Mechatronics (ROB-GY 5103 Section A)

- **Today's lecture:**
  BS2 Programming, Hardware, & Interfacing
- (See Topics #2 from Main Text for details)

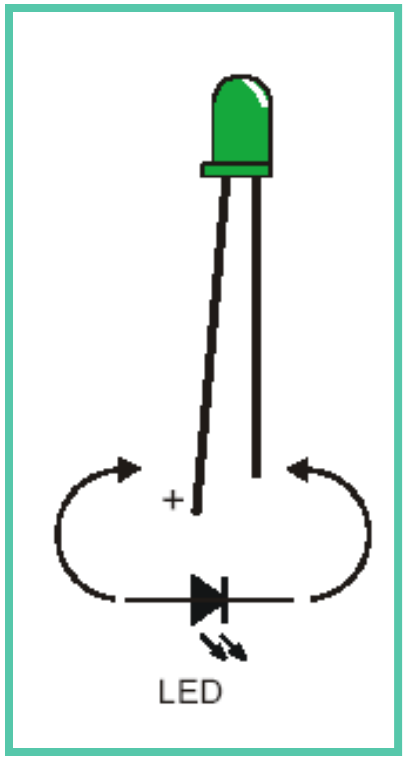
MakerSpace Reminder

- https://makerspace.engineering.nyu.edu/

Updated Project Guidelines

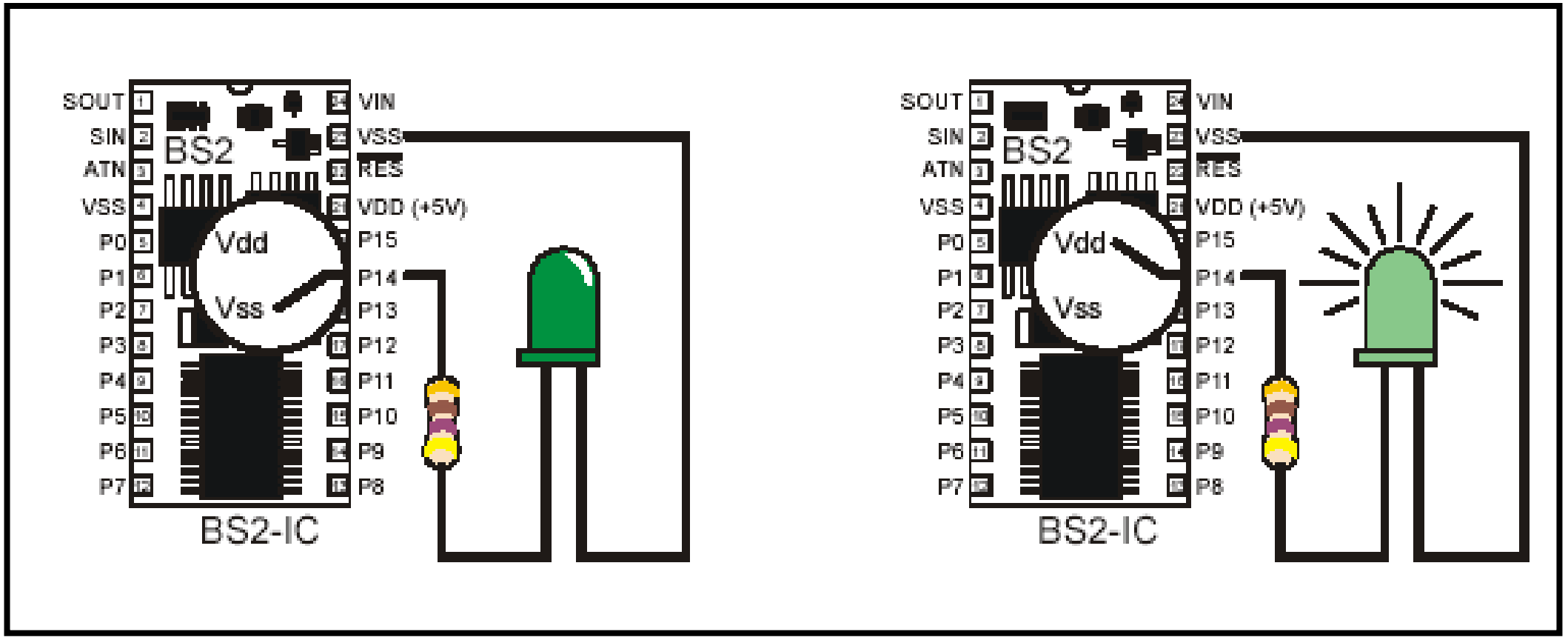
How to light up an LED

- Diode allows current to flow only one way
- **Light Emitting Diode (LED)** is a diode that emits light as current is passed through it.
- **Anode:** Connected to **+** side of voltage.
    - **ID: longer lead**
- **Cathode:** Connected to **–** side of voltage.
    - **ID: shorter lead and a flat portion on the lens**
- Conventional current flows into the anode and exits the cathode!



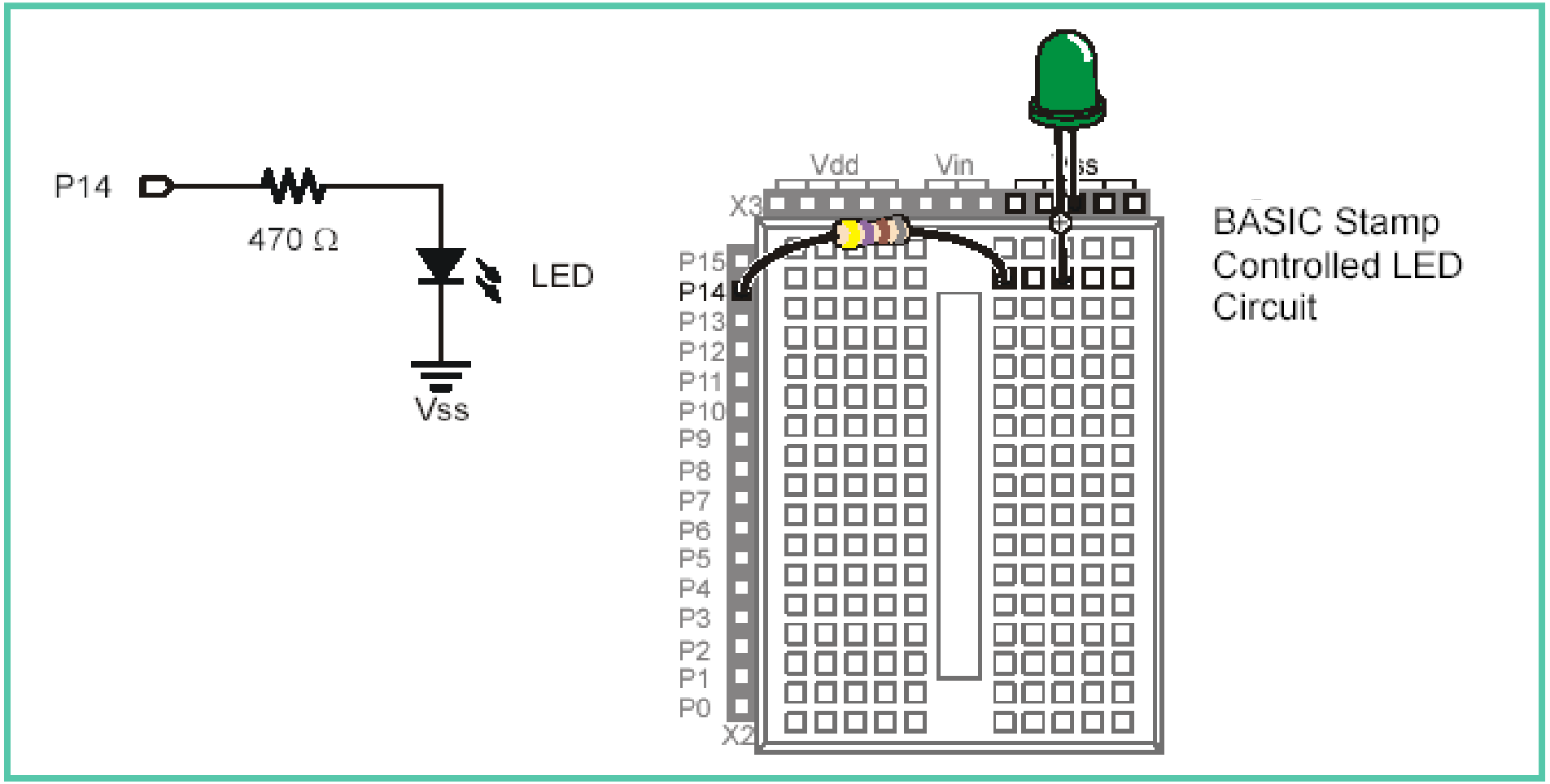
How to light up an LED with BS2

- Vss = 0V, Vdd = 5V, and pins P0–15 can be set to output either (high/low or Vdd/Vss)



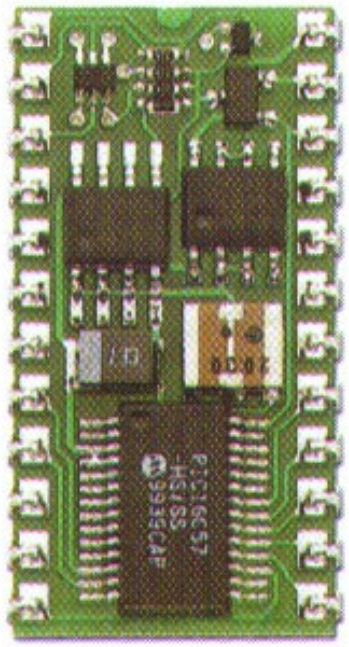
How to light up an LED with BS2

- Vss = 0V, Vdd = 5V, and pins P0–15 can be set to output either (high/low or Vdd/Vss)



Things that can go wrong

- I/O pins can **source** 20 mA and **sink** 25 mA



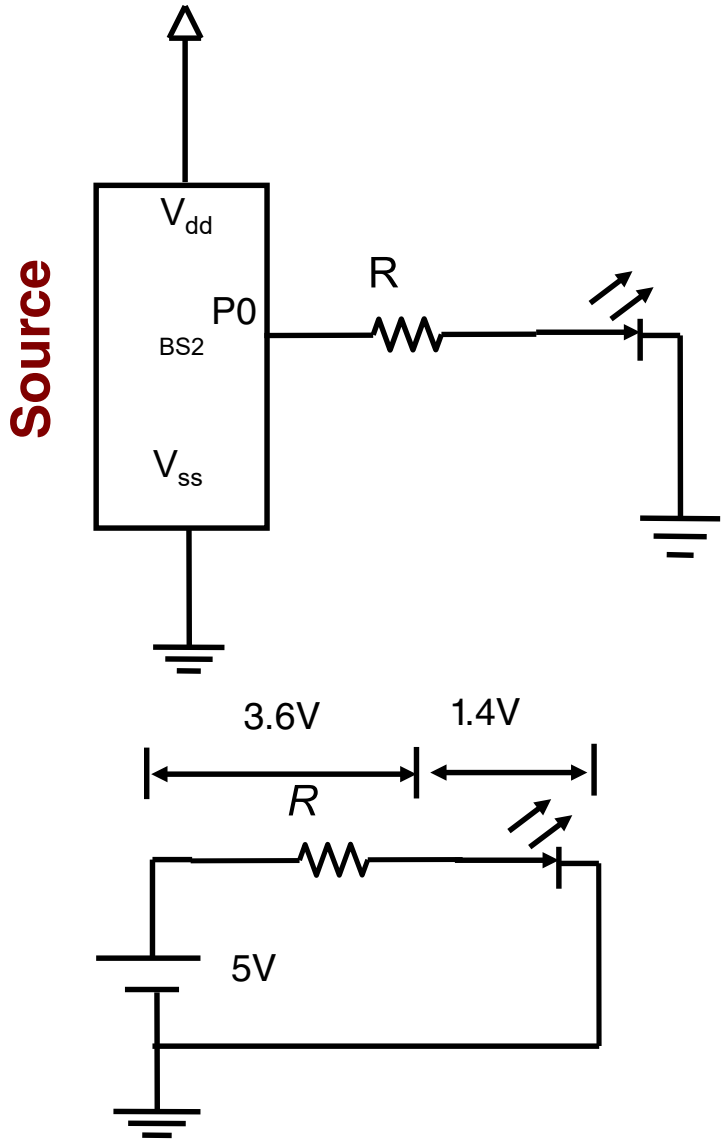
BASIC Stamp 2

Package:	24-pin DIP
Environment:	0°-70° C (32 -158° F)
Microcontroller:	Microchip PIC16C57
Processor Speed:	20 MHz
Program Execution Speed:	~4,000 instructions/sec
RAM Size:	32 Bytes (6 I/O,26 Variable)
EEPROM (Program) Size:	2K Bytes,~500 instructions
Number of I/O Pins:	16 + 2 Dedicated Serial
Voltage Requirements:	5 - 15 vdc
Current Draw @ 5V:	8 mA Run / 100 µA Sleep
Source/Sink Current per I/O:	20 mA / 25 mA
Source/Sink Current per unit:	40 mA / 50 mA per 8 I/O pins
PBASIC Commands:	36
PC Programming Interface:	Serial Port (9600 baud)
DOS Text Editor:	STAMP2.EXE
Windows Text Editor:	Stampw.exe

LED Math

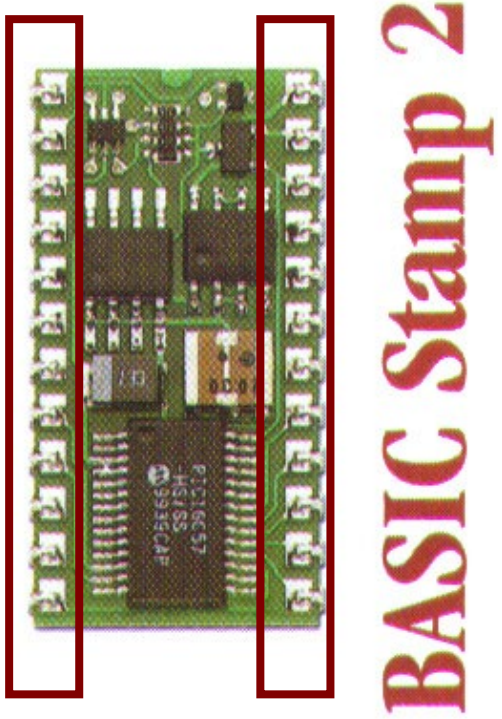
- LED requires forward voltage **1.4 V** and source current of **10 mA** from BS2
- Pin P0 is driven high (to Vdd = 5V).
- **Voltage divider circuit:**
  - The LED and current-limiting resistor divide the voltage
- Choose resistance *R* to protect I/O pin of the BS2 and the LED

$$R = \frac{V}{I} = \frac{3.6}{10 \times 10^{-3}} = 360\Omega$$



Can you drive 8 LEDs with BS2?

- Each row of 8 I/O pins can **together source** 40 mA and **sink** 50 mA.



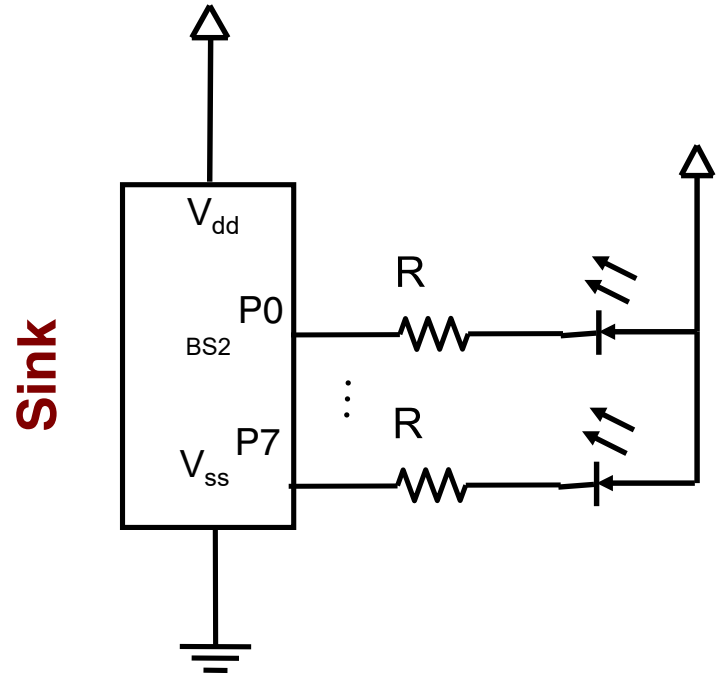
BASIC Stamp 2

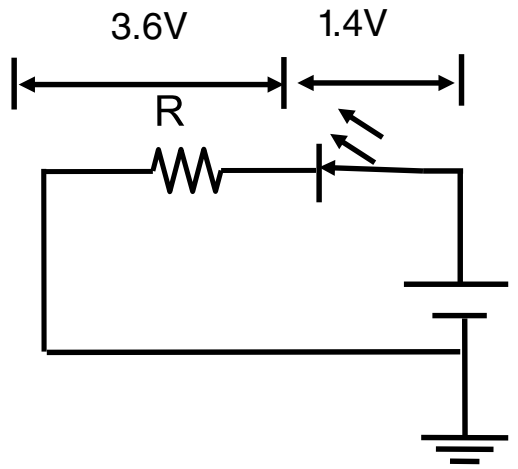
Package:	24-pin DIP
Environment:	0°-70° C (32 -158° F)
Microcontroller:	Microchip PIC16C57
Processor Speed:	20 MHz
Program Execution Speed:	~4,000 instructions/sec
RAM Size:	32 Bytes (6 I/O,26 Variable)
EEPROM (Program) Size:	2K Bytes,~500 instructions
Number of I/O Pins:	16 + 2 Dedicated Serial
Voltage Requirements:	5 - 15 vdc
Current Draw @ 5V:	8 mA Run / 100 µA Sleep
Source/Sink Current per I/O:	20 mA / 25 mA
Source/Sink Current per unit:	40 mA / 50 mA per 8 I/O pins
PBASIC Commands:	36
PC Programming Interface:	Serial Port (9600 baud)
DOS Text Editor:	STAMP2.EXE
Windows Text Editor:	Stampw.exe

LED Math

- Each row of 8 I/O pins can **together source** 40 mA and **sink** 50 mA.
- Each LED is allowed to sink 50 mA/8 = **6.25mA**. A bit dim but will work.
- Find *R* as before:

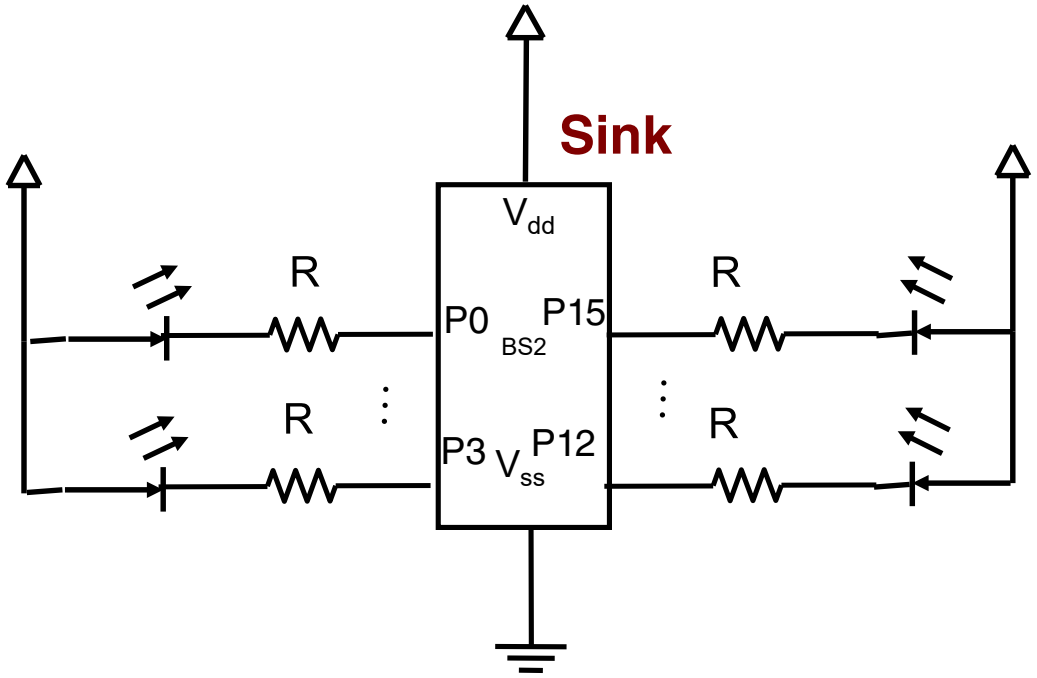
$$R = \frac{V}{I} = \frac{3.6}{6.25 \times 10^{-3}} = 576\Omega$$





Can you drive 8 LEDs with BS2?

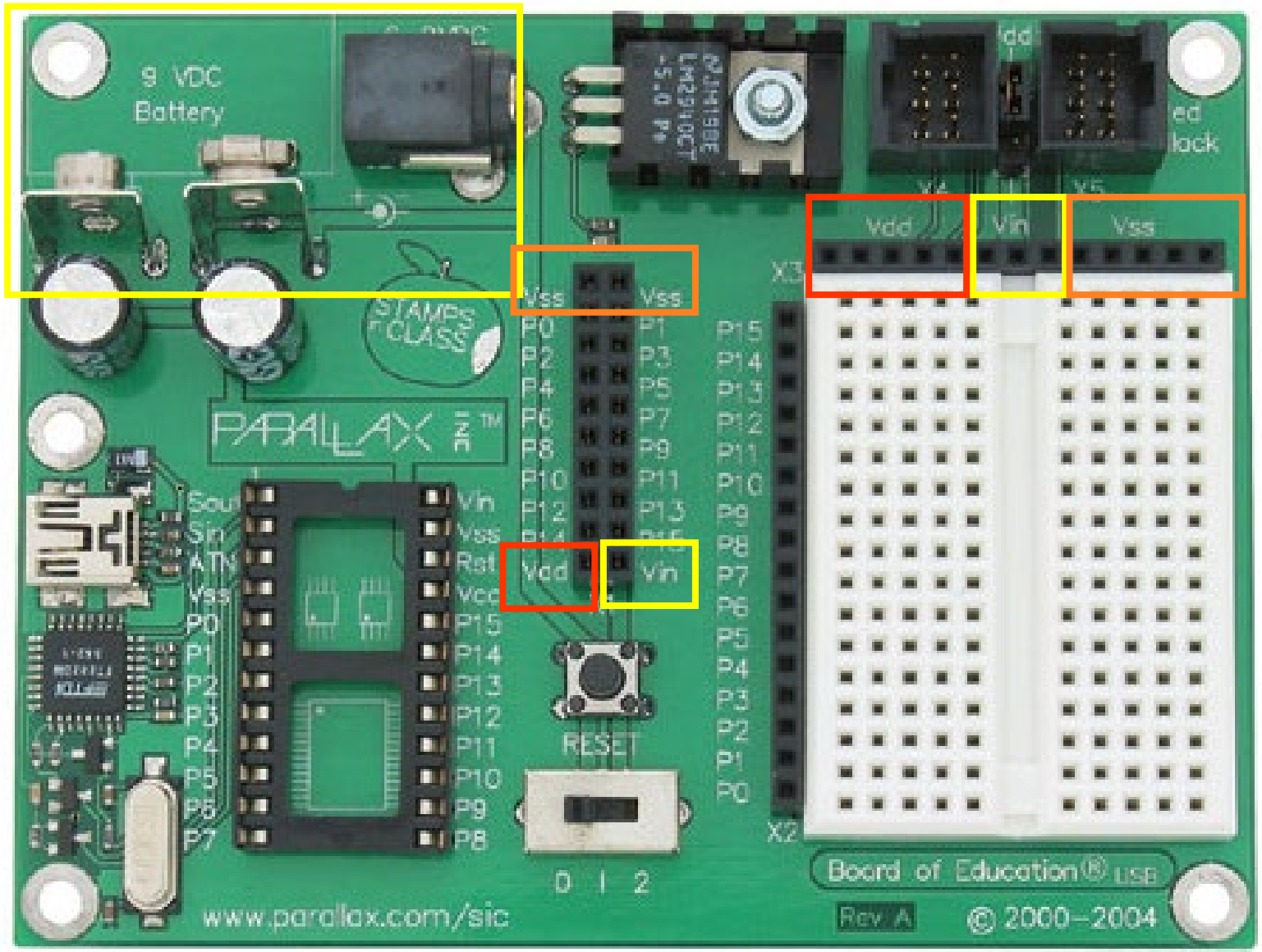
- Each group of 8 I/O pins can **together** **source** 40 mA and **sink** 50 mA.
- Or use both rows!



Regulation

- A lot hinges on the fact that we can expect 5V!
- Just like you expect that you get close to 110-120V when you plug into a wall (in the United States)

Board of Education: Power Connections



+5V (Vdd)
Regulated by Board of Education

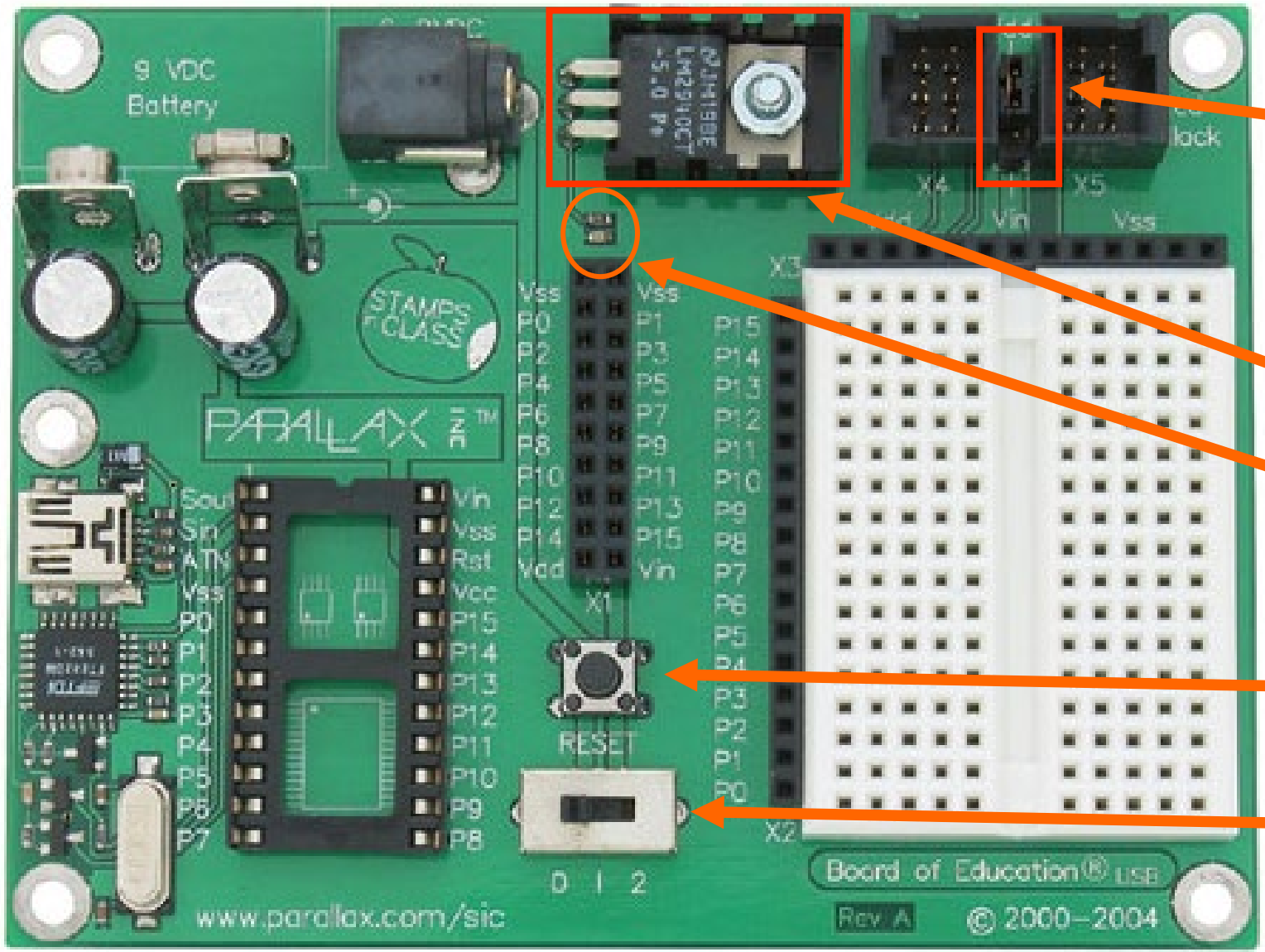
0V or ground (Vss)

Supply Voltage (Vin)
Unregulated by Board of Education, directly from Power Supply

NOTE: Avoid using Vin. May cause damaging voltages to be applied to the BASIC Stamp.

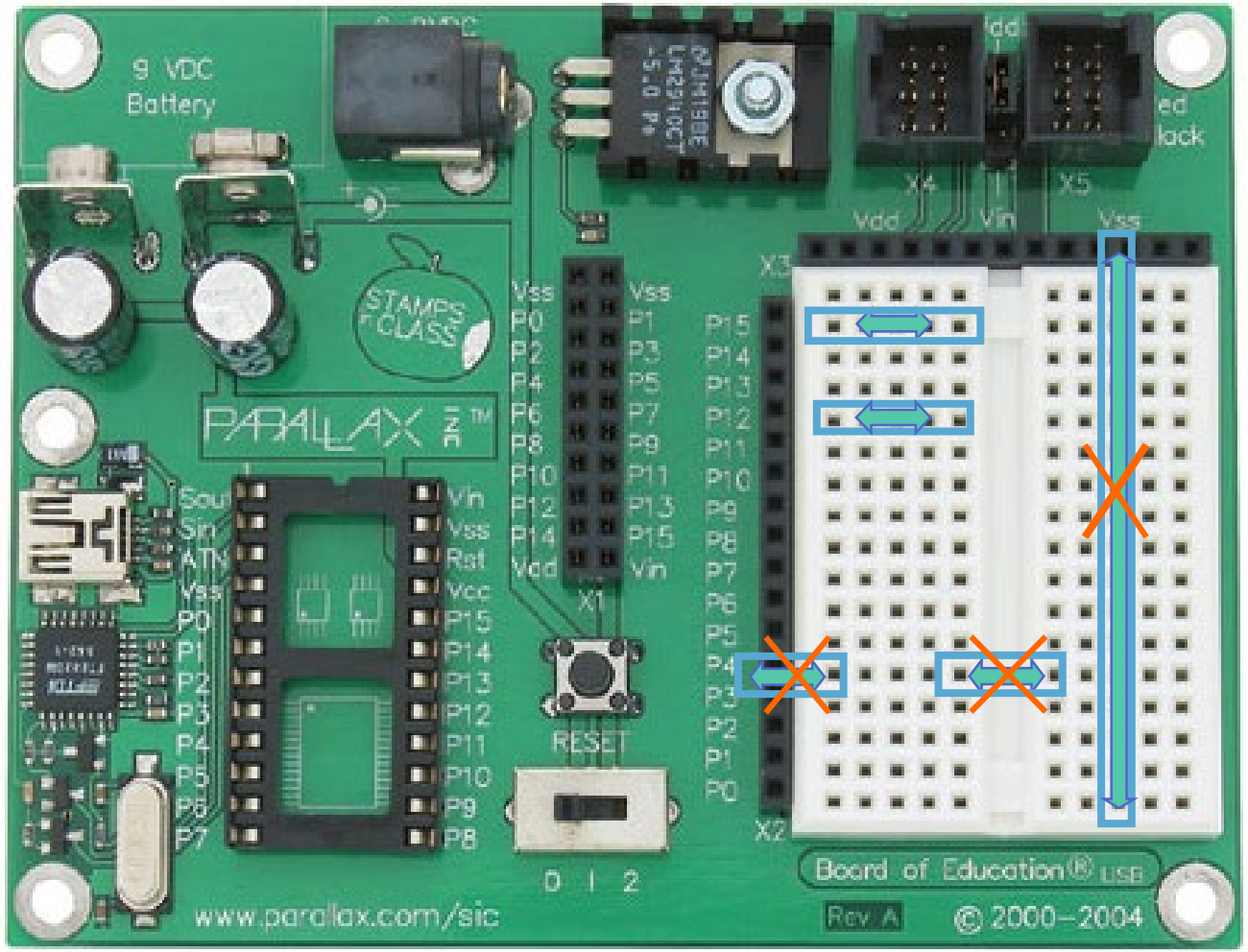
Use only under directed use!

Board of Education: Power Connections



- Jumper setting: Decides whether the red wire should be “jumped” or connected to Vin or Vdd.
- Regulator
- Power-on LED
- Reset button
- Three-position power switch

Board of Education: Breadboard Connections



Each row in each half of the breadboard are electrically the same point.

There exist no connections between the headers and the breadboards or in columns on the breadboard.

Components are connected between rows and to the headers to make electrical connections.

Components should NOT be connected on a single row or they will be shorted out of the circuit.

BS2: Power Connections

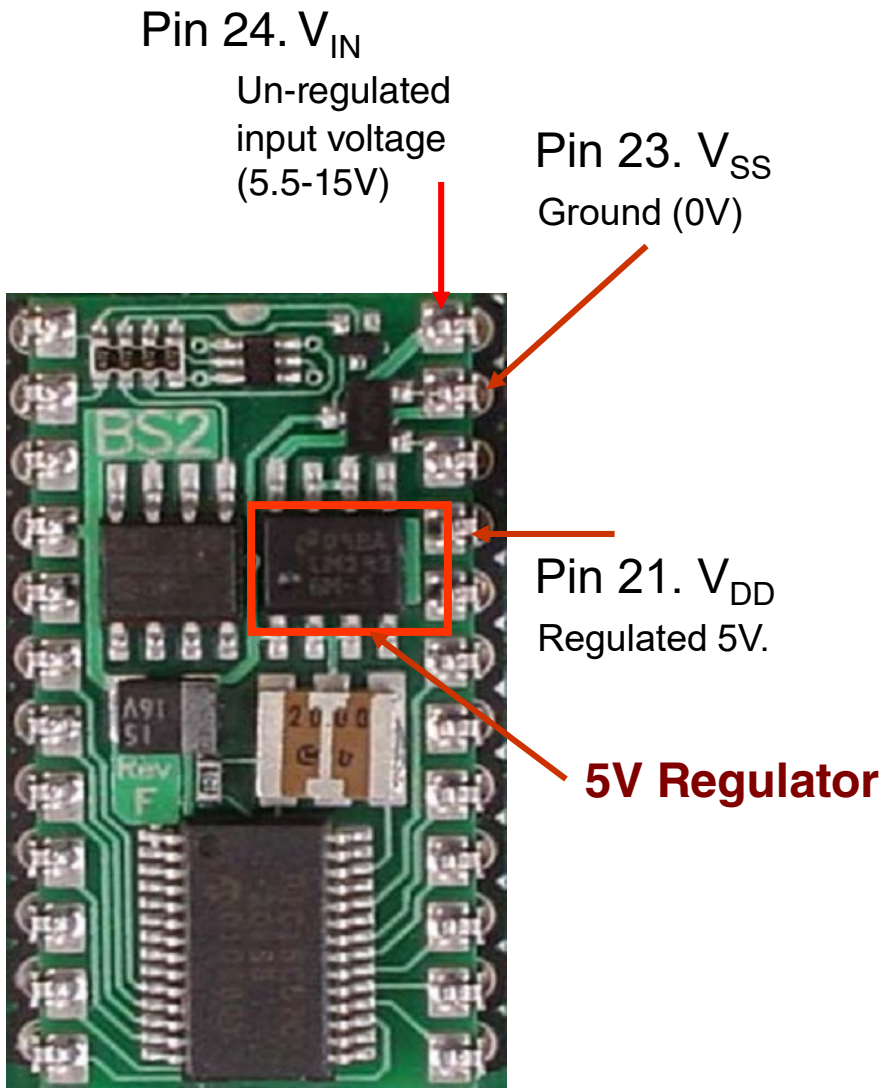
- BS2 requires 5 VDC power supply and approximately 8mA.
- Pin#23 (Vss) always connected to ground/negative terminal of power supply

THEN EITHER

- Pin#21 (Vdd) can be connected to regulated 5VDC external power supply

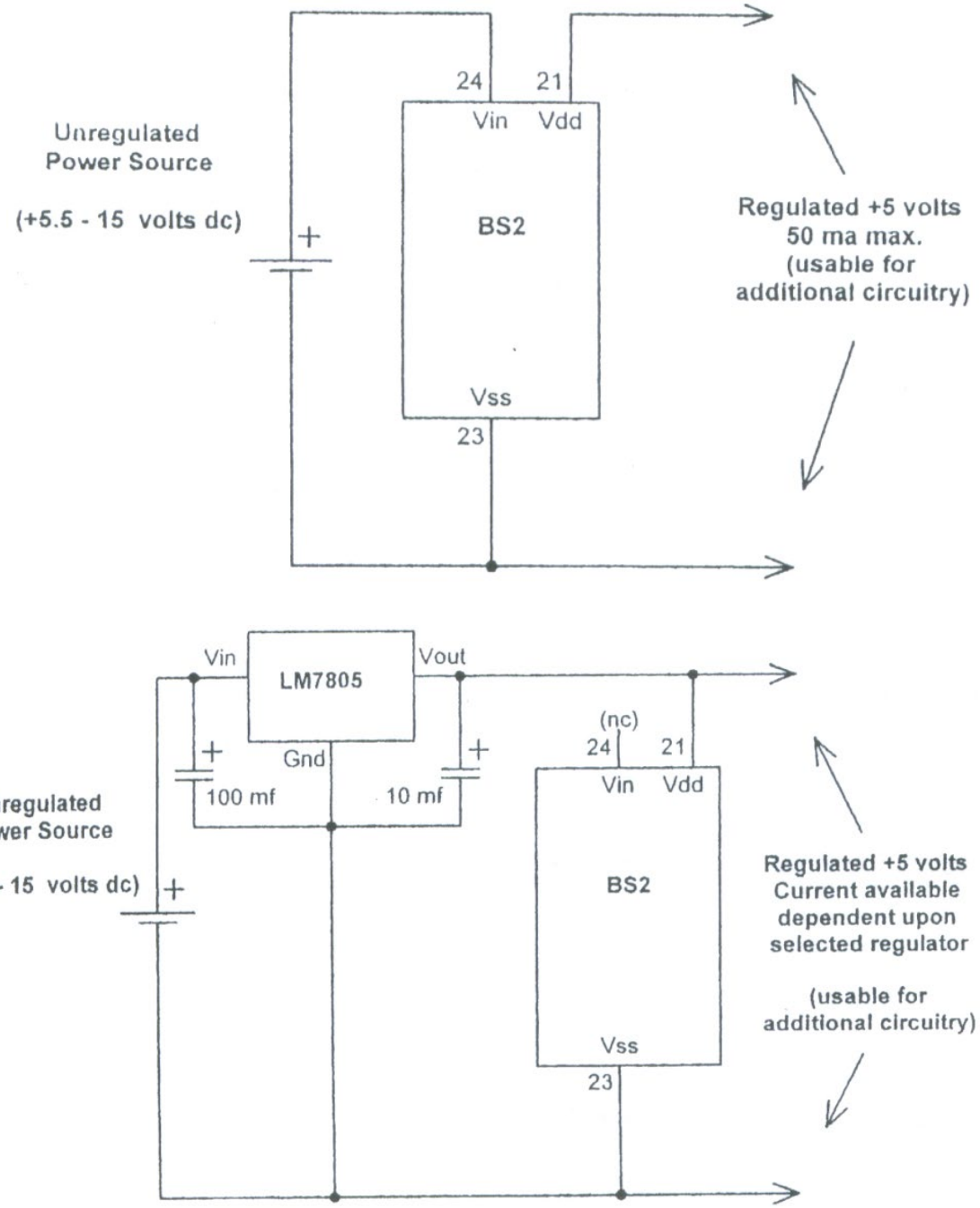
OR

- Pin#24 (V_{IN}) of BS2 can be connected to positive terminal of 5.5-12 V power supply. (BS2 has its own 5V regulator)



Power Connections Matter!

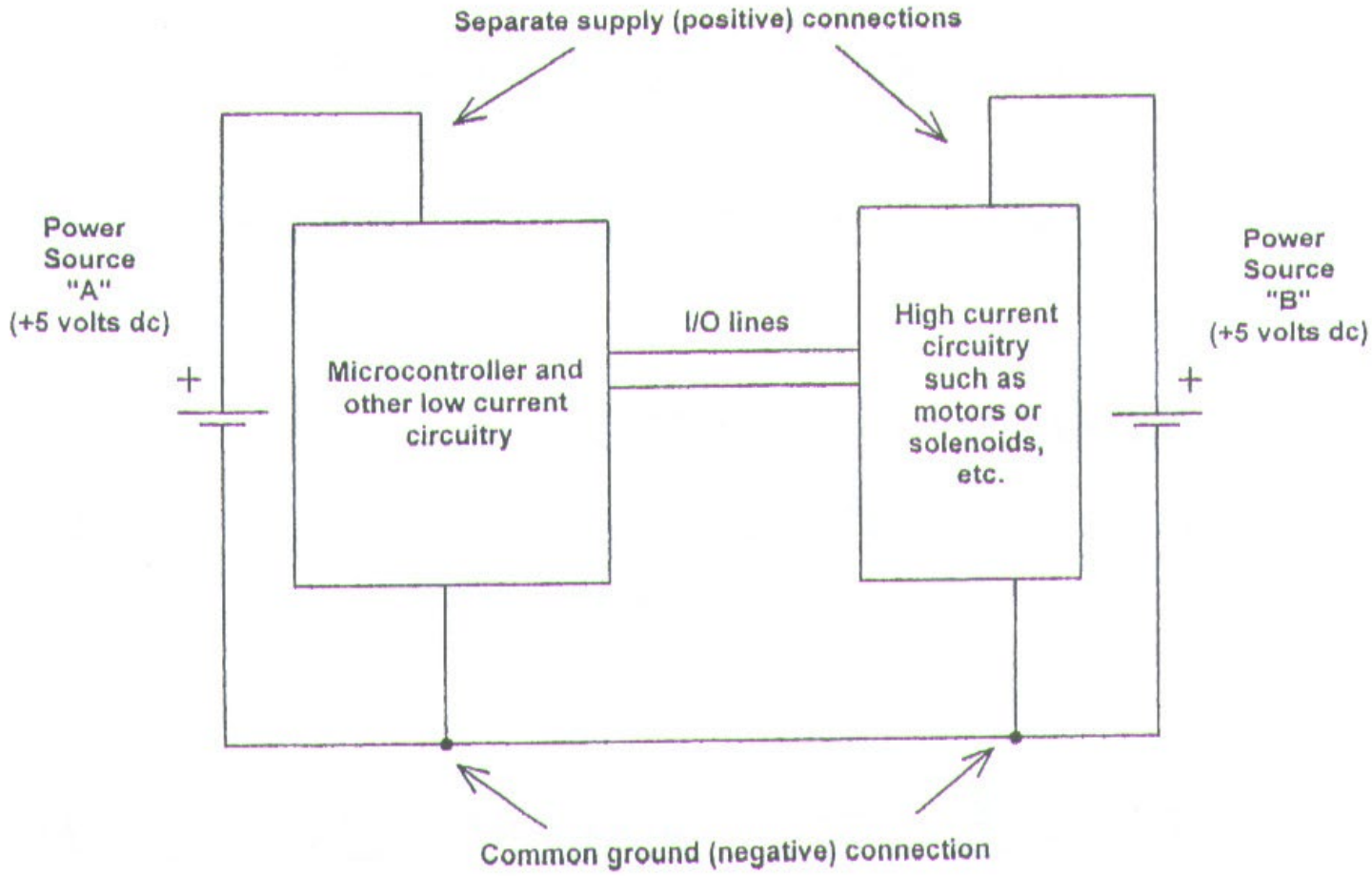
- If BS2's on-board voltage regulator is used to power external circuits, **ALL I/O pins** as a group can
  - source up to 40mA current
  - sink up to 50mA current
- When an external voltage supply with steady 5VDC is used to power BS2 and this external supply is capable of delivering at least 100mA, **then each group of 8 I/O pins (P0-P7 and P8-P15)** can
  - source up to 40mA current
  - sink up to 50mA current



Power Connections Matter!

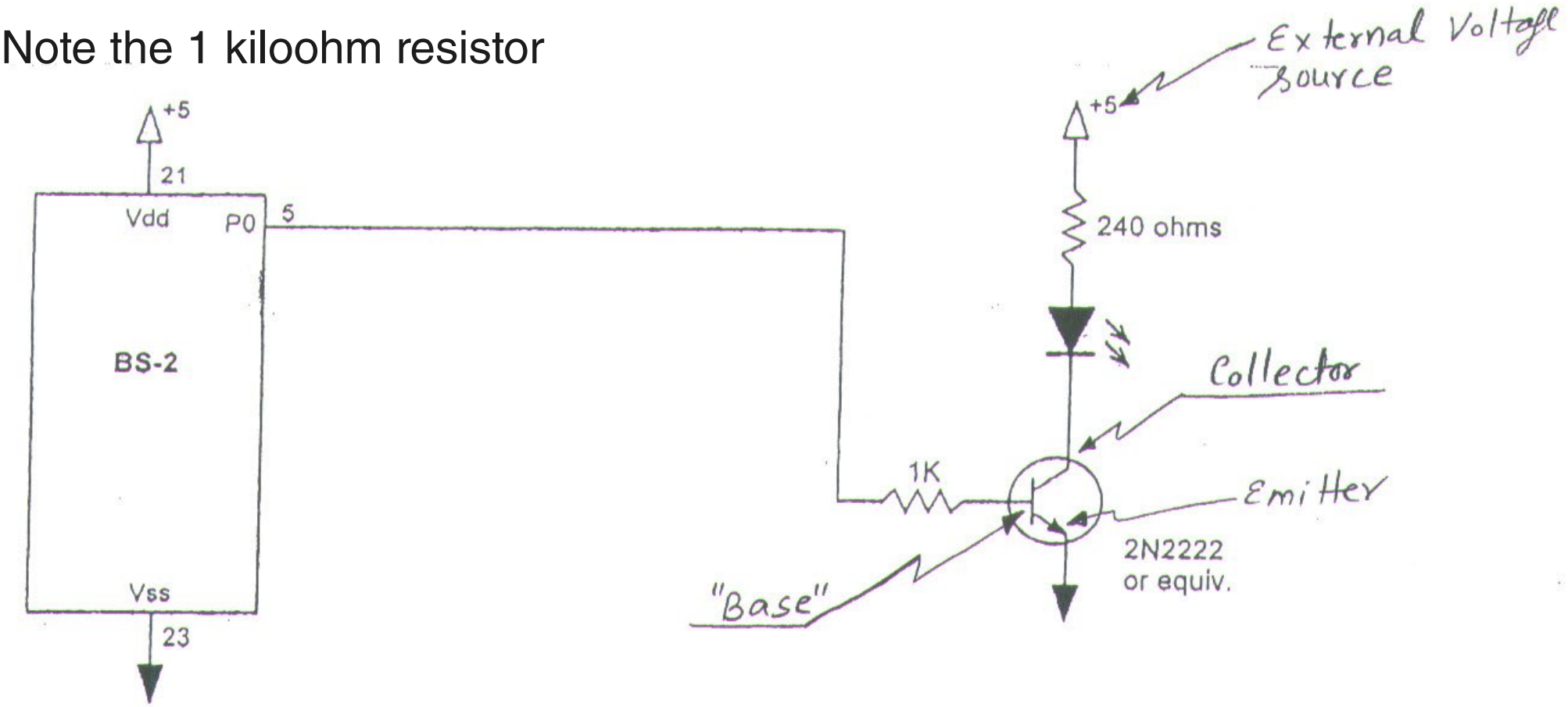
- Devices such as solenoids, relays, motors, etc., require current beyond BS2's capability.
  - **Solution:** Use a separate power supply
- When using one power supply for BS2 and another for high current load, both supplies must be properly isolated. **Common ground but uncommon positive lines.**

Positive Terminal Isolation



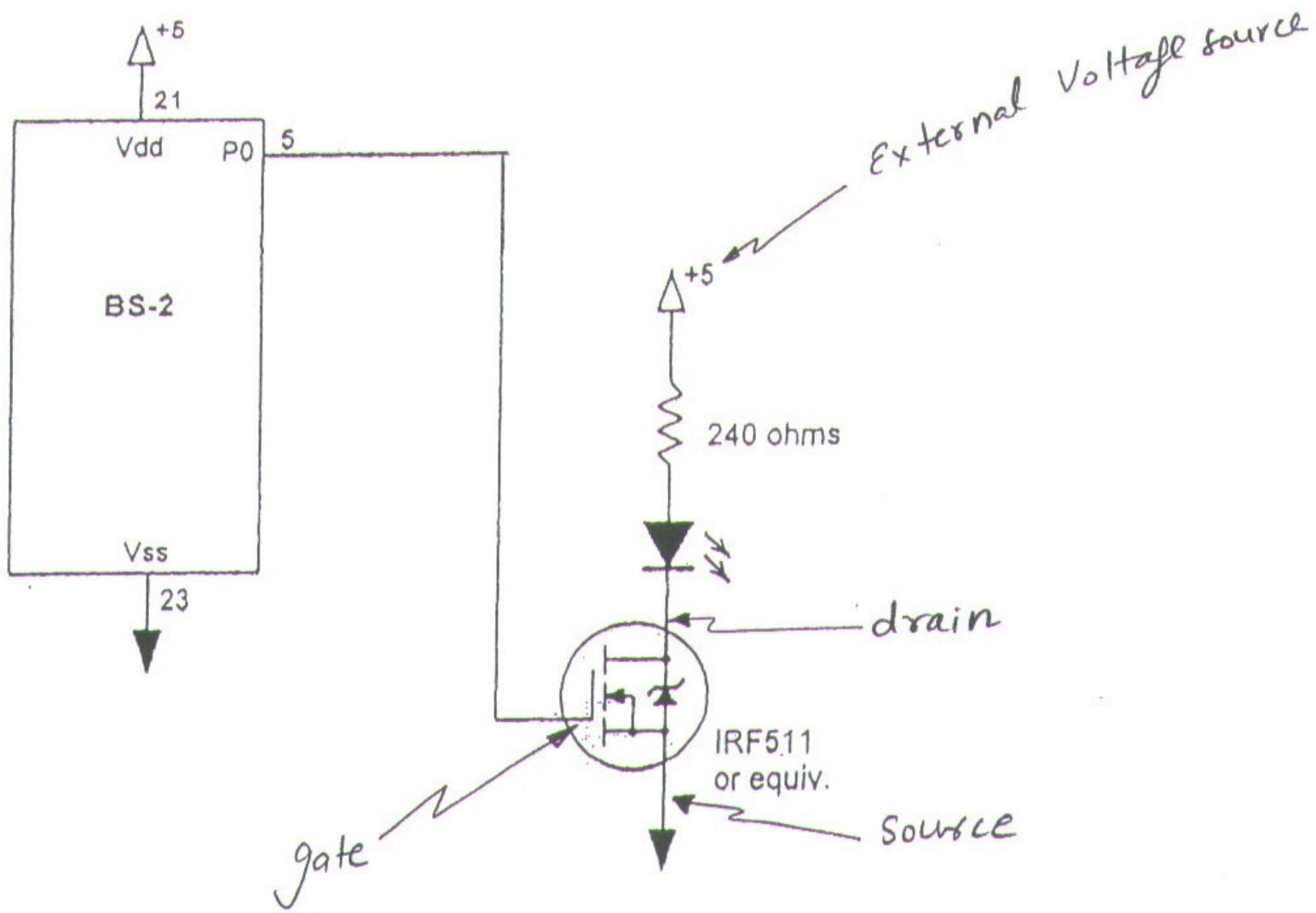
Buffering Circuit

- Applying a positive voltage to the **base** of the NPN transistor allows current to flow
- Note the 1 kiloohm resistor



Buffering Circuit

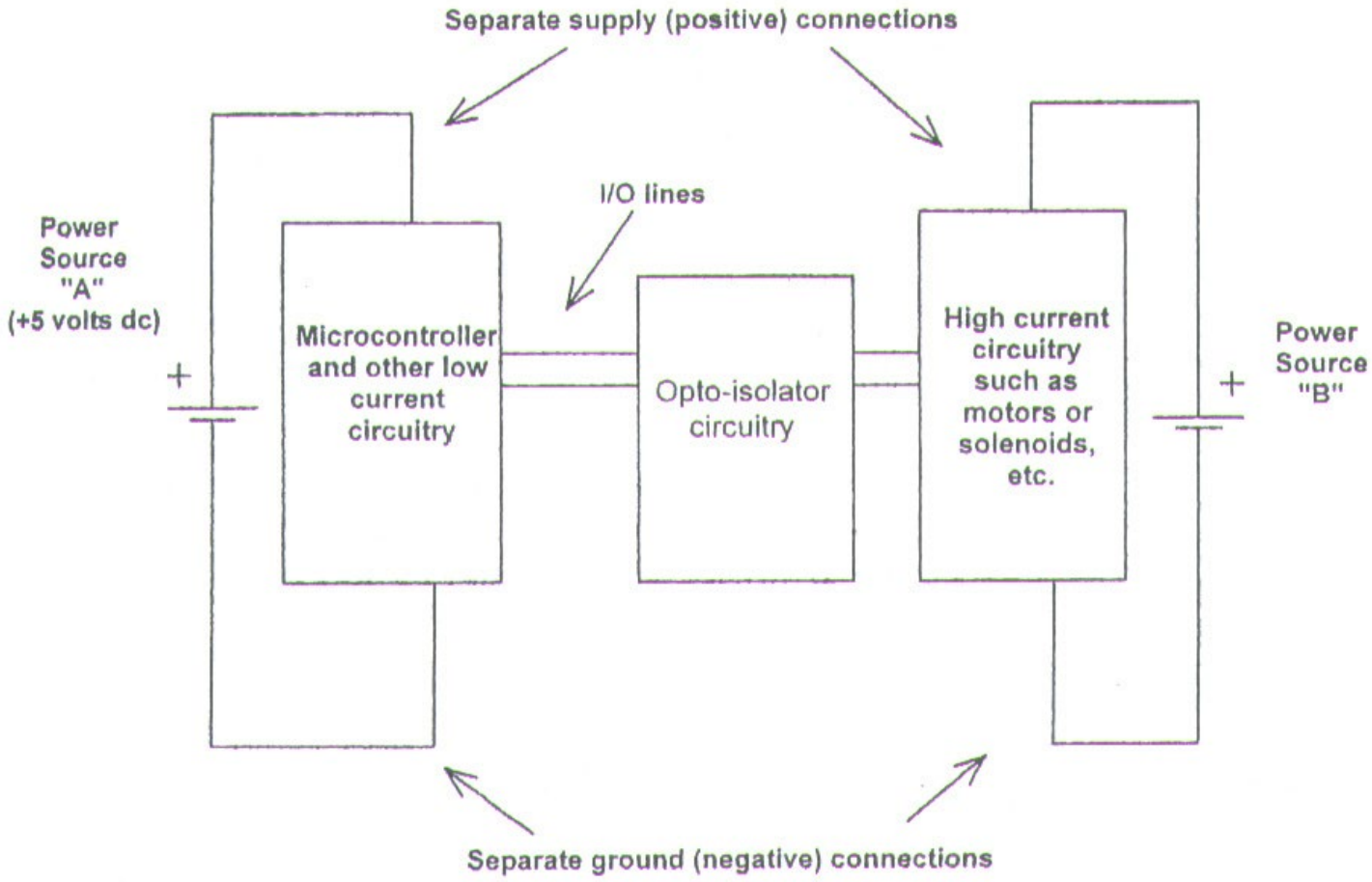
- Applying a positive voltage to the **gate** of the “n-channel” enhancement MOSFET allows current to flow
- MOSFETs are directly controlled by gate voltage while BJT output current depends on the base current which depends on the base voltage



Complete Isolation

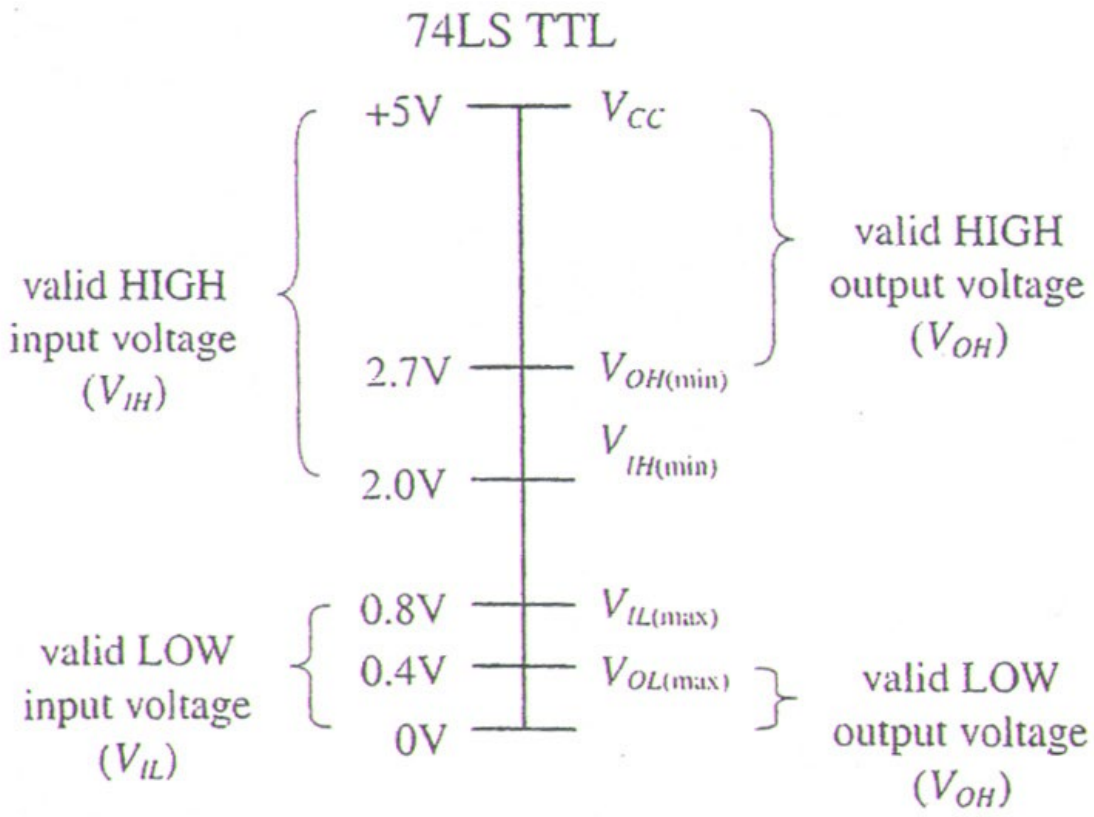
- Some application may necessitate complete isolation of sensitive BS2 components from high current loads. **Uncommon ground AND uncommon positive lines.**
- Possible with **opto-isolators**, which convert between light and electrical signals.

Complete Isolation

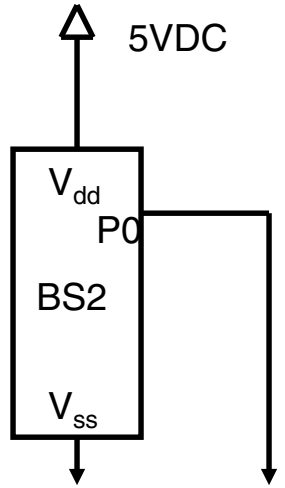


Voltage

- For many situations, voltage is the main signal and the current is the “result”
  - A pin communicates with an external circuit to “speak” one/zero or high/low via 5V/0V or Vdd/Vss.
    - Some current flows through the pin, but the current amount (amperage) is not that important unless it is too high and blows up the pin.
  - We will see this in many sensors, digital or analog.



More things that can go wrong



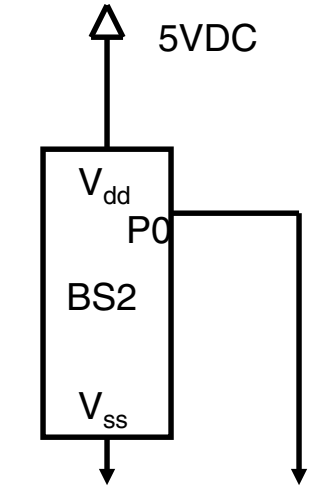
- P0 is an output pin that is mistakenly connected or "shorted" to ground.
  - When P0 is commanded to output high/Vdd/5V there is a **LARGE** current.
  - This is very **BAD**.
- Recall Ohm's Law V=IR with V=5V, R≅0Ω

$$I = \frac{V}{R} = \frac{5}{0} \approx \infty$$

More things that can go wrong

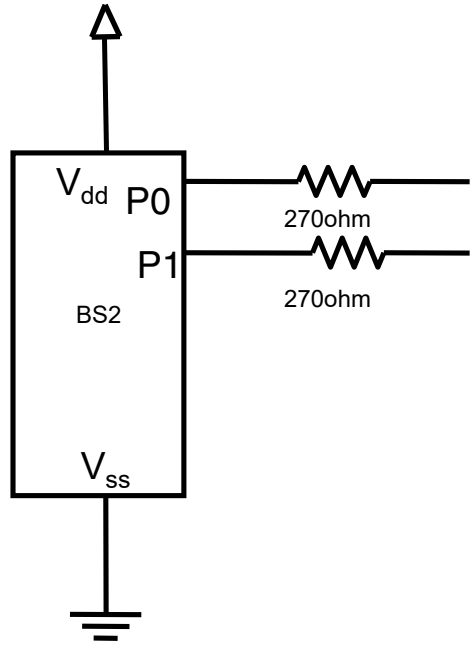
- P0 is an output pin that is mistakenly connected or "shorted" to ground.
  - When P0 is commanded to output high/Vdd/5V there is a **LARGE** current.
  - This is very **BAD**.
- Recall Ohm's Law V=IR with V=5V, R≅0Ω

$$I = \frac{V}{R} = \frac{5}{0} \approx \infty$$



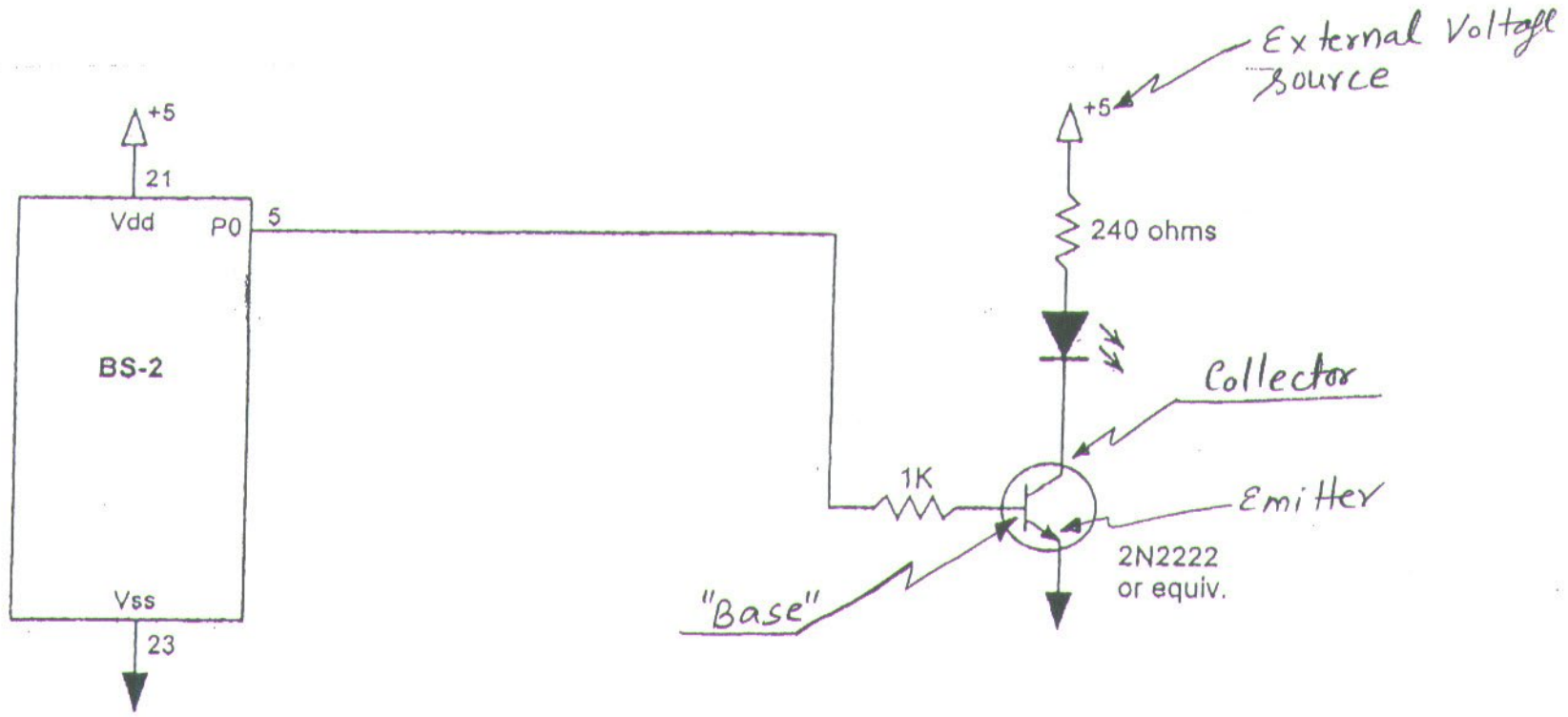
- Use a resistor R = 270Ω to connect P0 to external circuitry to protect I/O pins.

$$I = \frac{V}{R} = \frac{5}{270} \approx 0.019A = 19mA$$



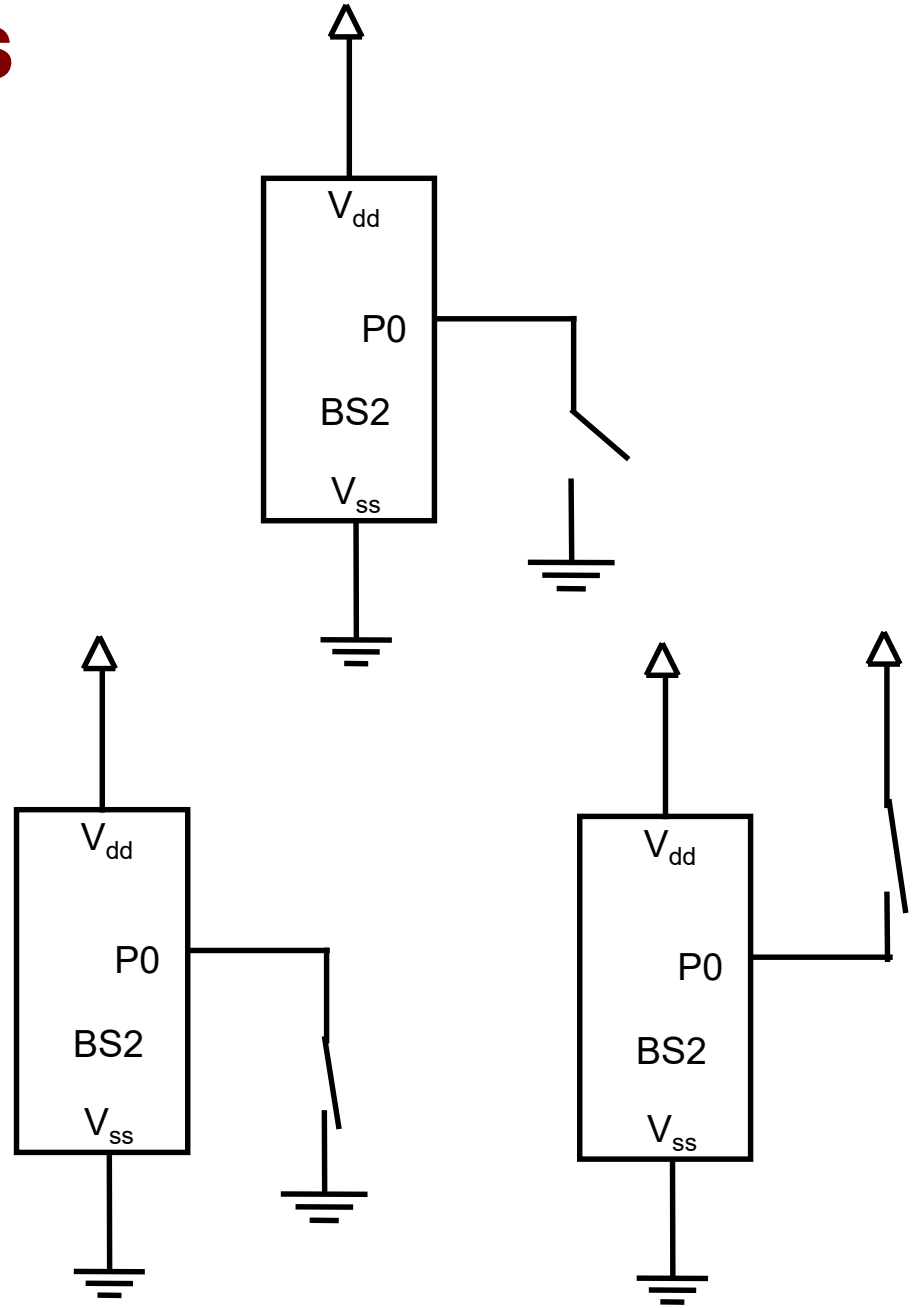
I/O Pin Protection

- We used a 1 kiloohm resistor to limit the base current through the NPN transistor in the previous example



Pullup and pulldown resistors

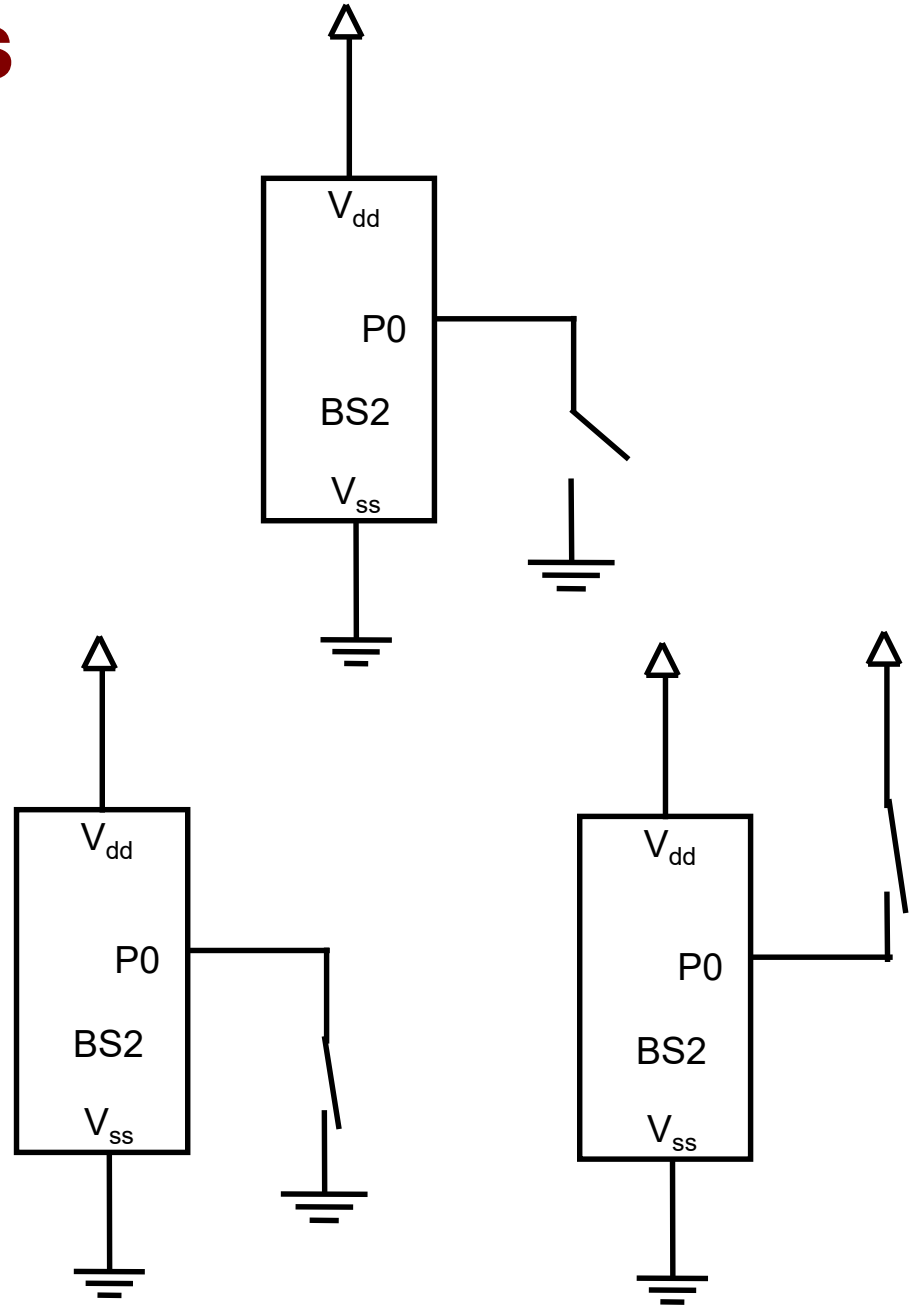
- Recall that voltage is how sensors communicate
- The most basic form of communication is pressing a switch.
  - Yes/no = switch closed/open.
  - Switch closed means it conducts (closed circuit)
  - Switch open means it does not conduct (open circuit)
- If is P0 is commanded to receive input, what is it reading in all these cases?



Pullup and pulldown resistors

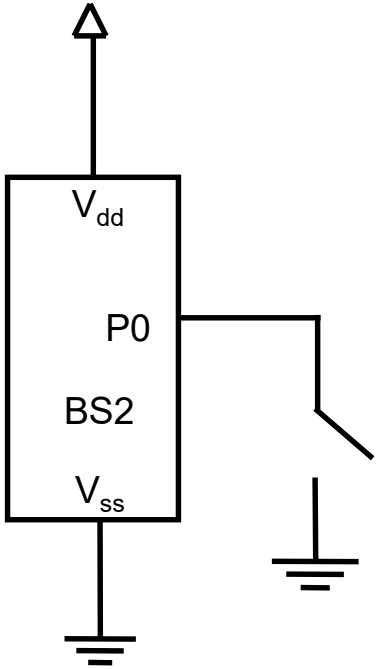
BAD. *Floating input*. Reads electric noise because it is not connected to anything. How to fix?

GOOD. Reads 0V/5V.

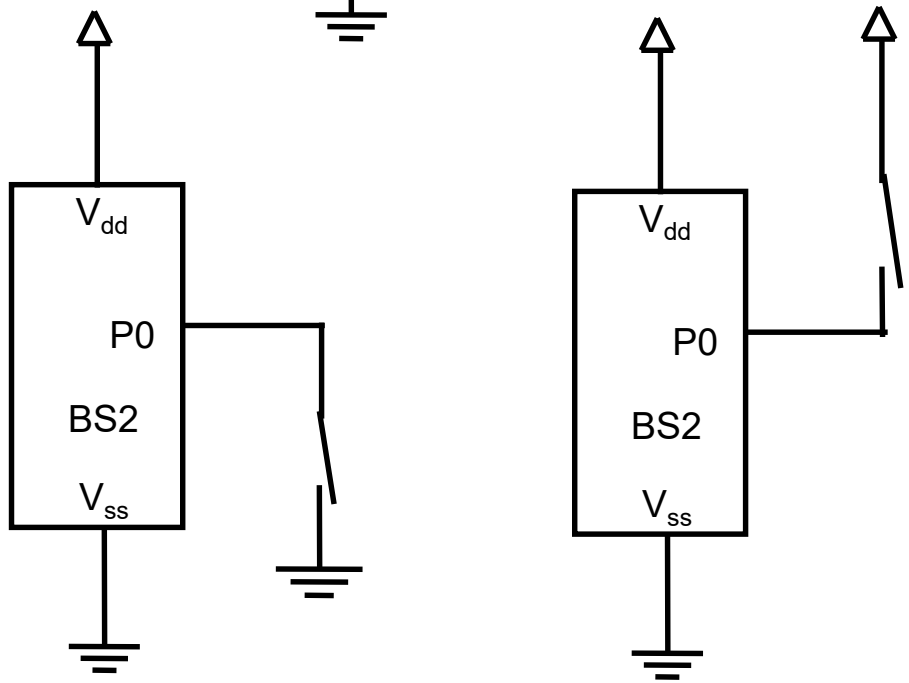


Pullup and pulldown resistors

BAD. *Floating input*. Reads electric noise because it is not connected to anything. How to fix?

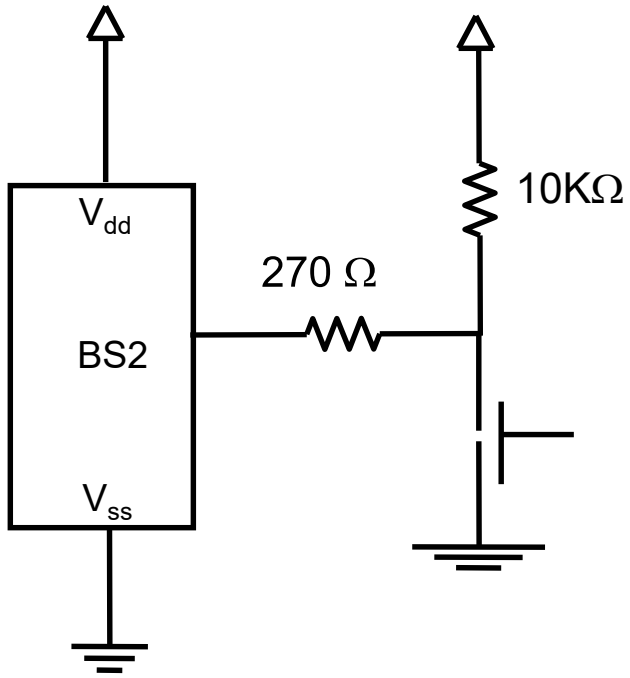


GOOD. Reads 0V/5V. A little unsafe in case we accidentally set P0 to be output. So let's add the 270 ohm resistor,

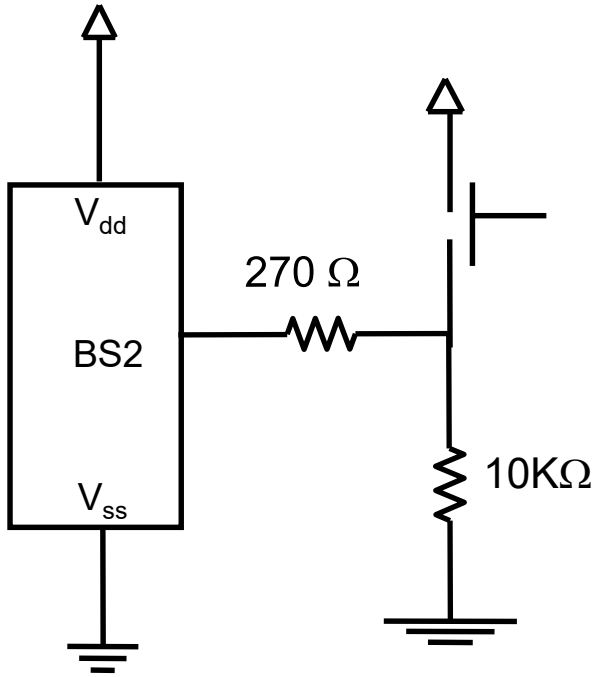


Pullup and pulldown resistors

- When button is not pressed, 10270Ω resistance pulls P0 high. 10K is pull up resistor.
  - Button not pressed is indicated by P0 high.



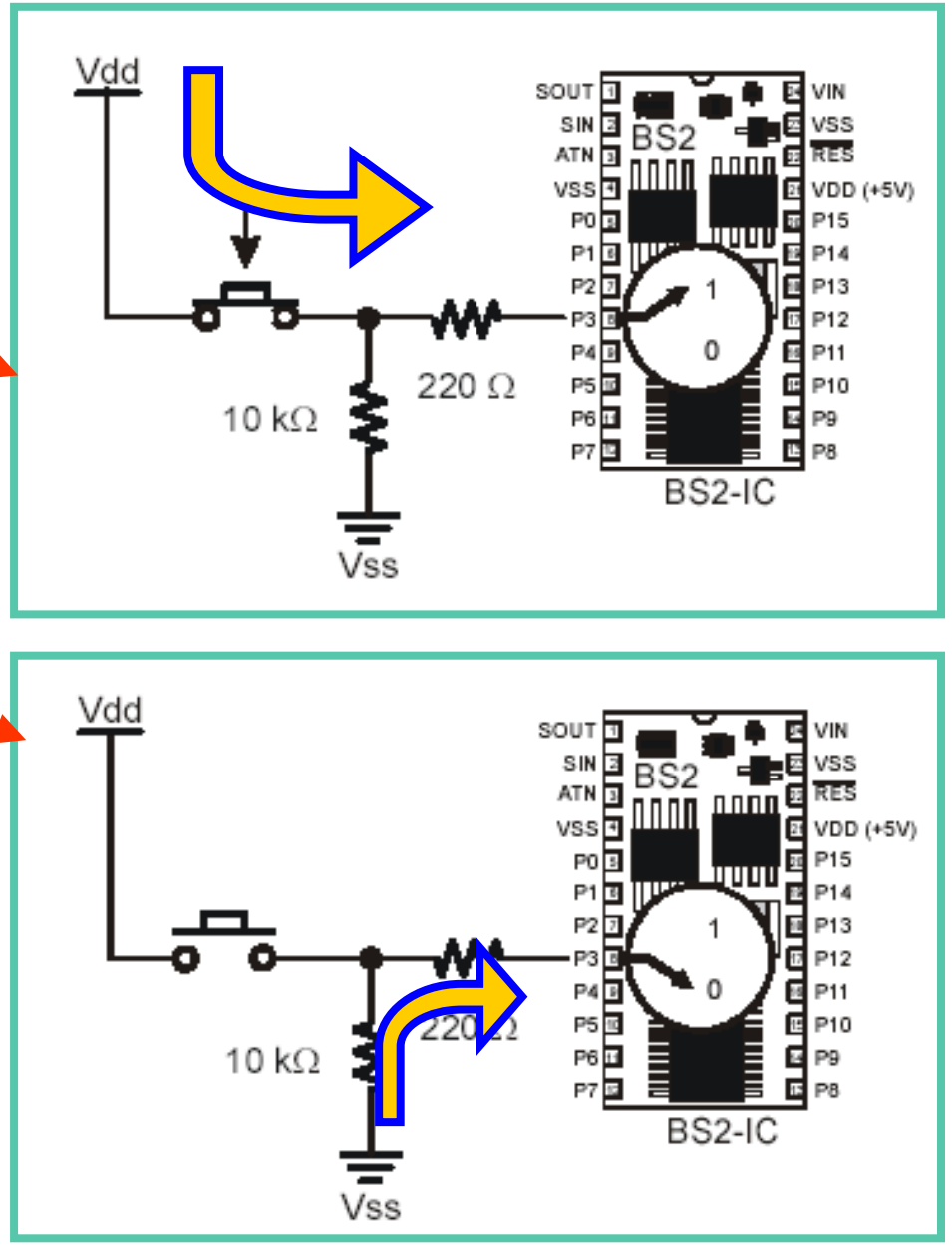
- When button is not pressed, 10270Ω resistance connects P0 to ground. 10K is pull down resistor.
  - Button not pressed is indicated by P0 low.



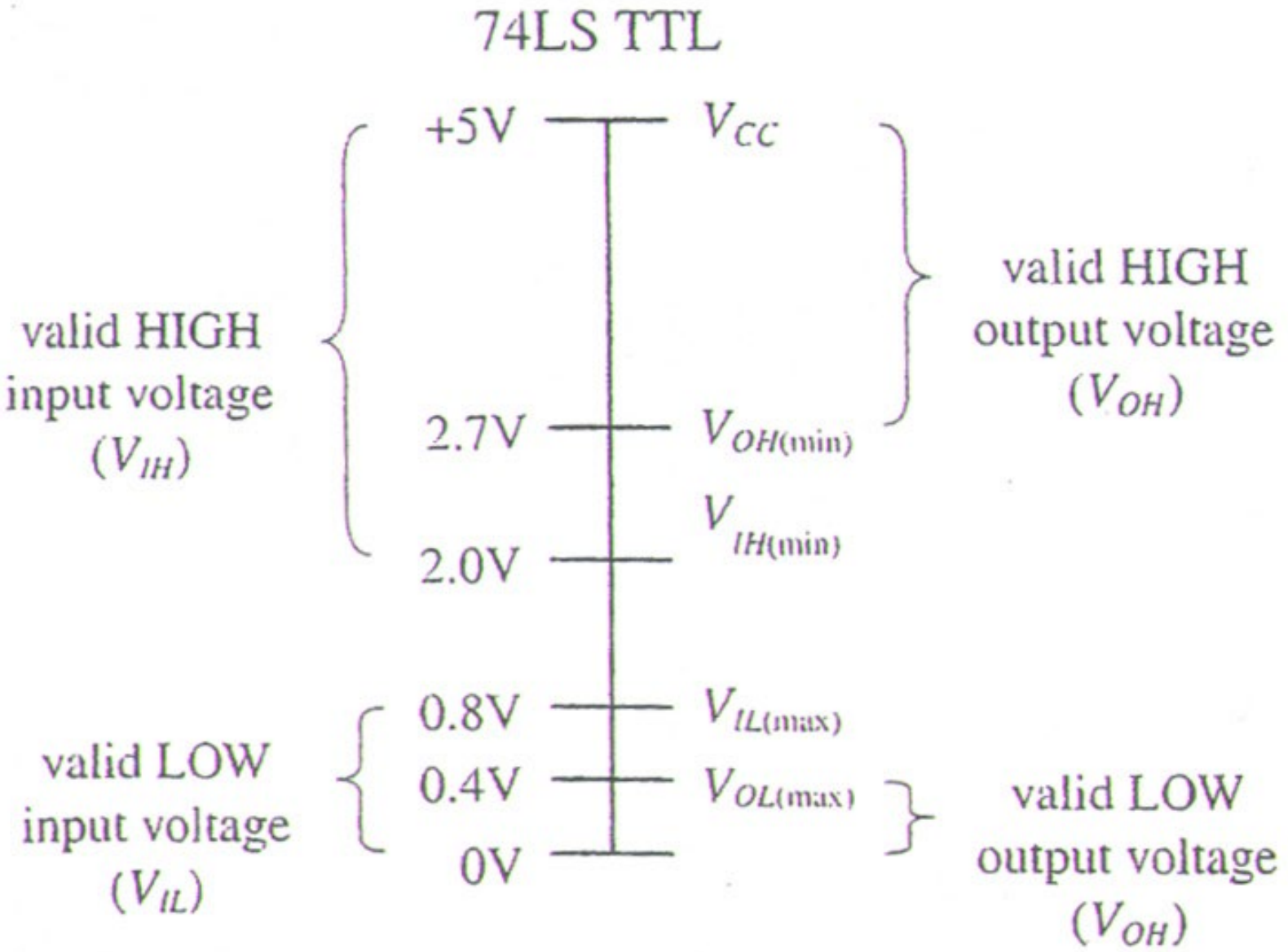
- Big resistance is needed so the switch doesn't short Vdd to Vss!

Pullup and pulldown resistors

- When the switch is pressed, V_{dd} (+5V) is sensed at the input of P3.
- When the switch is released, V_{ss} (0V) is sensed at the input of P3
- **The 10KΩ resistor prevents a short circuit from V_{dd} to V_{ss}**



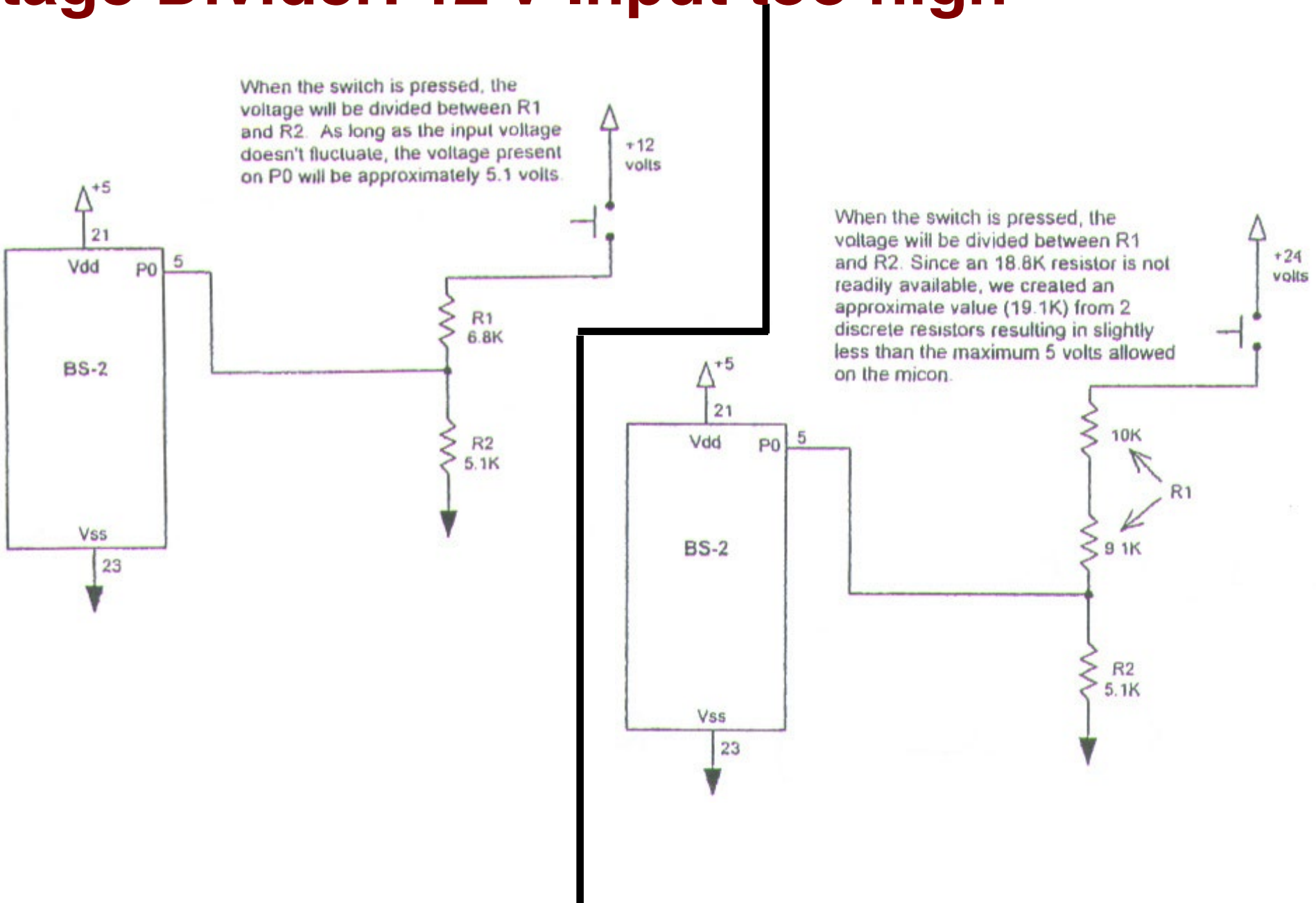
Valid Input/Output Logic Levels for TTL



Signal Level Shifting

- Input signals applied at any BS2 I/O pins must be:
  - 0 volt to be sensed off.
  - 5 volts to be sensed on.
- Input signals must not sink more than 20mA on any I/O pins.
- **Many real-world sensors may present voltage levels that exceed 5V or do not cross 2V threshold to be detected as high!**
- Larger than 5V will destroy BS2.
- If a sensor switches from 0V (off) to 1.0V (on), BS2 cannot detect it!
- TTL logic requires input to be larger than 2V to be sensed as high (BS2 senses voltage inputs $\geq$ 1.4V as high).

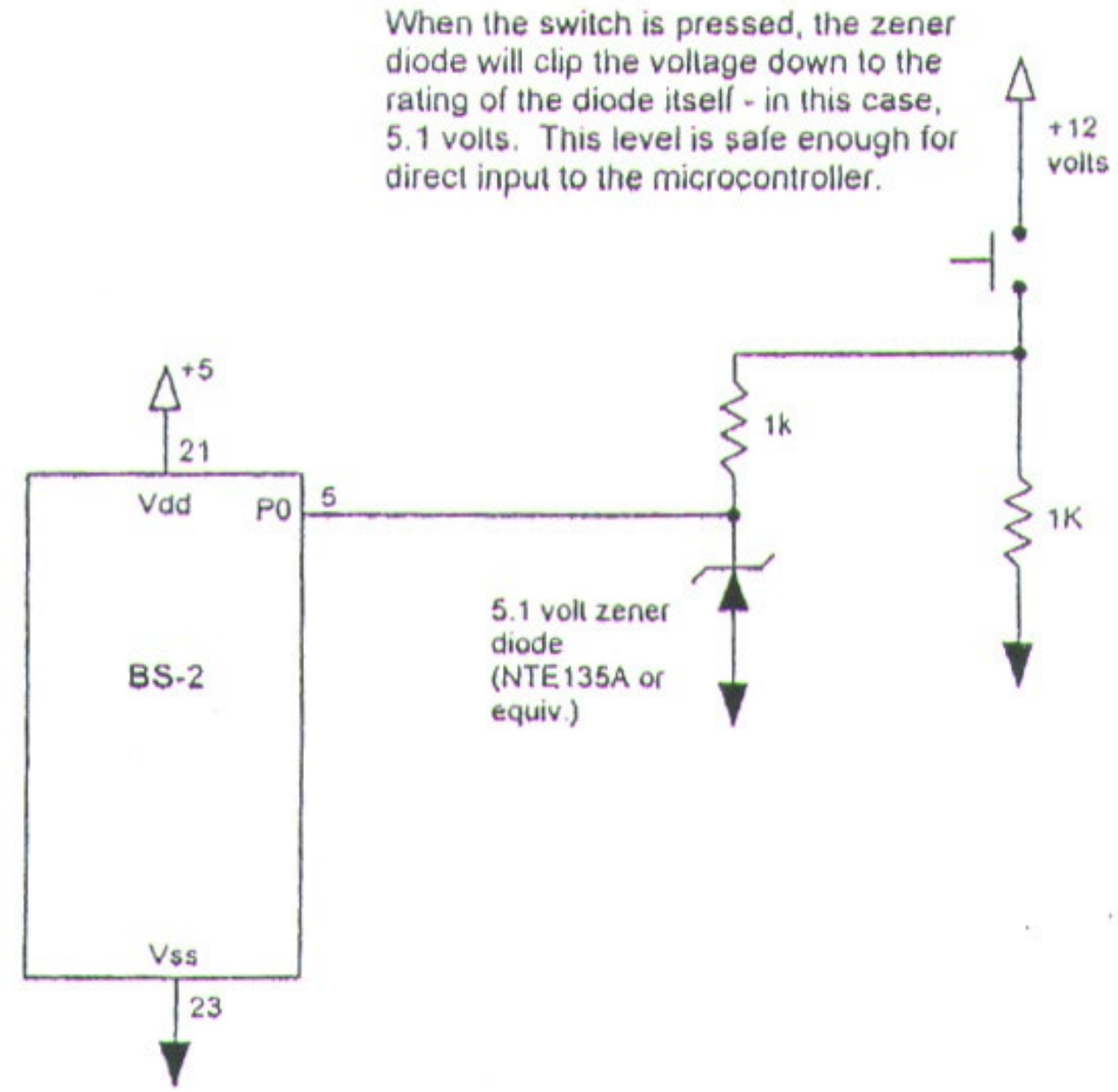
Voltage Divider: 12 V Input too high



Voltage Divider

- Resistor math is very annoying!
- Can we make a simpler voltage divider for our 12 V input?
  - Use a Zener diode!
    - NTE 135A has a forward voltage drop of 1.2 V (not helpful) but a reverse voltage of 5.1 V (very close to HIGH for microcontroller!)

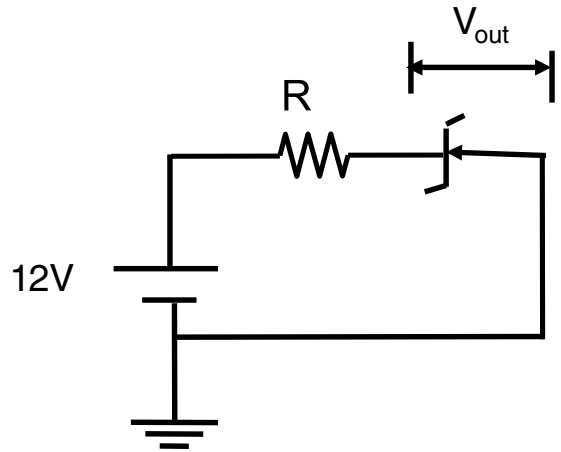
Downward Shift in Signal Level: Zener Circuit —II



Zener Diode Math

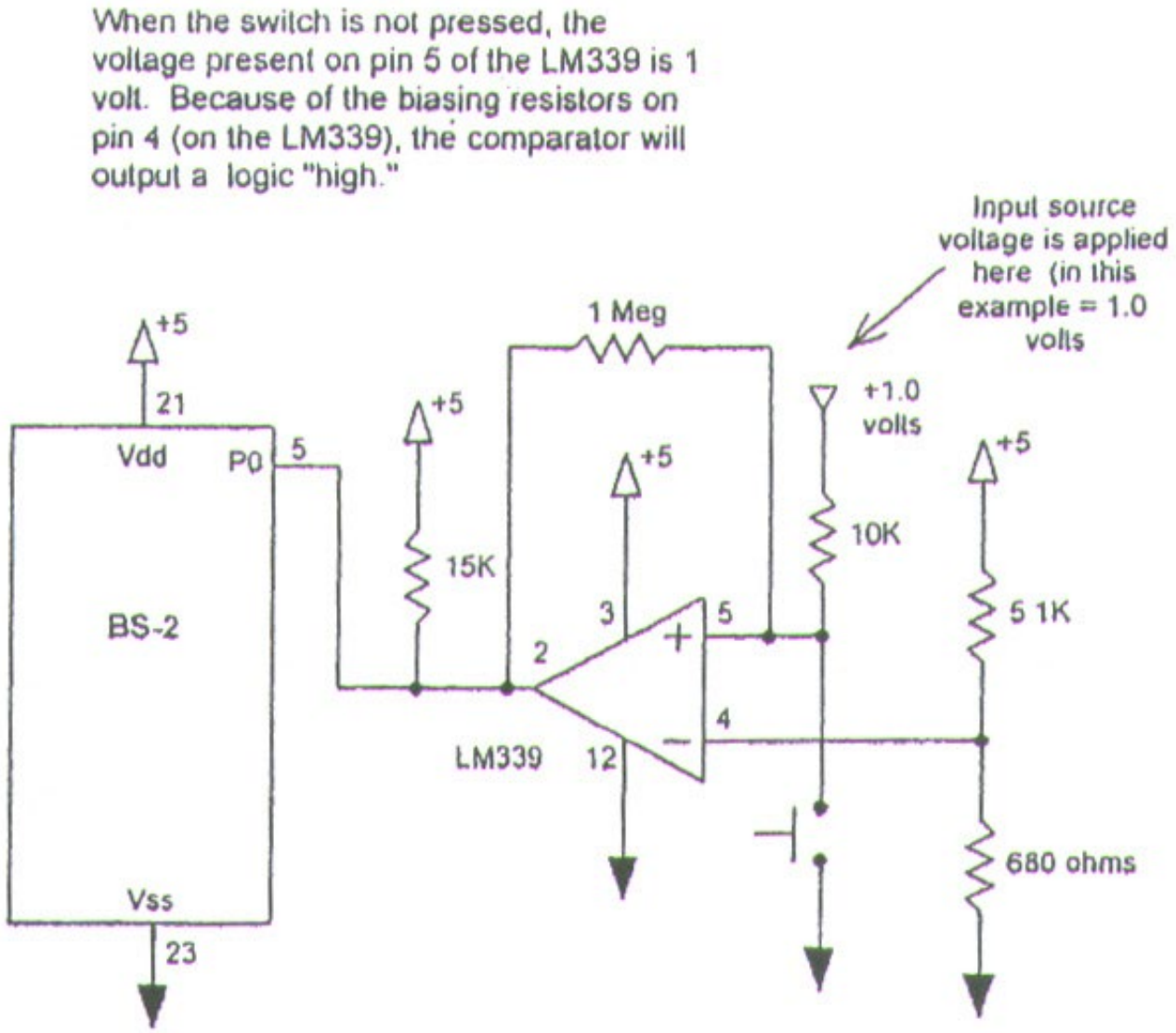
- A series circuit of a zener diode (NTE 135A or equivalent) and a resistor R is connected to a 12V battery.
- The zener is rated at 5.1V so, of the 12V input, 5.1V will be presented across zener and the rest (6.9V) across R.
- Select R to limit current sunk into BS2 I/O pin to less than 20mA. Let R= 1KΩ →

$$I = \frac{6.9\ volts}{1000\ \Omega} = 6.9mA << 20mA$$



Comparator: 1 V Input too low

- For your reference, just know that there exists circuits out there that can do whatever you need



Back to lighting LEDs

Back to lighting LEDs

- Isn't it odd there are 16 I/O pins on BS2?
  - We can light 16 LEDs in various combinations
- Everything about computers comes in two's

bit	1 bit	0,1
nib	4 bits (nibble)	0–15
byte	8 bits (byte)	0–255
word	16 bits (word)	0–65535

Variables and Data

bit	1 bit	0,1
nib	4 bits (nibble)	0–15
byte	8 bits (byte)	0–255
word	16 bits (word)	0–65535

Variables

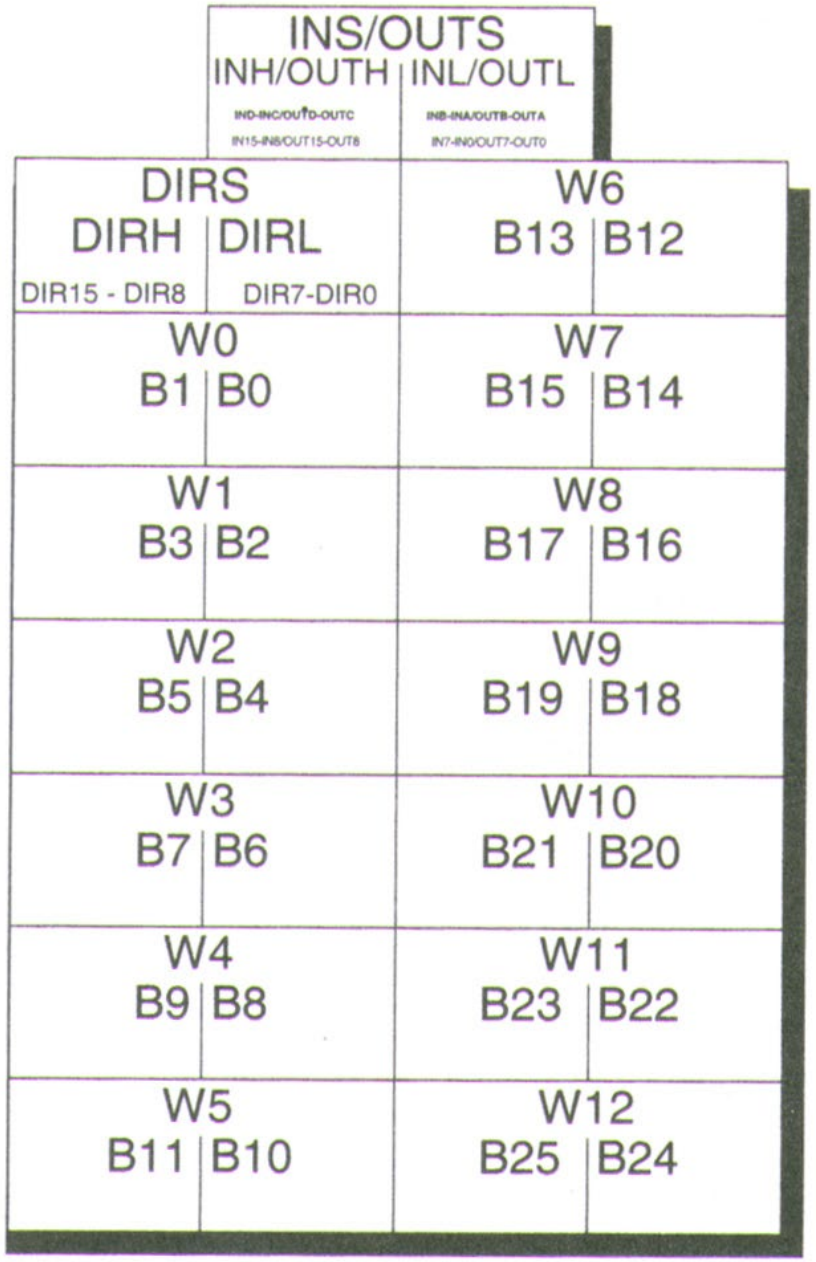
- High and Low mean exactly what you think
- Highbyte is the leftmost 8 bits in a word, Lowbyte is the rightmost

lowbyte	low byte of a word
highbyte	high byte of a word
byte0	low byte of a word
byte1	high byte of a word
lownib	low nibble of a word or byte
highnib	high nibble of a word or byte
nib0–3	numbered (low to high) nibbles of a word or byte
lowbit	low bit of a word, byte, or nibble
highbit	high bit of a word, byte, or nibble
bit0–15	numbered (low to high) bits of a word, byte, or nibble

Variable Modifiers

BS2 Memory Map

- BS2 has 32 bytes of RAM
- 26 bytes of RAM can be used for data/variables.
- 6 bytes are reserved for I/O:
  - 2 bytes for INS register
  - 2 bytes for OUTS register
  - 2 bytes for DIRS register
  - Register is like cash register, a temporary holding place for data



I/O Registers —DIRS

DIRS: 1 for output, 0 for input

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIRD				DIRC				DIRB				DIRA			
DIRH								DIRL							

- DIR0=0 and DIR1=1 → P0 is input pin and P1 is output pin, respectively.
- DIRA=%0110 → P0: input, P1: output, P2: output, P3: input
  - This can also be written as DIRA=6; however clearly binary form is more readable.
  - % is to indicate binary number

I/O Registers —OUTS

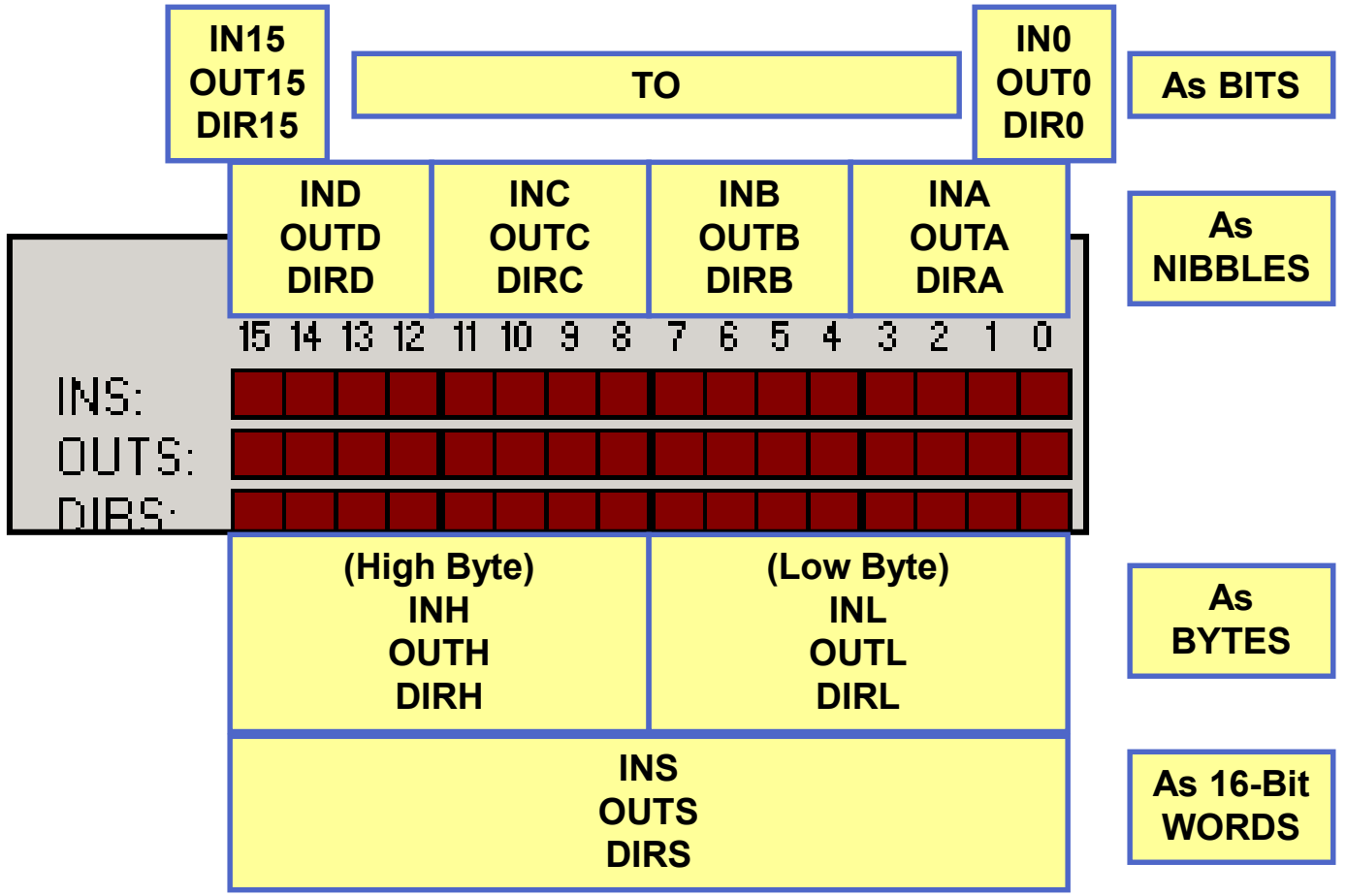
OUTS: 1 high, 0 for low

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OUTD				OUTC				OUTB				OUTA			
OUTH								OUTL							

- OUT0=0 and OUT1=1 → P0 is driven low and P1 is driven high, respectively.
- OUTA=%0110 → P0: low, P1: high, P2: high, P3: low
- Aside: INS works similar to OUTS. INS is used to access state of various input pins.

I/O Register Modifiers

- The I/O can also be addressed as nibbles, bytes or the entire word.



Summary: BS2 Memory Map

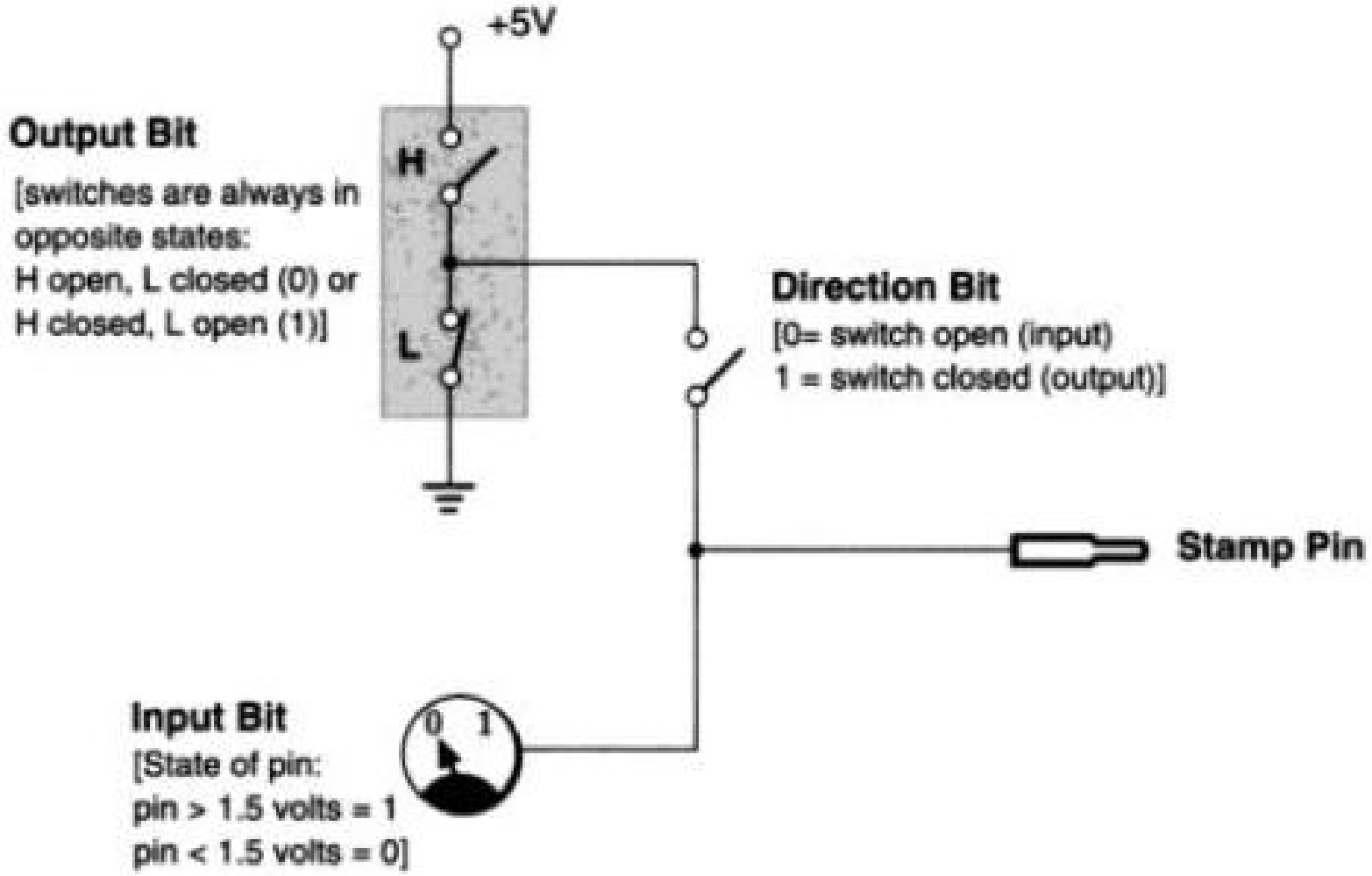
- 16 words, of two bytes each for a total of 32 bytes.
- All bits are individually addressable through variable modifiers.
- The bits within the upper three words are also individually addressable though the pre-defined names shown.
- All registers are word, byte, nibble, and bit addressable.

Word Name	Byte Names	Nibble Names	Bit Names	Special Notes
INS*	INL, INH	INA, INB INC, IND	IN0 – IN7 IN8 – IN15	Input pins
OUTS*	OUTL, OUTH	OUTA, OUTB OUTC, OUTD	OUT0 – OUT7 OUT8 – OUT15	Output pins
DIRS*	DIRL, DIRH	DIRA, DIRB DIRC, DIRD	DIR0 – DIR7 DIR8 – DIR15	I/O pin direction control
W0	B0, B1			
W1	B2, B3			
W2	B4, B5			
W3	B6, B7			
W4	B8, B9			
W5	B10, B11			
W6	B12, B13			
W7	B14, B15			
W8	B16, B17			
W9	B18, B19			
W10	B20, B21			
W11	B22, B23			
W12	B24, B25			

BS2 Power Up/Reset

- When the BASIC Stamp is powered up, or reset, all memory locations are cleared to 0
  - All pins are inputs (DIRS = %0000000000000000).
- If the PBASIC program sets all the I/O pins to outputs (DIRS = %1111111111111111)
  - All pins will initially output low, since the output latch (OUTS) is cleared to all zeros upon power-up or reset, as well.

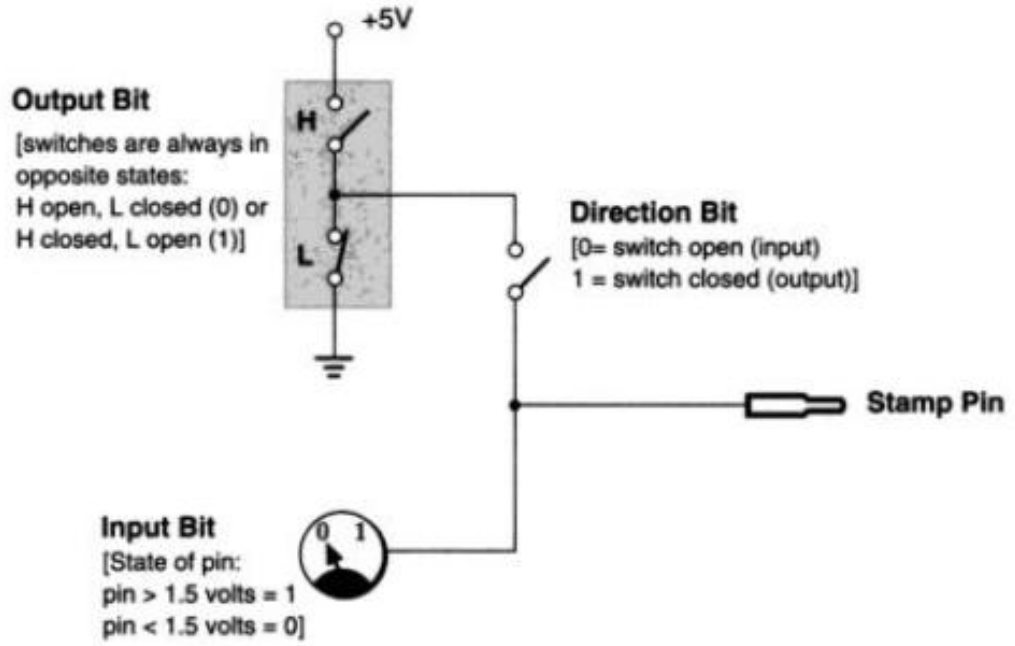
I/O—Behind the Scenes



Input/output circuitry of a Stamp pin.

I/O—Behind the Scenes

- An I/O pin's behavior is determined by the direction bit, output bit, and input bit
- Output bit, e.g., OUT0
  - a pair of MOSFET switches controlled by BS2
  - always in opposite states (one H, one L)
  - OUT0=1 puts a 1 in the output bit of P0, regardless of the direction bit of P0

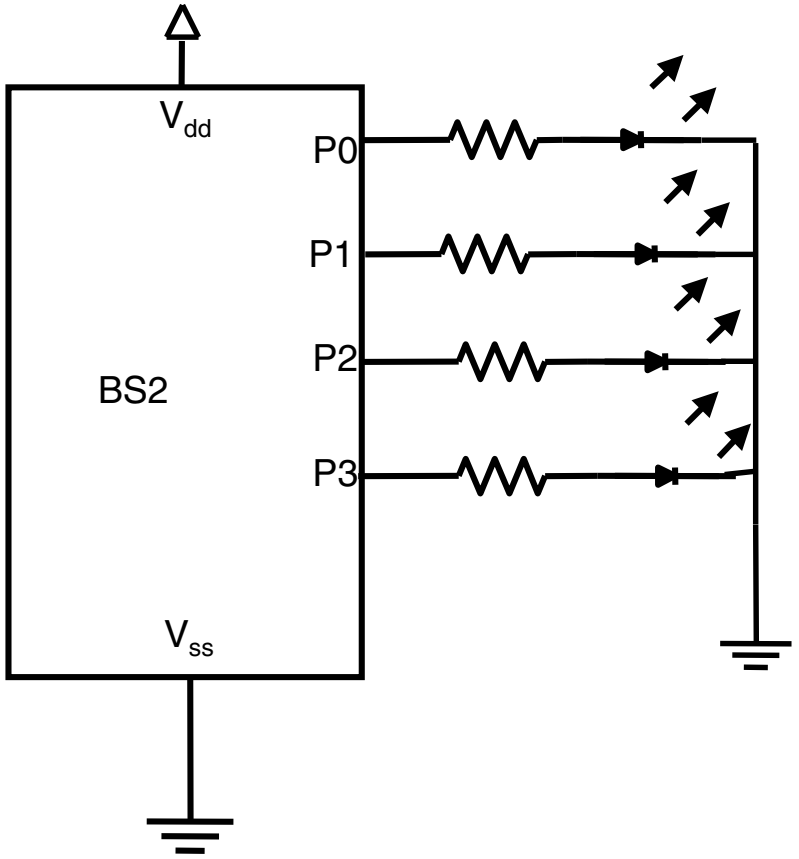


Input/output circuitry of a Stamp pin.

- Direction bit, e.g., DIR0
  - a MOSFET switch that is open when DIR0=0 and closed when DIR1=1
  - closing the direction bit switch puts the output bit OUT0 on P0
- Input bit, e.g., IN0
  - always connected to P0 and its state always matches the state of P0
  - when DIR0=1, IN0 will be equal to OUT0
  - when DIR0=0, IN0 will match the state of P0

Lighting up 4 LEDs

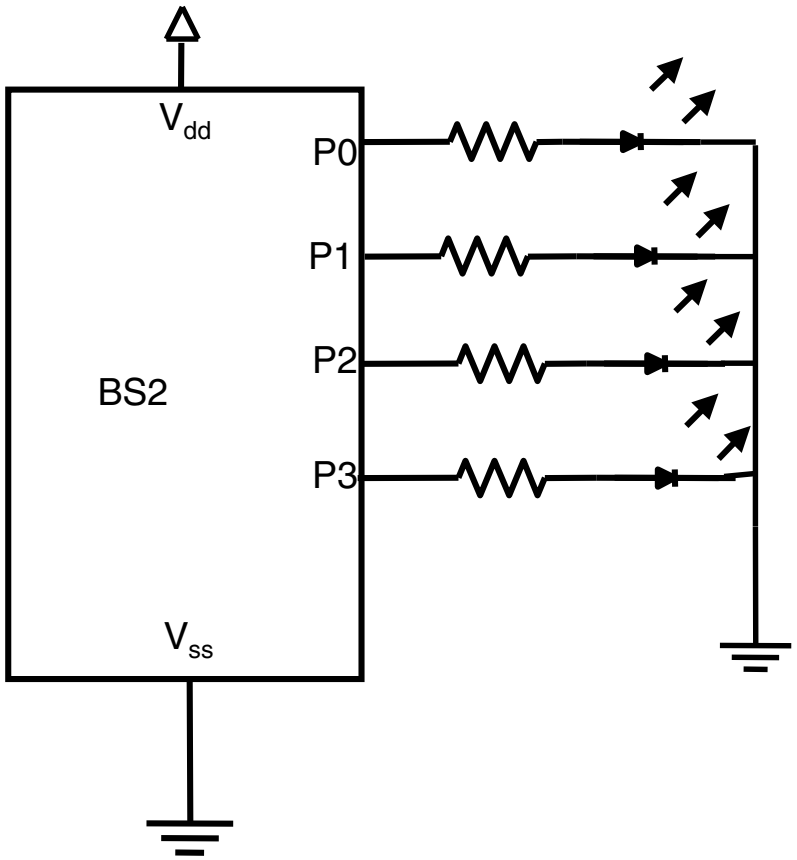
- Consider the following diagram with 4 LEDs connected to P0-P3 where BS2 sources current.



LED Display

```
Ledpins VAR Nib
  for Ledpins = 0 to 3
       output Ledpins
       high Ledpins
       pause 1000
       low Ledpins
  next
```

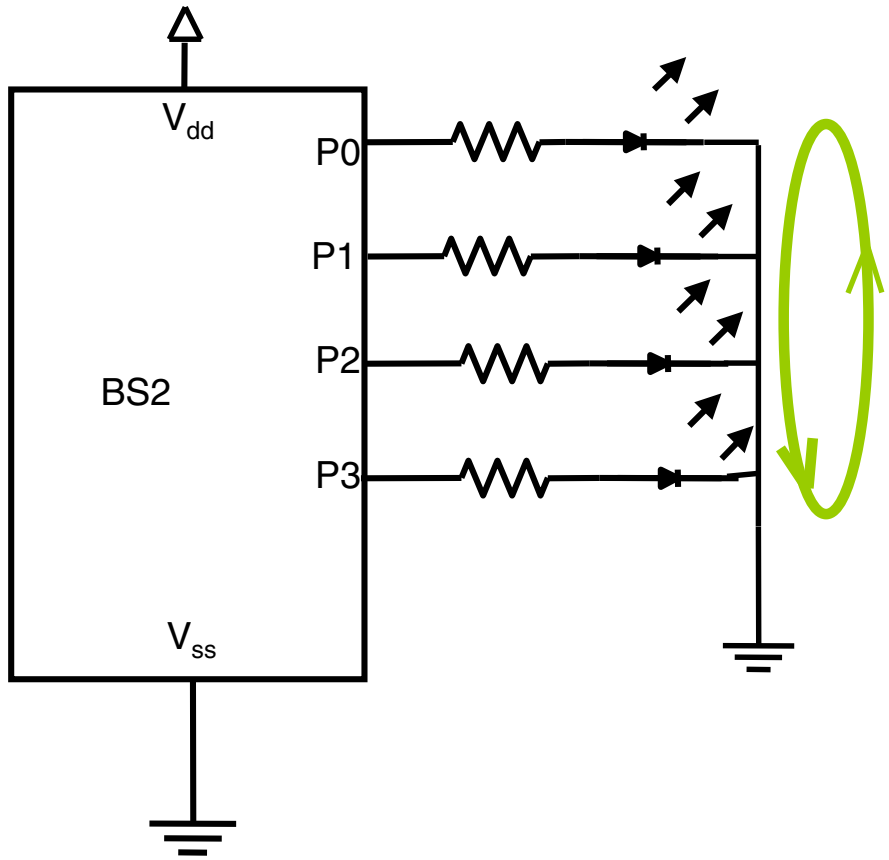
```
Ledpins VAR Nib
   For Ledpins = 3 to 0
    output Ledpins
    high Ledpins
    pause 1000
    low Ledpins
   next
```



PBASIC is case-insensitive!
And does not care about whitespace!

Forward/Backward LED Display

```
LEDs VAR OutL
DlyTime CON 1000
DirL = %00001111
LEDs = %00000001
FwdDisp:
pause DlyTime
LEDs = LEDs<<1
If LEDS = 00001000 then RevDisp
Goto FwdDisp
RevDisp:
Pause DlyTime
LEDs = LEDs>> 1
If LEDs = 00000001 then FwdDisp
Goto RevDisp
End
```



Hands-on Exercises: Circuit Components

What's a Microcontroller? LED	Chapter 2
StampWorks Manual LED	pp. 25–56
What's a Microcontroller? Button	Chapter 3
StampWorks Manual Button	Exp #15

Hands-on Exercises: Code

BASIC Stamp Syntax and Reference Manual 2.2	
If … then	pp. 231 – 242
For … next	pp. 191 – 197
Variables	pp. 85 – 86
**	pp. 111 – 112
*/	p. 112
//	pp. 113 – 114
Binary Operators	pp. 109 – 122