University of Glasgow

电子科技大学格拉斯哥学院
Glasgow College, UESTC

UESTC1008: Microelectronic Systems

Academic year 2019/2020 – Semester 2 – Presentation 4

Qammer H. Abbasi, Lei Zhang, **Sajjad Hussain**and Guodong Zhao
*{qammer.abbasi,Lei.zhang,**sajjad.Hussain**,guodong.zhao}@glasgow.ac.uk*

This week topics

- Digital input/output
- CMOS as switch
- Digital to Analog Converters

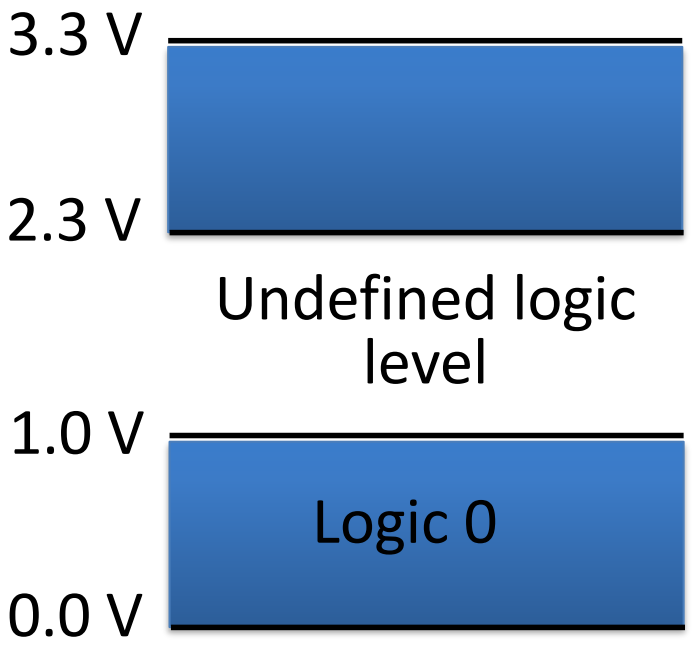
DIGITAL IN/OUT

Digital vs Analog

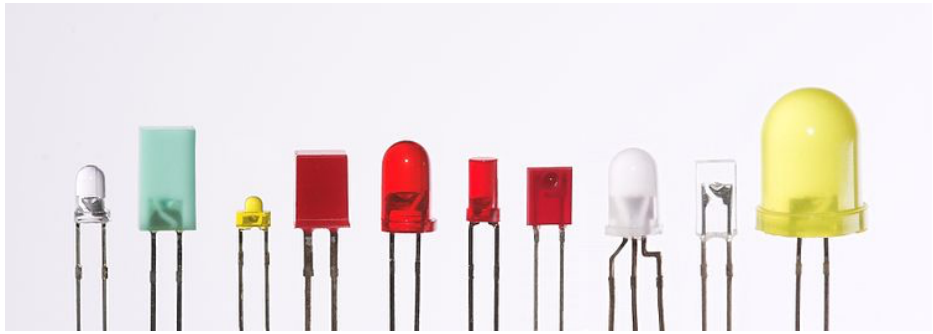
- https://www.diffen.com/difference/Analog_vs_Digital

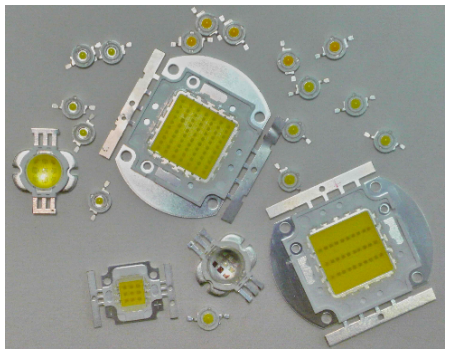
Digital Input and Output

0V	3.3V
Open	Closed
Off	On
Low	High
Clear	Set
logic 0	logic 1
False	True



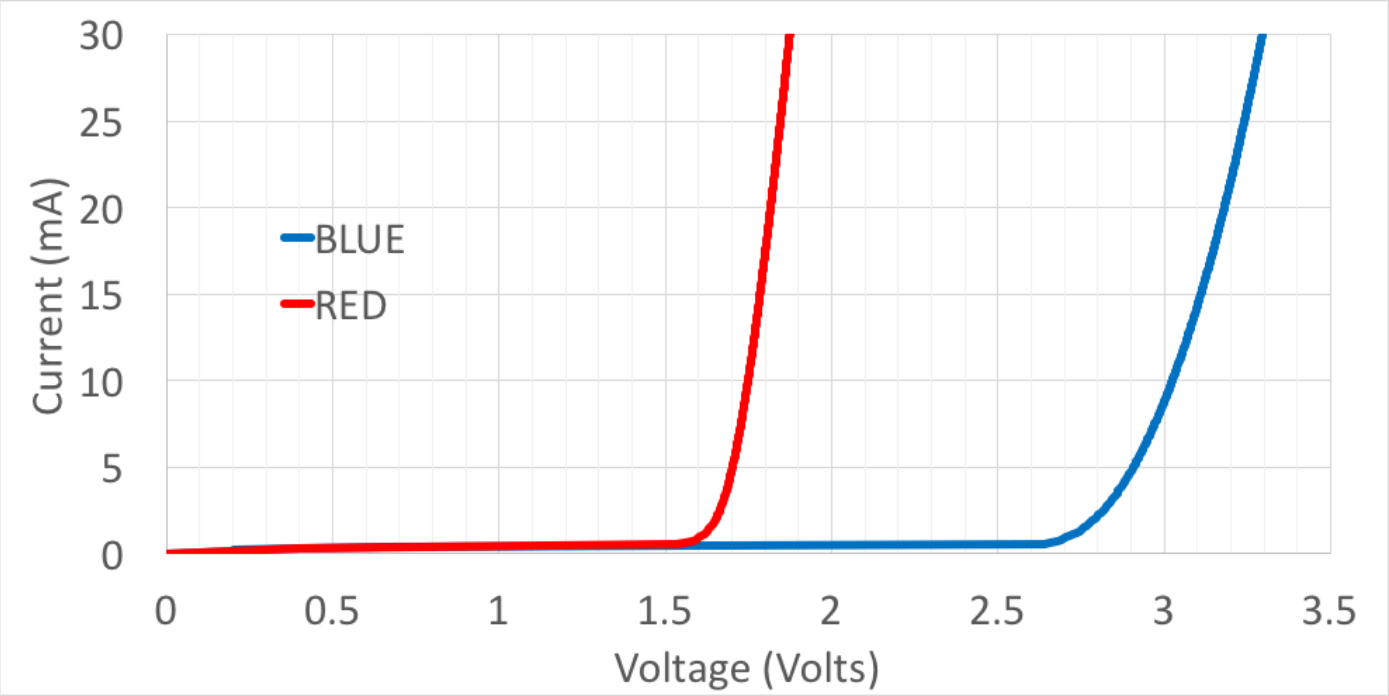
LEDs





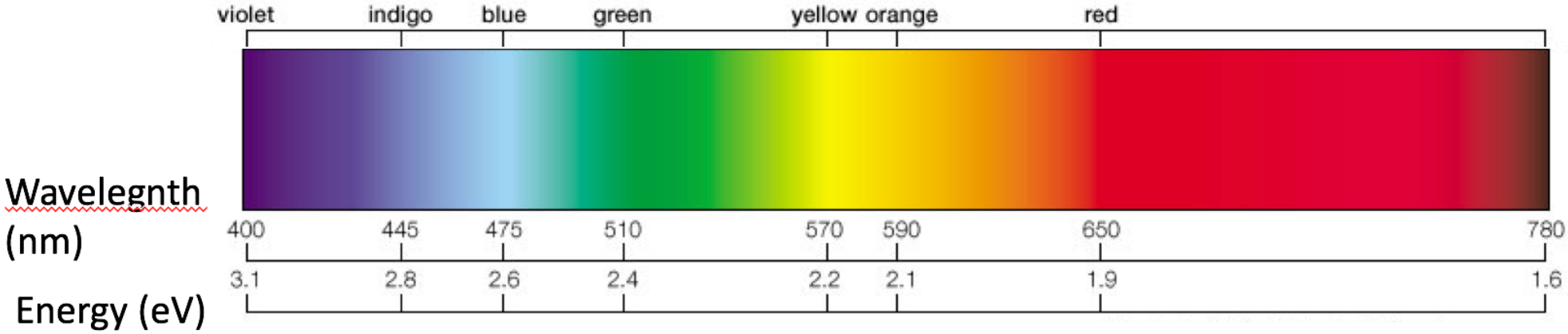
- Range from low power : Ideal as an indicator i.e. On/Off, or Run/Fault, or PANIC : fridge empty
- To high brightness : Ideal for lighting, displays
- The semiconductor material it’s made from determines the color
- Most of the LED is in packaging, and the colored done is just to filter out unwanted light and focus the emission.

V-I characteristic

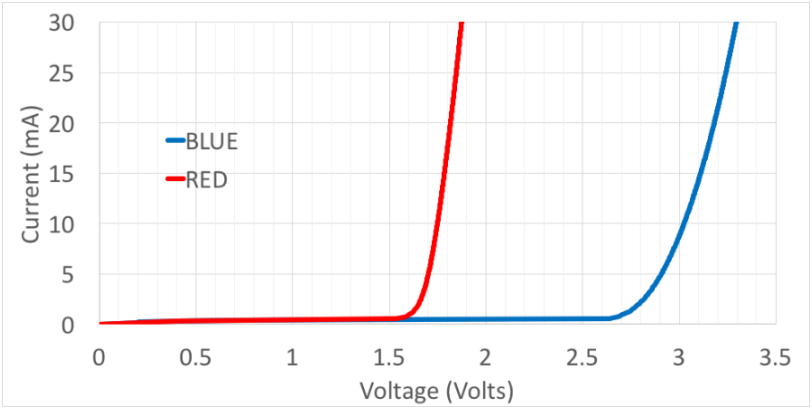


- Since it's a diode, we expect a sharp turn on at a 'turn-on' voltage
- Above this voltage, significant current flows
- The diode is not ideal, there is still a slope – why?

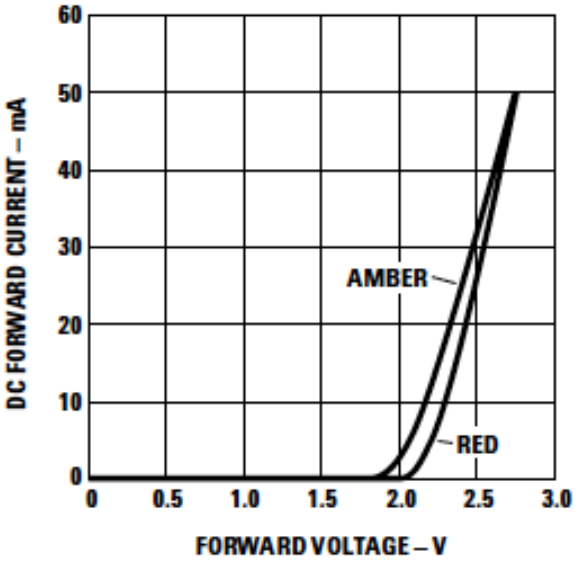
Wavelength, Energy, Voltage



- Your eye can see colors ranging from 400nm (dark violet) to 750nm dark red
- Blue light is higher energy than red
  - needs more voltage



Connecting LEDs



LEDs are diodes – two terminals

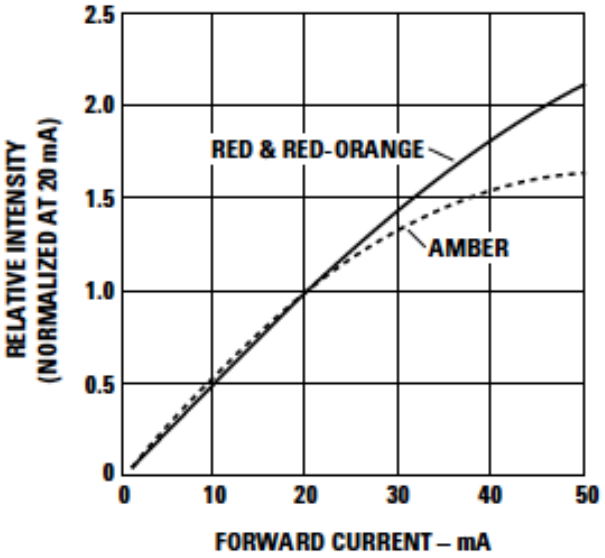
anode and cathode.

conduct current in one direction only.

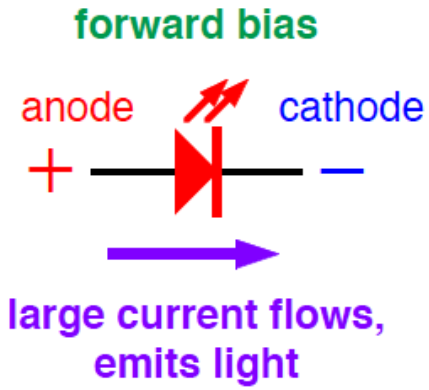
When conducting current

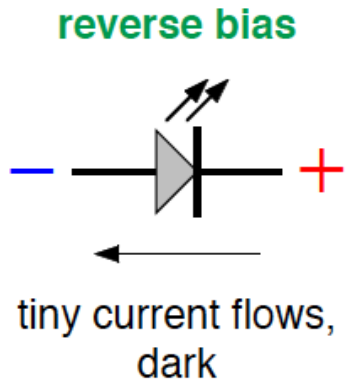
they emit light.

the diode is “forward” biased.



Increasing “forward” current increases light output





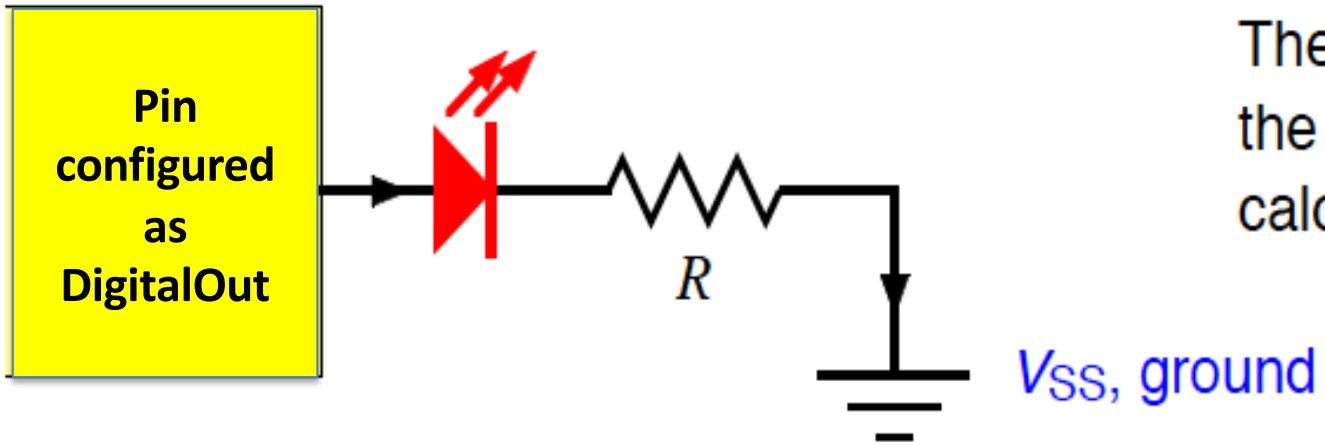
Remember that LEDs are **diodes**, so we must connect them the correct way round! This can be done in two ways.

1. Suppose that we want the LED to illuminate when the logical value of the output is high (1) — **active high**.

   Thus the LED should light when the pin is at V_{DD} (supply, positive, logic 1) but not when the pin is at V_{SS} (ground, negative, logic 0).

   We therefore connect the LED between the pin and V_{SS}.

   Conventional current flows from the pin to V_{SS} with this connection.



The resistor R is to limit the current — we'll calculate its value shortly.

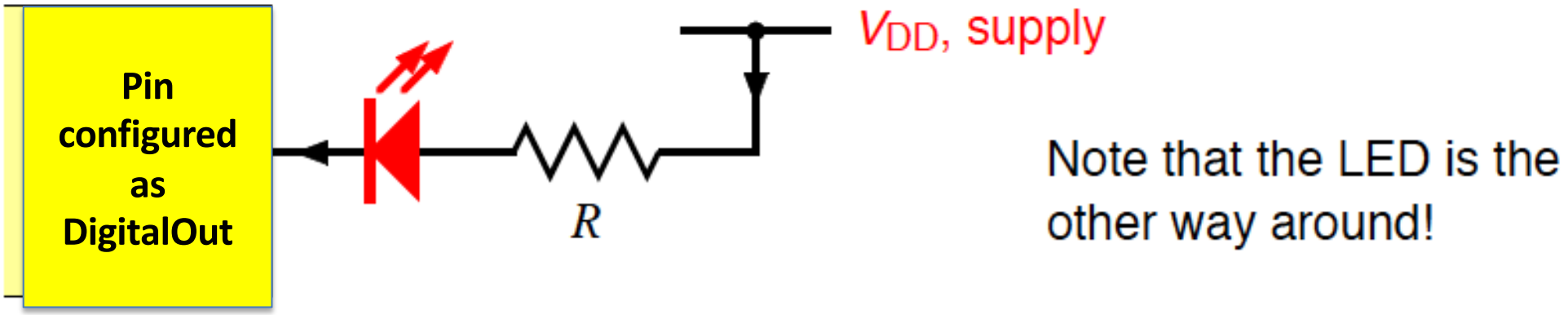
It might seem obvious that you would always want to connect an LED like this but there is a second, complementary way in which this can be done.

2. Suppose that we want the LED to illuminate when the logical value of the output is low (0) — **active low**.

   Thus the LED should light when the pin is at V_{SS} (ground, negative, logic 0) but not when the pin is at V_{DD} (supply, positive, logic 1).

   We therefore connect the LED between the pin and V_{DD}.

   Conventional current flows from V_{DD} to the pin with this connection.

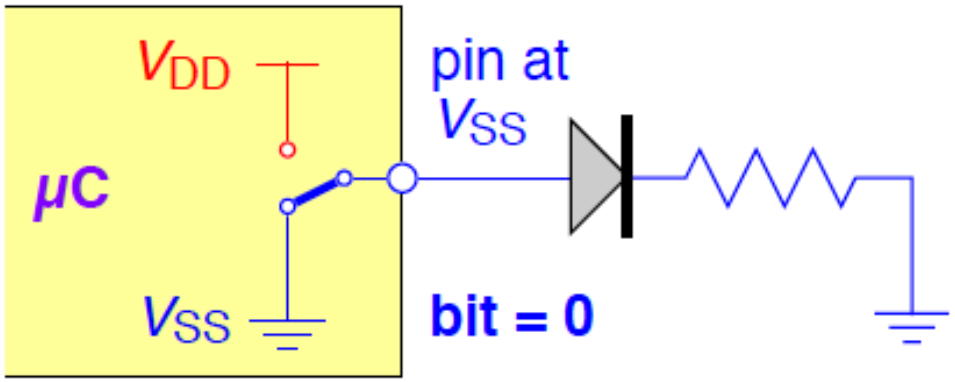


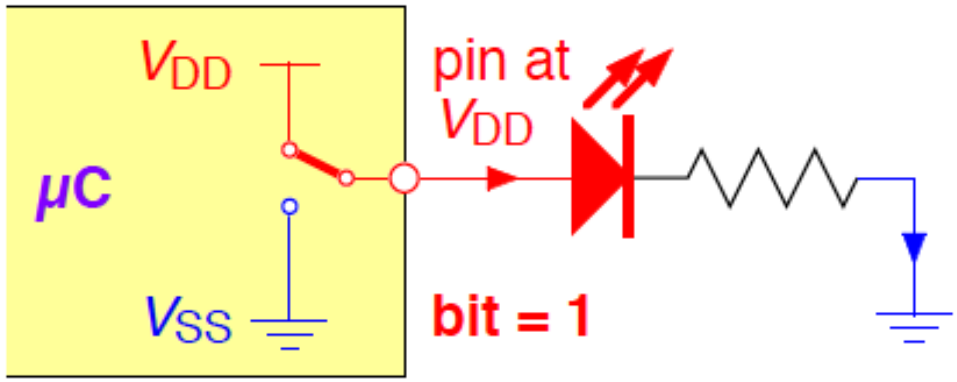
Note that the LED is the other way around!

Remember how the output works for each bit of the port:

logical 0 causes the μC to drive the pin low, to V_{SS}, ground, negative

logical 1 causes the μC to drive the pin high, to V_{DD}, supply, positive





Both ends of the LED are at V_{SS} so no current flows

The anode of the LED is at V_{DD} and the cathode is at V_{SS}.

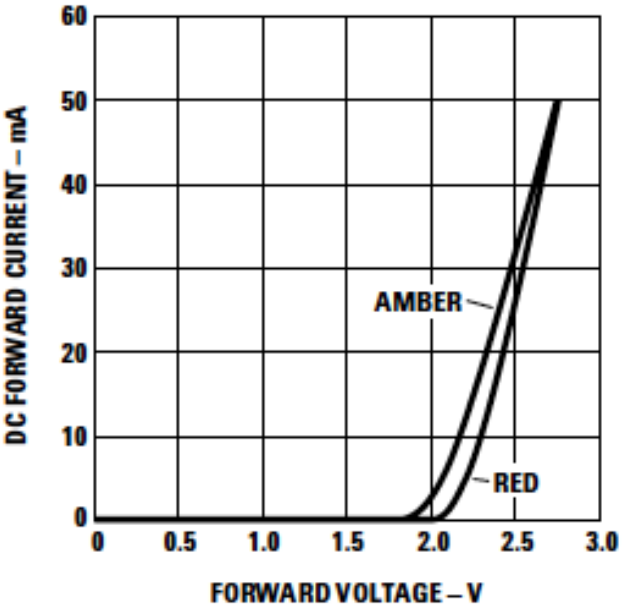
A forward current can flow (in the direction of the arrows) so the LED illuminates

(What would happen if the LED were connected the other way around?)

May wish to control brightness or reduce current

- In this case, add series resistance between LED and mbed.

- Assume that to increase battery lifetime, want to limit current through red LED connected to mbed DigitalOut pin to 5 mA.
  What value of resistor should be used ?



Say V from MCU is 3.3V

V drop across diode is 2.2V

V remaining is 3.3-2.2 = 1.1V

V=IR so

1.1 = 0.005 x R

1.1/0.005 = **220 Ohms**

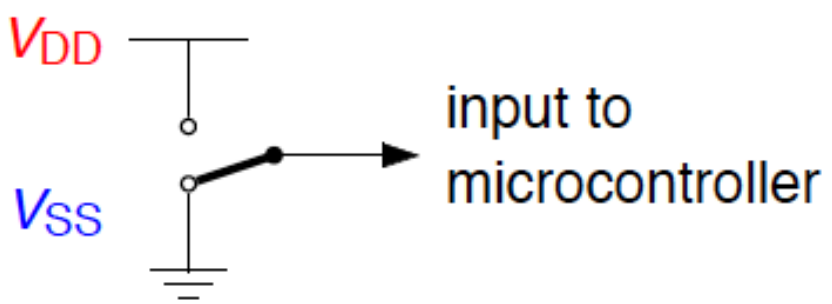
Digital In

- Connecting switches to the mbed

Suppose that we want to connect a simple on–off switch or pushbutton as an input to a μC. How should this be done?

The input should be either:

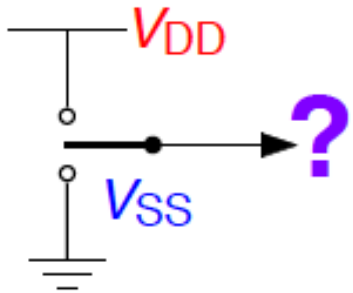
- V_{DD} (supply, positive) for logic 1
- V_{SS} (ground, negative) for logic 0



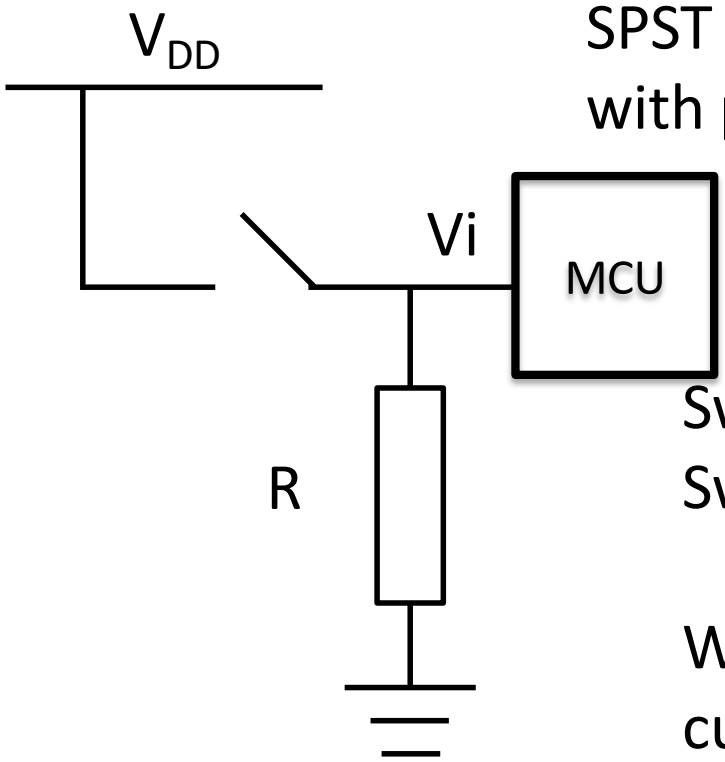
This may be the obvious approach:

However, this is not used in practice, for several reasons:

- it needs an expensive switch (3 terminals instead of 2, not just a simple pushbutton!)
- it needs 2 or 3 wires
- **what happens while the switch is being operated?**



What is actually done in practice



SPST (Single Pole Single Throw) Switch with **pull down resistor**

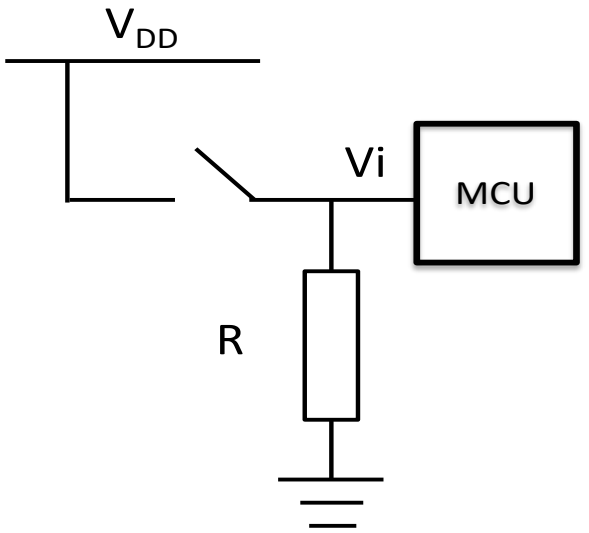
Switch open – Vi is connected to ground
Switch closed – Vi is connected to Vss

When switch is closed, there is a wasted current flowing through resistor R, so resistor is usually large

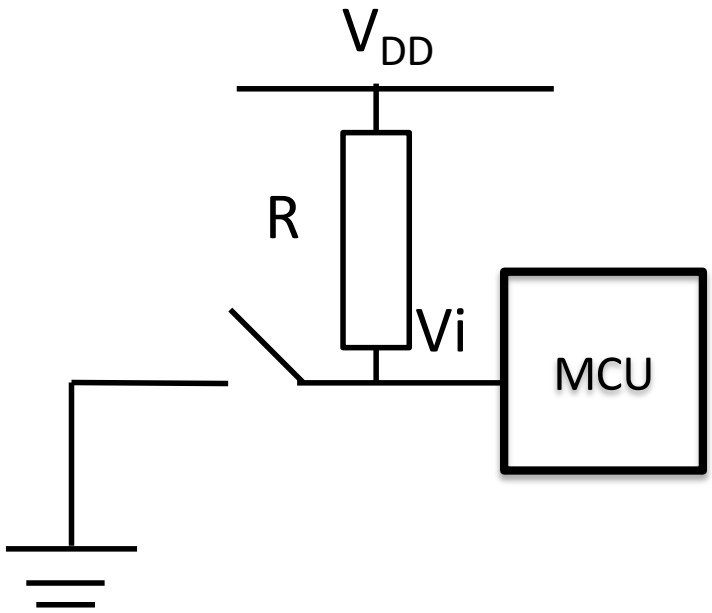
On the mbed, the resistor is available within the microcontroller – the default when a pin is configured as DigitalIn, is the pull-down configuration, so the default of an input pin is that it will be at 0 V. (this can be changed via the "mode" function of the DigitalIn command).

Pull down vs Pull up resistor

- Pull down resistor



- Pull up resistor



How much current is wasted?

A typical supply voltage is 3 V and pullup resistors are often 10 kΩ. How much current is wasted through the pullup when the button is pressed? For comparison, the microcontroller draws about 4 mA in normal operation.

Ohm's law tells us that current I, voltage V and resistance R are related by

$$I = \frac{V}{R} = \frac{3\,\mathrm{V}}{10\,\mathrm{k}\Omega} = \frac{3\,\mathrm{V}}{10^4\,\Omega} = 3 \times 10^{-4}\,\mathrm{A} = 0.3\,\mathrm{mA} = 300\,\mu\mathrm{A}$$

The normal operating current of the microcontroller is about 4 mA, so the pullup draws nearly a tenth of this. This is why larger resistors are now used.

Of course, the current flows only while the button is pressed.

DEMO – DIGITAL IN / OUT

```
/*Program to flash 1 of 2 LEDS, depending on the state of a 2 way switch

*/
#include "mbed.h"
DigitalOut redled(p21);
DigitalOut greenled(p22);
DigitalIn switchinput(p23);
int main() {
while(1) {
      if(switchinput==1) {    //test value of switch input

      //execute following block of code if switch input is 1
            greenled = 0;     //green LED is off

            redled = 1;       //flash red LED
            wait(1.0);

            redled = 0;
            wait(1.0);

      }                       //end of if

      else {

      //execute this block of code if switchinput is 0
      redled = 0;             //redled is off

      greenled = 1;           //flash green LED

      wait(1.0);
      greenled = 0;
      wait(1.0);

      }                       //end of else

}                             //end of while(1)

}                             //end of main
```