS version %d.%d.%d\n",
           MBED_MAJOR_VERSION, MBED_MINOR_VERSION, MBED_PATCH_VERSION);

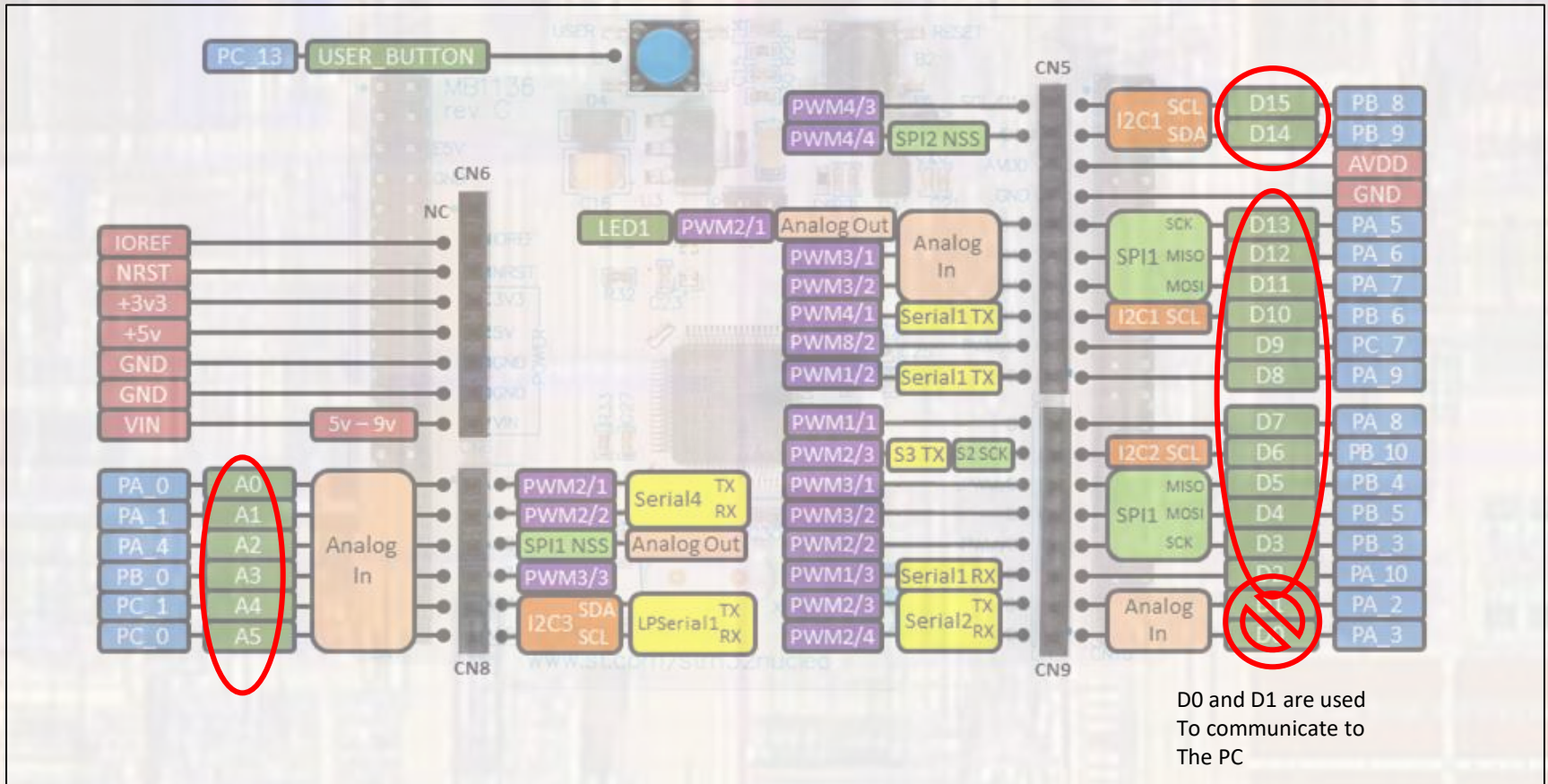
    // working variables
    uint8_t in_val;

    // run an infinite loop
    while(1){
        // check pin and print value
        in_val = MyInput.read();
        printf("Input is now: %i\n", in_val);
        wait_us(T_WAIT);
    } // end while

    return 0;
} // end main
```

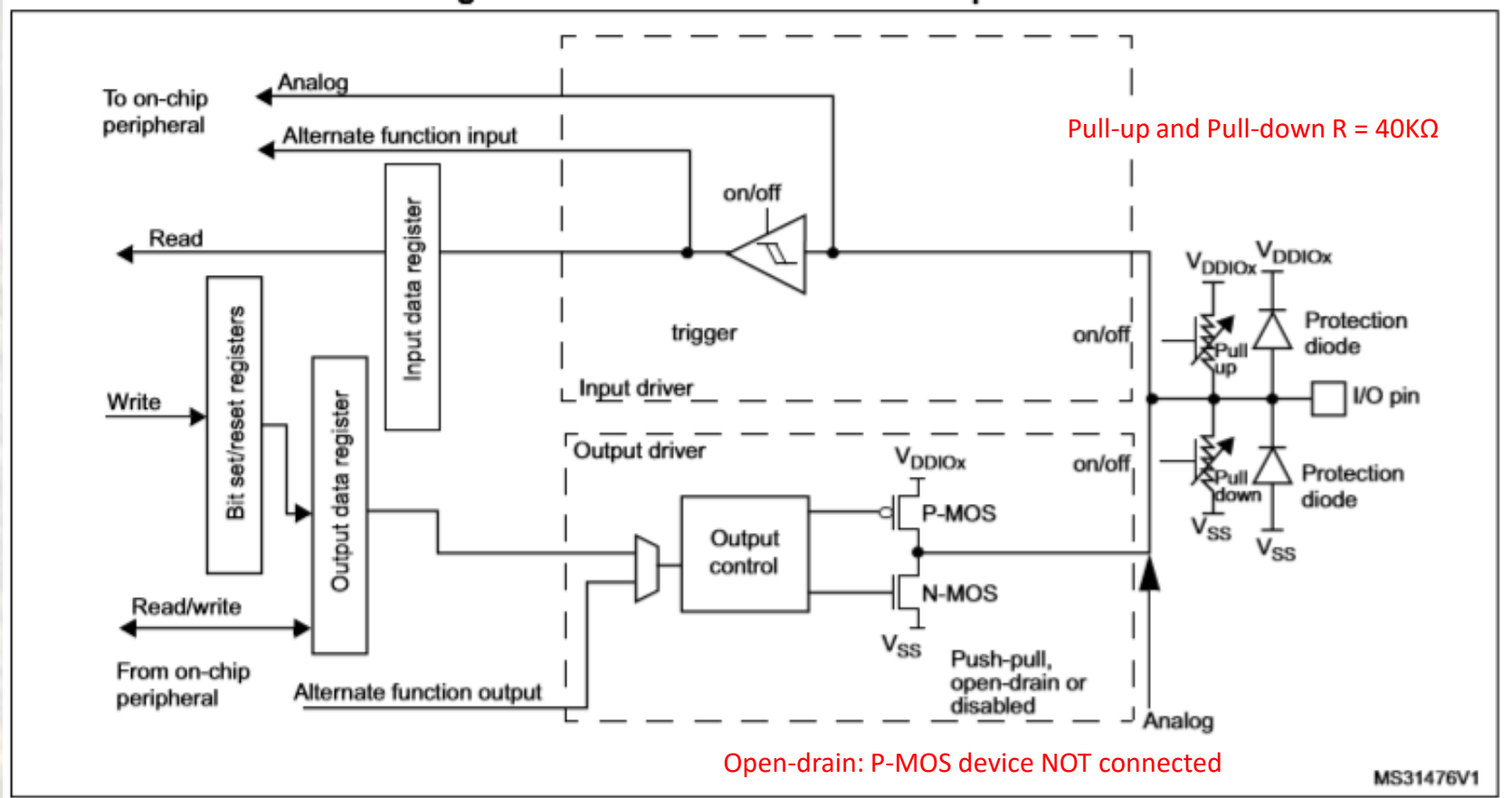
Digital Inputs

- DigitalIn Pins – Arduino Headers



Digital Inputs

- Pin Structure



Digital Inputs

- DigitalIn Class

Public Member Functions	
	<code>DigitalIn (PinName pin)</code>
	Create a <code>DigitalIn</code> connected to the specified pin. More...
	<code>DigitalIn (PinName pin, PinMode mode)</code>
	Create a <code>DigitalIn</code> connected to the specified pin. More...
	<code>~DigitalIn ()</code>
	Class destructor, deinitialize the pin. More...
int	<code>read ()</code>
	Read the input, represented as 0 or 1 (int) More...
void	<code>mode (PinMode pull)</code>
	Set the input pin mode. More...
int	<code>is_connected ()</code>
	Return the output setting, represented as 0 or 1 (int) More...
	<code>operator int ()</code>
	An operator shorthand for <code>read()</code> More...

`DigitalIn MyPin(D5)`

`DigitalIn MyPin(D5, PullUp)`

PullUp, PullDown, PullNone, OpenDrain

`foo = MyPin.read()`

`MyPin.mode(PullNone)`

PullUp, PullDown, PullNone, OpenDrain

`boo = MyPin.is_connected()`

`foo = MyPin`

