Lecture 3

Digital circuits, power supplies and regulation

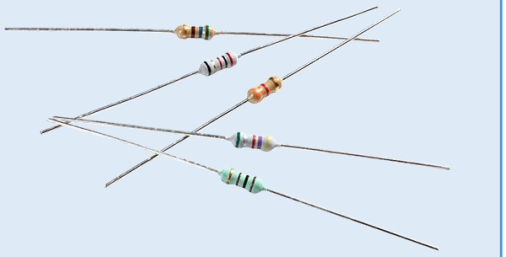
Graphics used with permission from AspenCore (http://electronics-tutorials.ws)

Agenda

1. Review
2. The digital abstraction: logic gates and flip-flops
3. Power supplies and regulation
4. Design project overview
5. Lab 3 overview

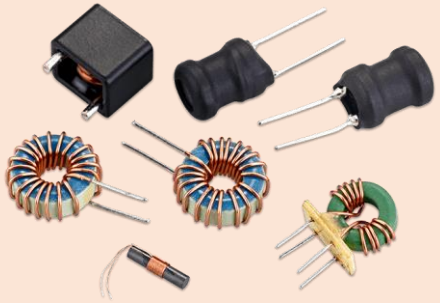
Review: Lecture 1 and Lab 1

- Voltage and current: think **pressure** and **flow**
- Ohm's Law: resistor current is linear, **V = IR**
- Capacitors and inductors: **energy storage** elements
- Capacitors block **low frequencies**, inductors block **high frequencies**



Resistors

Capacitors



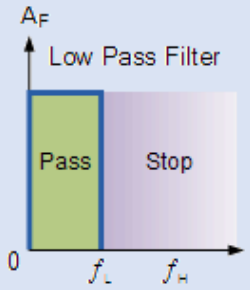
Inductors

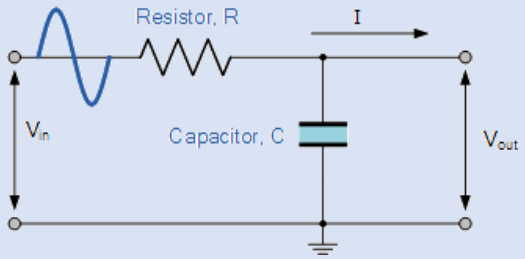
Review: Lecture 2 and Lab 2

- **Filter:** Response depends on the frequency of the input
- **Diode:** One-way device, current from anode to cathode
- **BJT:** Current-controlled amplifier
- **MOSFET:** Voltage-controlled switch
- **Op-amp:** Performs mathematical operations

Types of filters

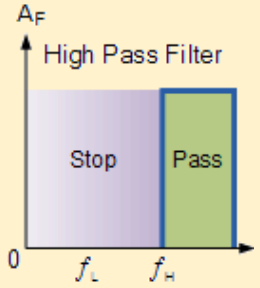
LPF

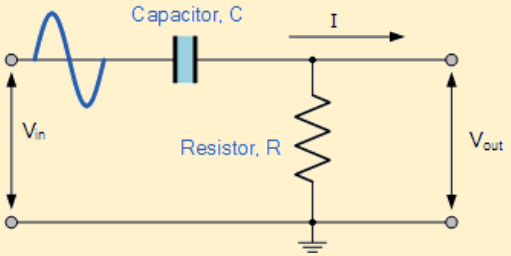




$$f_c = \frac{1}{2\pi RC} Hz$$

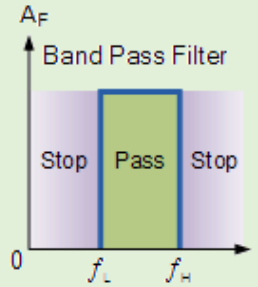
HPF

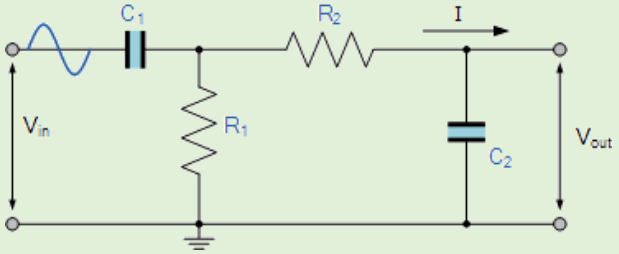




$$f_c = \frac{1}{2\pi RC} Hz$$

BPF





$$f_H = \frac{1}{2\pi R_1 C_1} Hz$$

$$f_L = \frac{1}{2\pi R_2 C_2} Hz$$

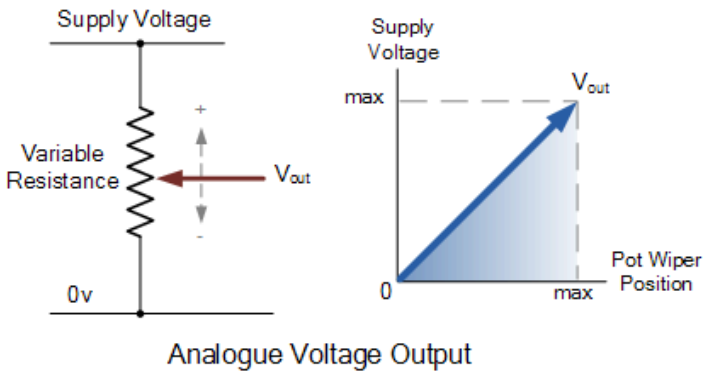
The digital abstraction

Logic gates and flip-flops

Analog vs. digital

Analog

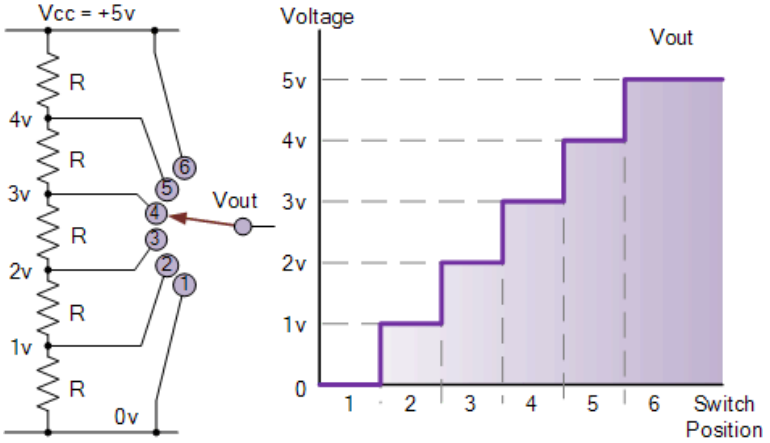
- Continuous
- Sensitive to noise
- Modular design difficult



Analogue Voltage Output

Digital

- Discrete
- Noise-resistant
- Modular design easy



Digital Voltage Output

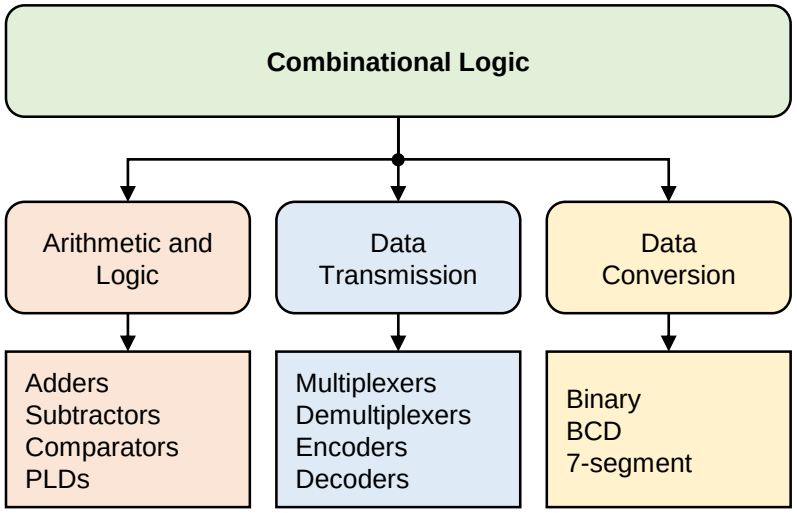
Binary numbers

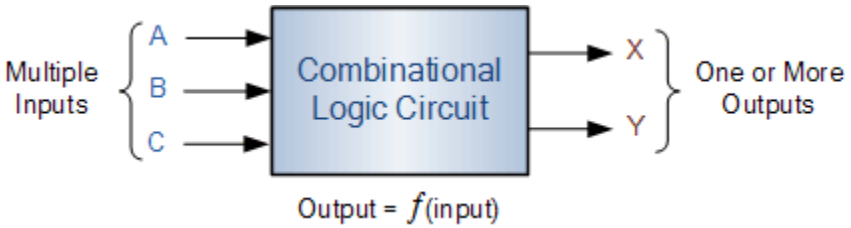
- **Binary:** represents numbers as 1's and 0's
- Enables us to perform computations with digital circuits
- Conversion to decimal: *n*th bit worth 2^n

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**		
Value	**128**	**64**	**32**	**16**	**8**	**4**	**2**	**1**		
Example 1	1	1	1	1	1	1	1	1	=	**255**
	128 +	*64 +*	*32 +*	*16 +*	*8 +*	*4 +*	*2 +*	*1*		*(sum)*
Example 2	0	1	1	1	0	1	0	1	=	**117**
	0 +	*64 +*	*32 +*	*16 +*	*0 +*	*4 +*	*0 +*	*1*		*(sum)*

Combinational logic

- **Combinational logic circuit:** Output is a function of its inputs, independent of time
- Constructed from basic elements called **gates**

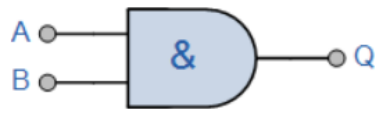




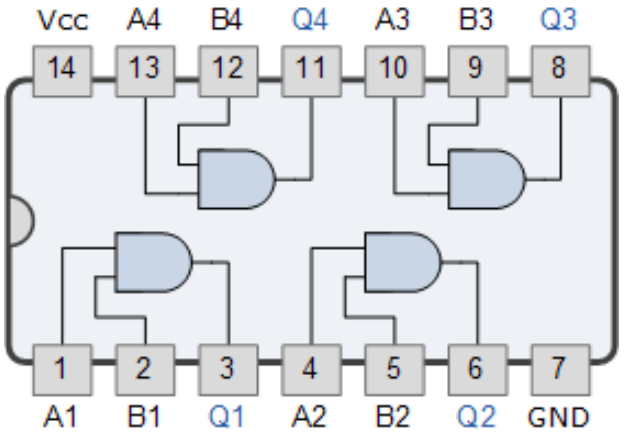
AND gate

- **AND gate:** Output is 1 (“high”) when all inputs are 1
- Equivalent to “**multiplication**”: denoted by asterisk (*) or period (.)

2-input AND gate truth table

Symbol	Truth Table		
 2-input AND Gate	B	A	Q
	0	0	0
	0	1	0
	1	0	0
	1	1	1
Boolean Expression **Q = A.B**	Read as A **AND** B gives Q		

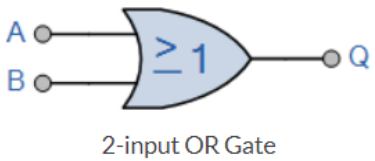
74LS08 Quad 2-input AND gate



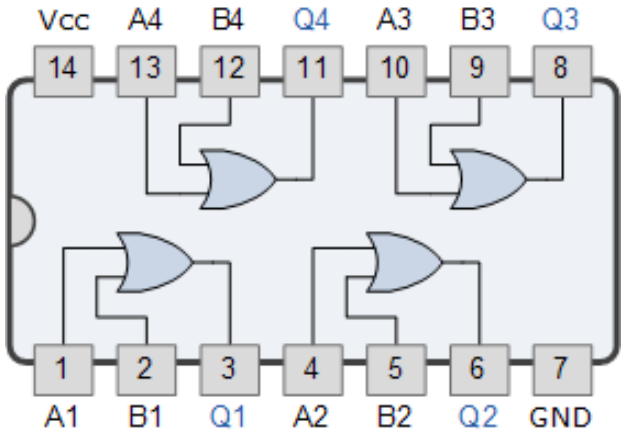
OR gate

- **OR gate:** Output is 1 when at least 1 input is 1
- Equivalent to "**addition**": denoted by plus sign (+)

2-input OR gate truth table

Symbol	Truth Table		
 2-input OR Gate	B	A	Q
	0	0	0
	0	1	1
	1	0	1
	1	1	1
Boolean Expression **Q = A+B**	Read as A **OR** B gives Q		

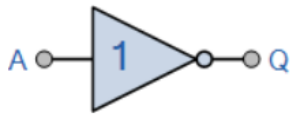
74LS32 Quad 2-input OR gate



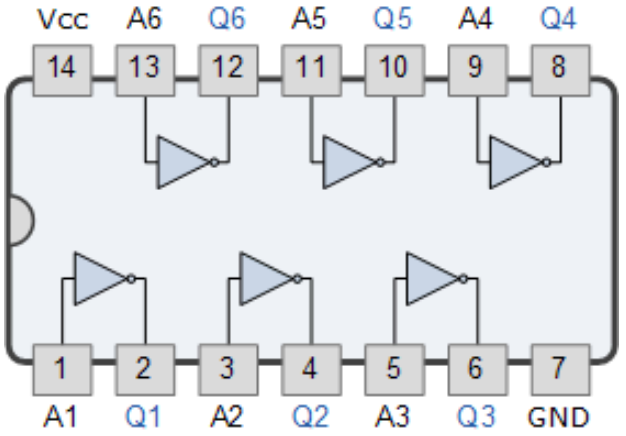
NOT gate

- **NOT gate:** single input, output is “inverse” of input
- Denoted by a horizontal line (not A = $\overline{A}$)
- Inverted outputs and inputs denoted by small “**bubble**”

NOT gate truth table

Symbol	Truth Table	
 Inverter or NOT Gate	A	Q
	0	1
	1	0
Boolean Expression **Q = not A or $\overline{A}$**	Read as inverse of **A** gives Q	

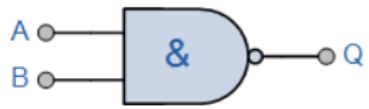
74LS04 Hex NOT gate (inverter)



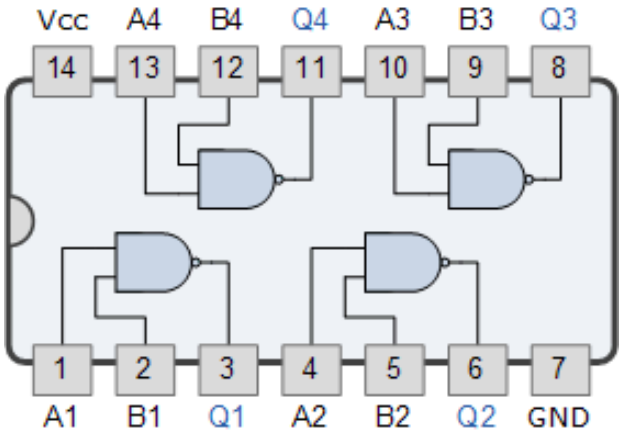
NAND gate

- **NAND gate:** Output is 0 when all inputs are 1
- Same as an AND gate followed by an inverter
- NAND gate is “universal”: any function can be constructed using only NAND gates

2-input NAND gate truth table

Symbol	Truth Table		
 2-input NAND Gate	B	A	Q
	0	0	1
	0	1	1
	1	0	1
	1	1	0
Boolean Expression $Q = \overline{A.B}$	Read as A **AND** B gives **NOT** Q		

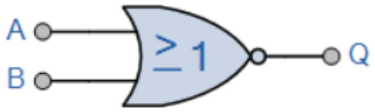
74LS00 Quad 2-input NAND gate



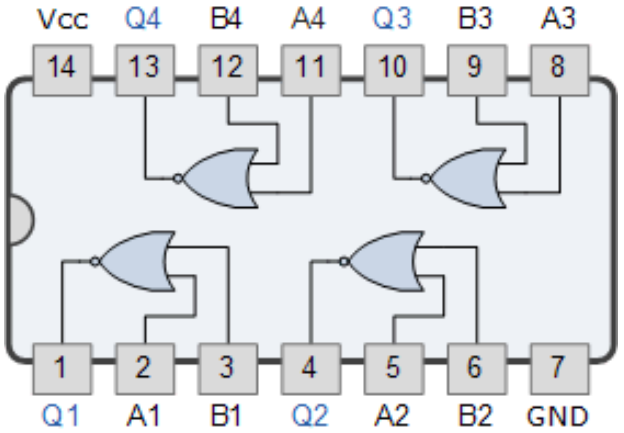
NOR gate

- **NOR gate:** Output is 0 when at least 1 input is 1
- Same as an OR gate followed by an inverter
- NOR gate is “universal”: any function can be constructed using only NOR gates

2-input NOR gate truth table

Symbol	Truth Table		
 2-input NOR Gate	B	A	Q
	0	0	1
	0	1	0
	1	0	0
	1	1	0
Boolean Expression $Q = \overline{A+B}$	Read as A **OR** B gives **NOT** Q		

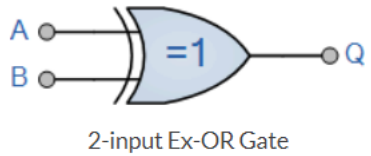
74LS02 Quad 2-input NOR gate



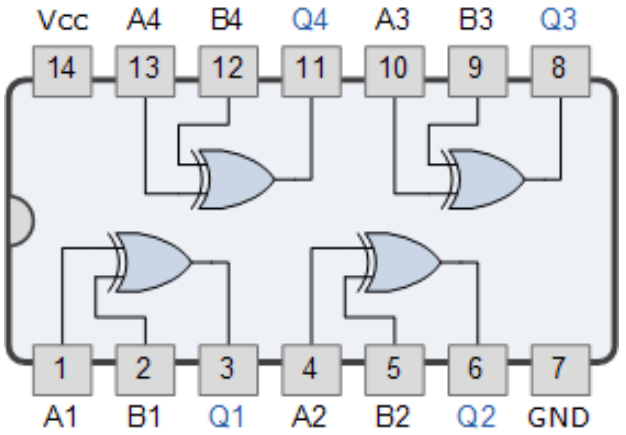
XOR gate

- **XOR gate:** Output is 1 when one (but not 2) input is 1
- XOR = “exclusive OR”: output is only 1 when inputs are different from each other

2-input XOR gate truth table

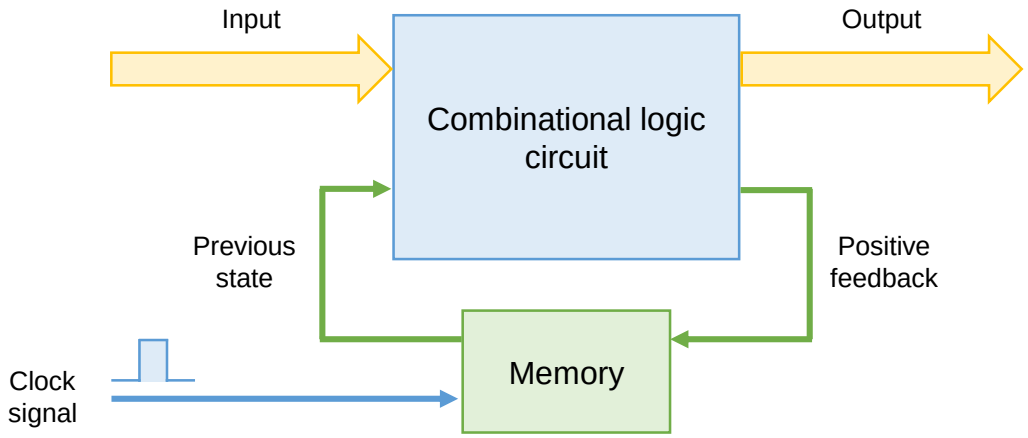
Symbol	Truth Table		
 2-input Ex-OR Gate	B	A	Q
	0	0	0
	0	1	1
	1	0	1
	1	1	0
Boolean Expression Q = A ⊕ B	A **OR** B but NOT **BOTH** gives Q		

74LS86 Quad 2-input XOR gate



Sequential logic

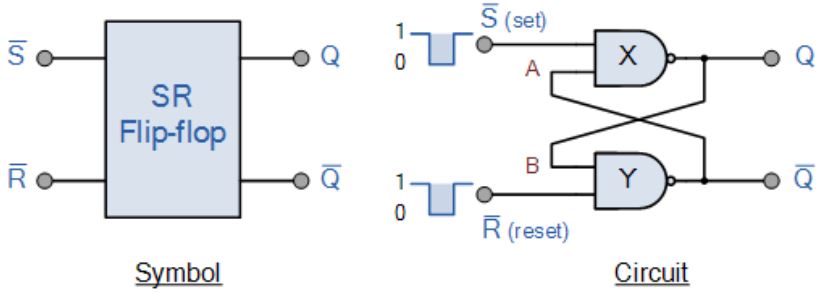
- **Sequential logic:** Logic circuit with memory built in
- Output is time-dependent: function of current inputs, previous inputs and previous outputs



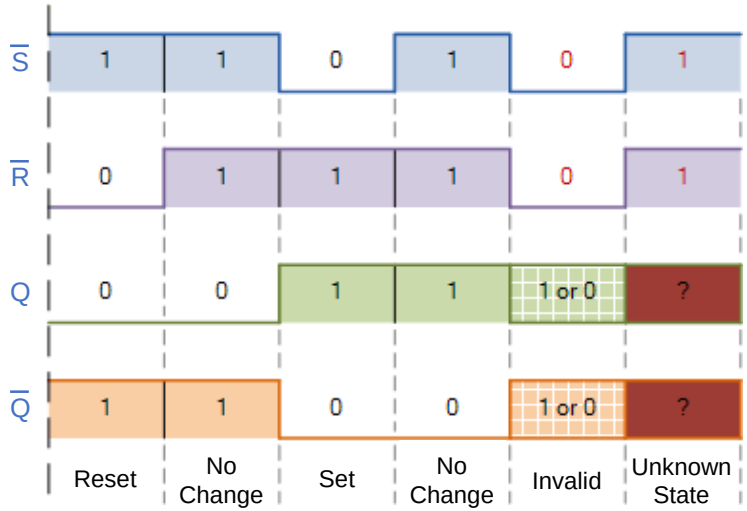
SR Flip-flop

- **SR Flip-flop:** One-bit memory element with 2 inputs
- **Set input** (S): Active-low input, sets output (Q) to 1
- **Reset input** (R): Active-low input, sets output (Q) to 0

SR Flip-flop circuit



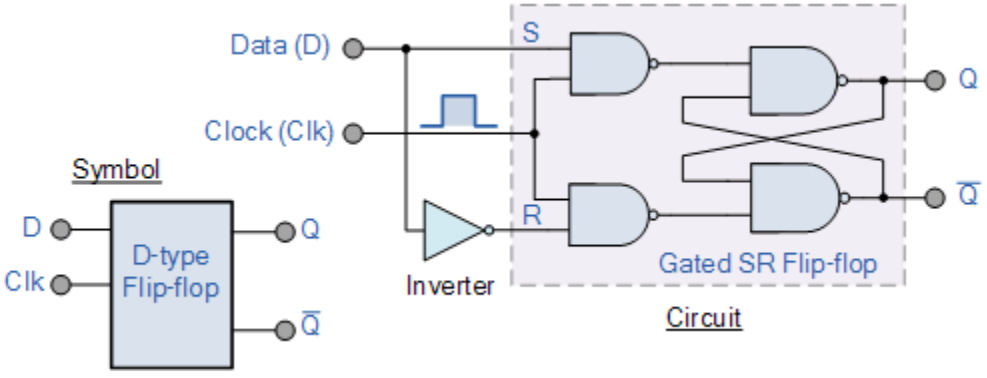
SR Flip-flop timing diagram



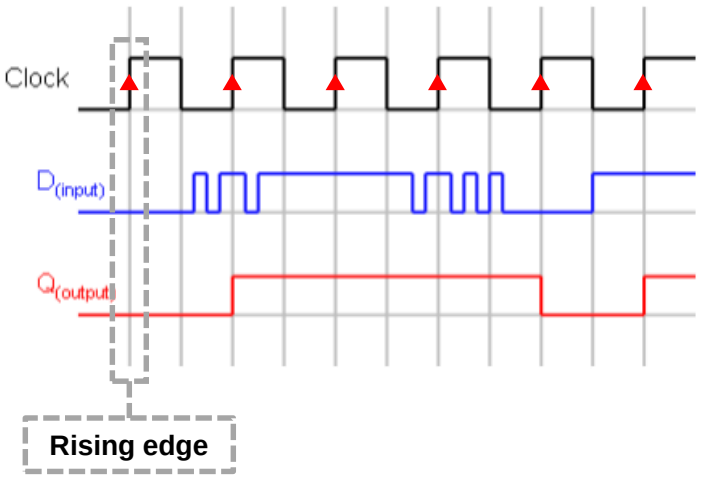
D Flip-flop

- **D Flip-flop:** One-bit memory element with 1 input
- Output takes the value of input at the **rising edge** of CLK
- Output stays constant until next rising edge

D Flip-flop circuit



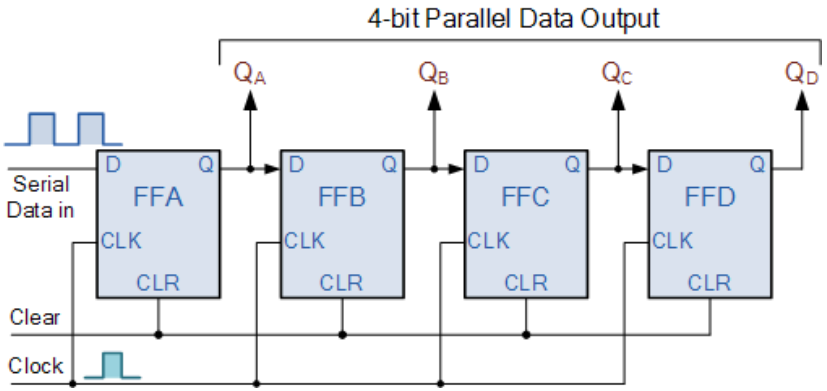
D Flip-flop timing diagram



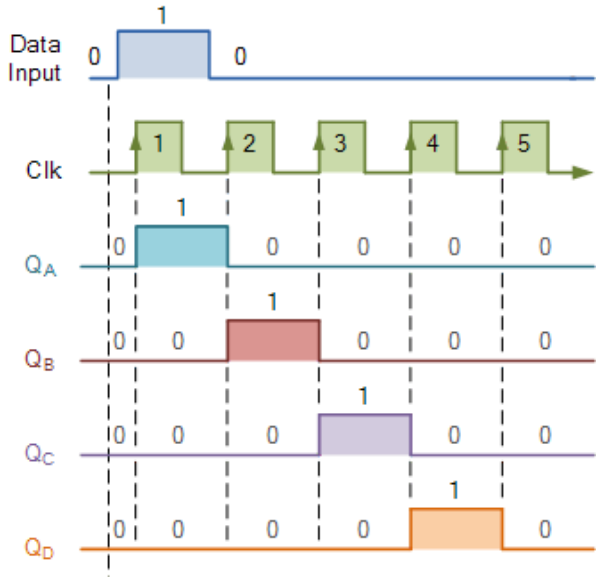
Shift register

- **Serial data:** Bits of data presented sequentially
- **Parallel data:** Bits of data presented concurrently
- **Shift register:** Converts data from serial to parallel

Shift register circuit



Shift register timing diagram

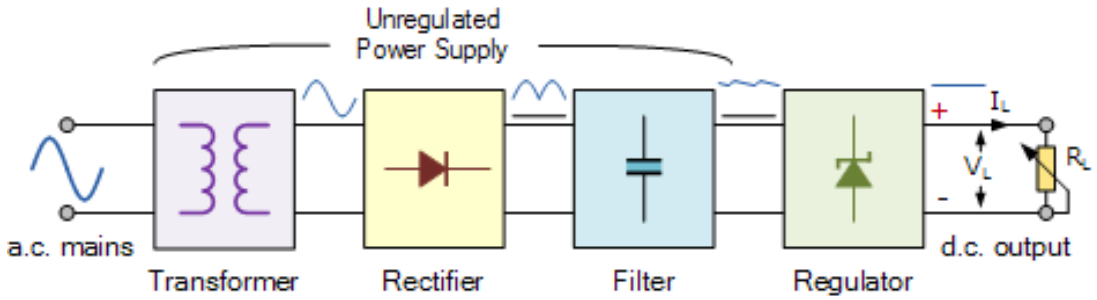


Power supplies and regulation

Power supplies

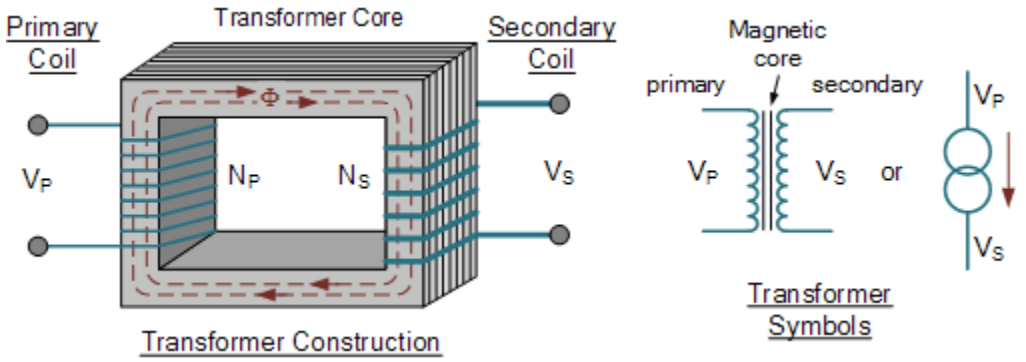
- **Goal:** Convert AC line voltage (120V RMS) into a lower DC voltage usable with common components
- **Process:** Convert to lower voltage with transformer, rectify low voltage, apply regulation (optional) and filter

- May be either unregulated (easy) or regulated (less easy)
- May be either linear (inefficient) or switching (efficient)

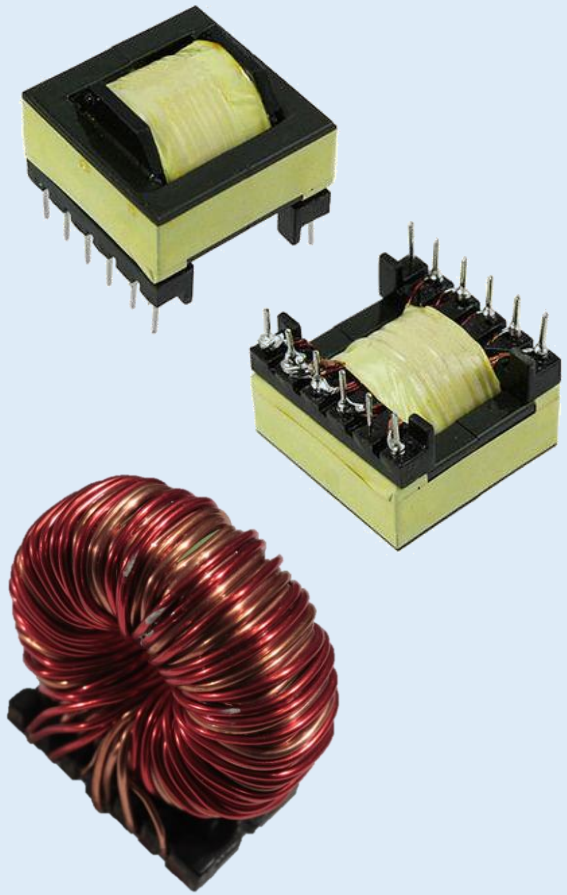


Transformers

- **Transformer:** Converts from one AC voltage to another
- Conversion proportional to **turn ratio** (N_S : N_P)
- Output current calculated by **power conservation**



Transformers

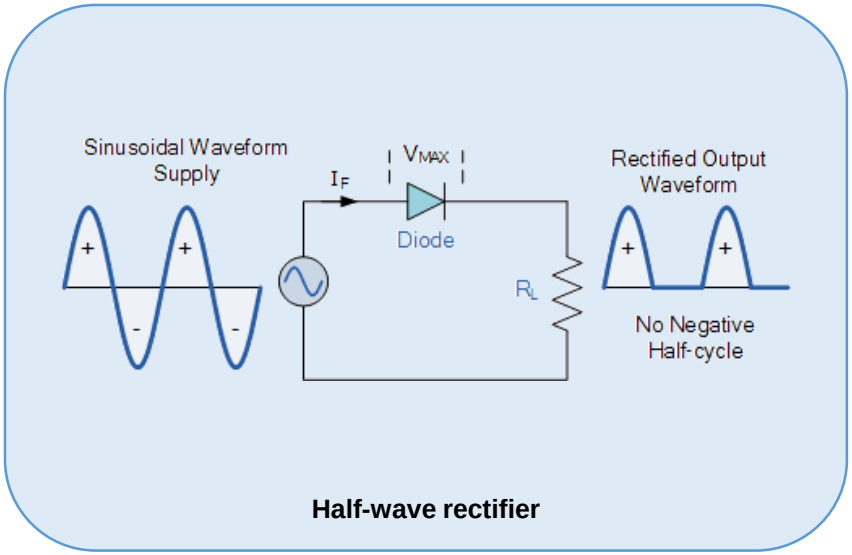


PCB-mounted

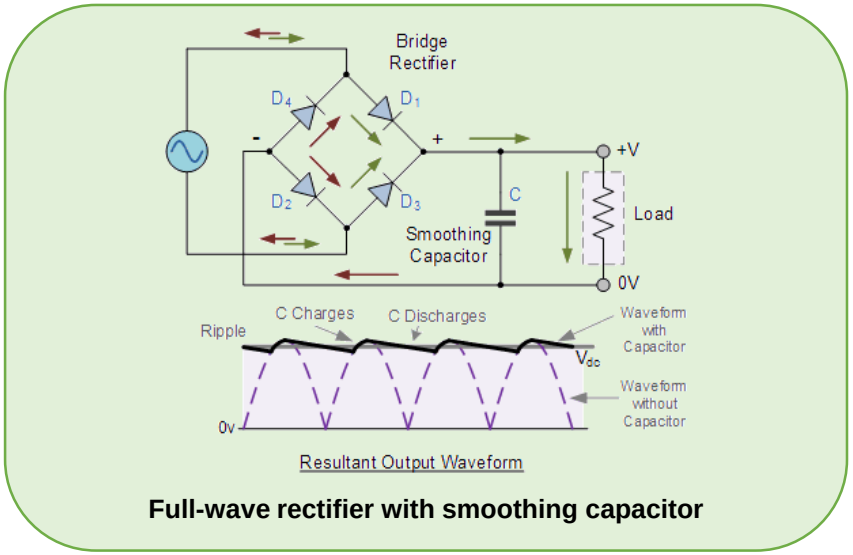
Chassis-mounted

Rectification

- **Rectifiers** convert AC voltages to DC
- Can be either half-wave (one diode) or full-wave (four diodes)
- Full-wave rectifiers use both half cycles of the AC input

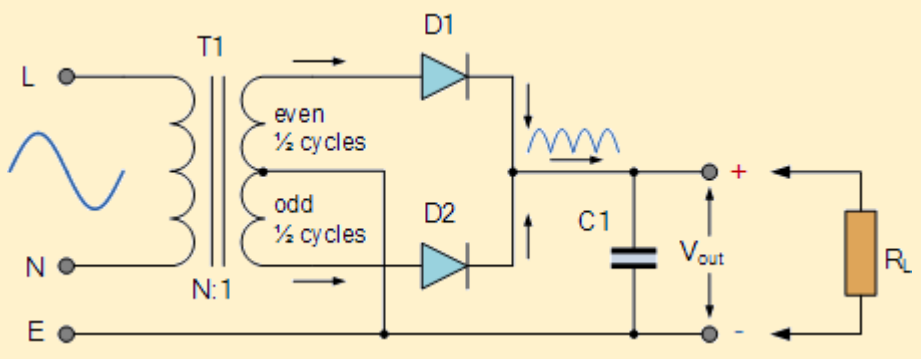


Half-wave rectifier



Full-wave rectifier with smoothing capacitor

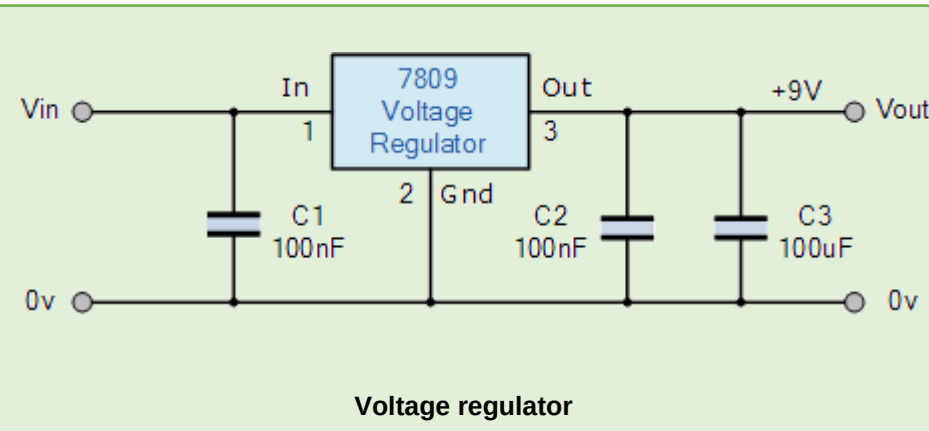
Linear power supply



Unregulated power supply

Unregulated power supply

- T1 converts AC line voltage into lower voltage AC
- **Center tap** of T1 placed in middle of secondary winding
- D1, D2 form a full-wave rectifier
- C1 filters output to produce DC



Voltage regulator

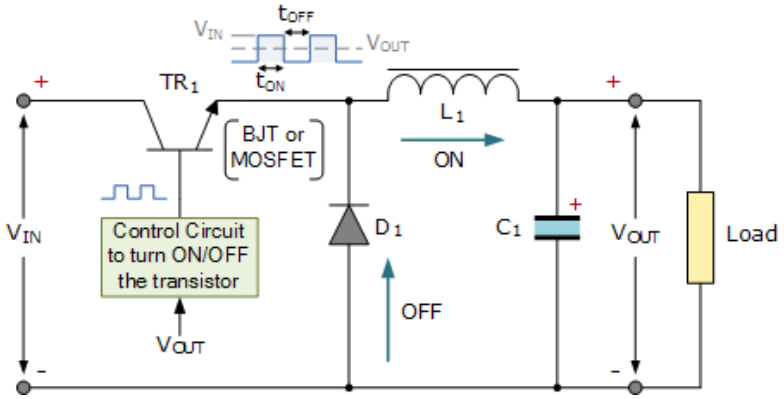
Voltage regulator

- LM78xx series: voltage regulator IC, available from 5V to 24V
- C1, C2, C3 are input and output filter capacitors
- Produces reliable DC voltage from unreliable transformer output

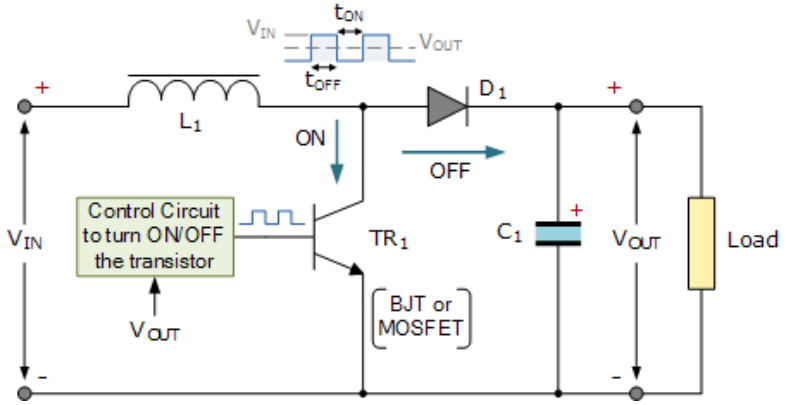
Switching power supply

- **Switching power supply:** Uses transistors (usually MOSFETs) to switch a transformer/inductor
- Much more **efficient** than linear power supplies, but extremely complicated to design
- May either step down ("**buck**") or up ("**boost**")

Switching "buck" converter

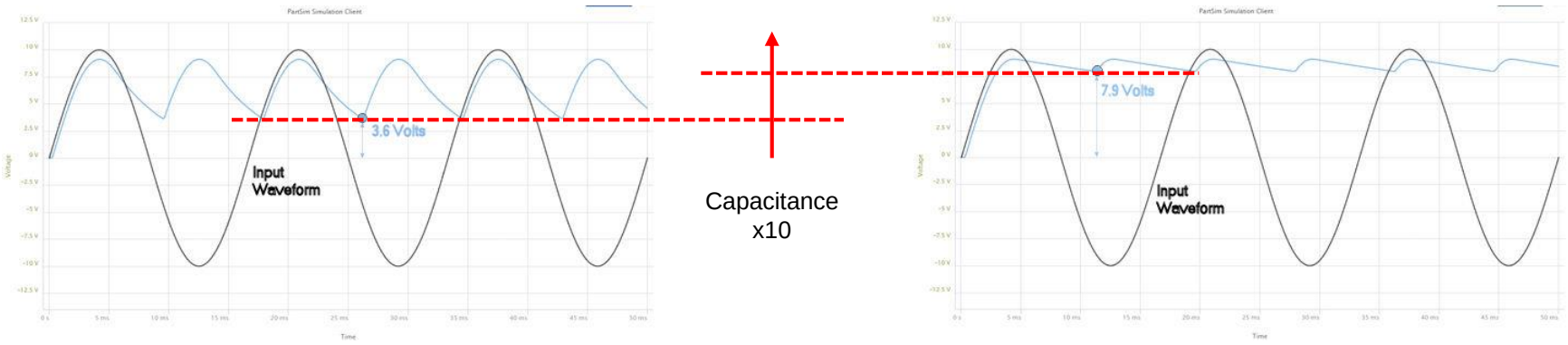


Switching "boost" converter



Supply filtering

- After rectification, supply must be **filtered** to get DC
- When combined with load and source resistance, output capacitor forms an RC low-pass filter (LPF)
- In general, larger capacitance = low cutoff frequency = **smoother output**



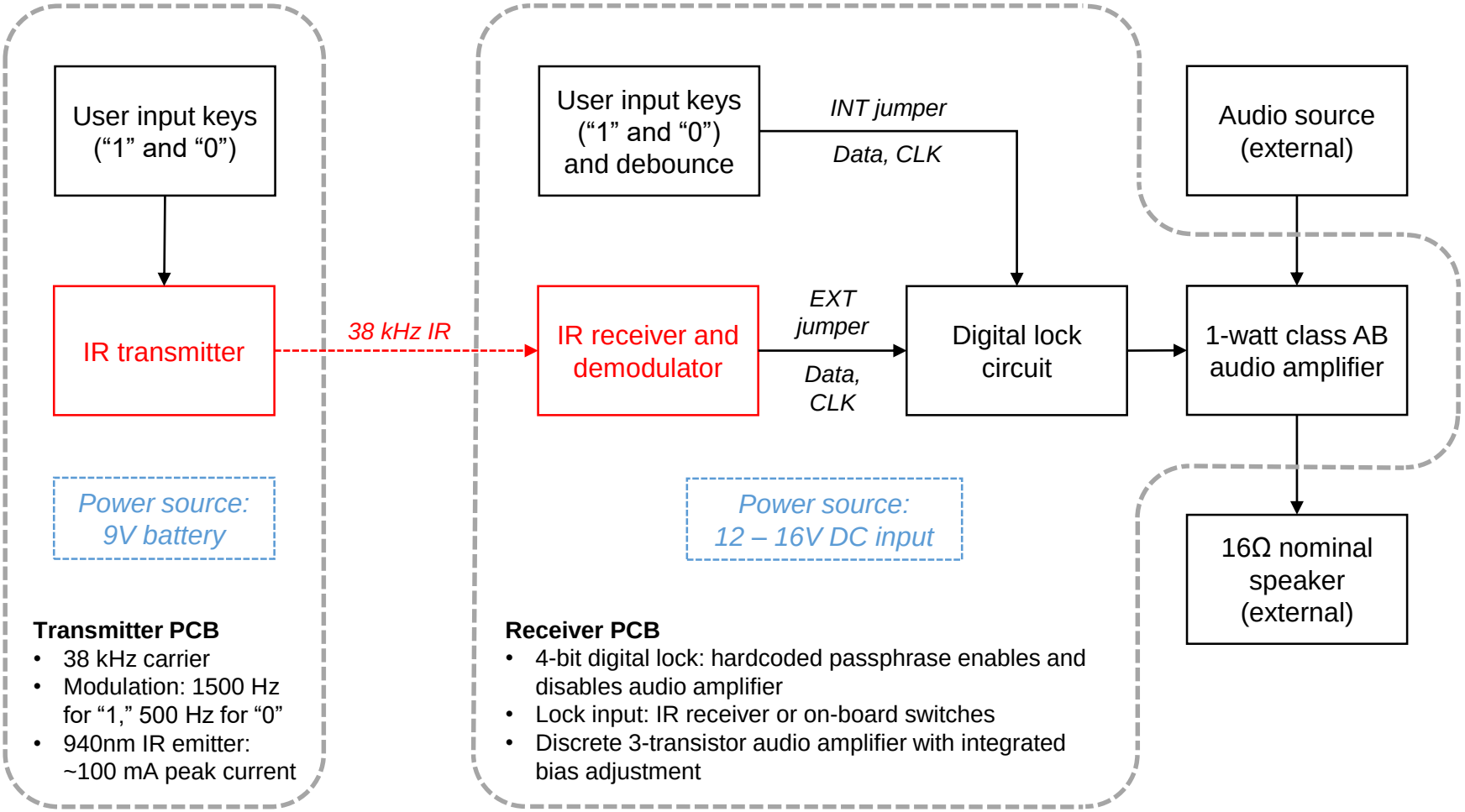
Design project overview

Project description

- **Goal:** Understand how simple blocks can be combined to make a complex system

- **Function:** Infrared (IR) remote-controlled audio amplifier
- **Main components**: power supply, digital lock, audio amplifier, IR receiver, IR transmitter (separate PCB)
- IR receiver/transmitter are optional, only if time allows

Block diagram

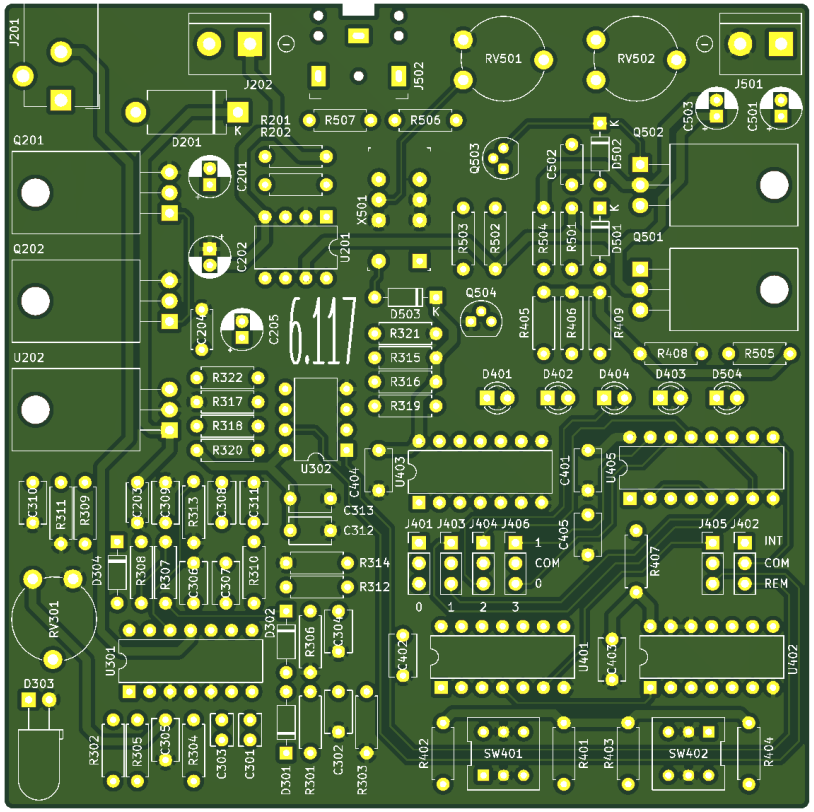


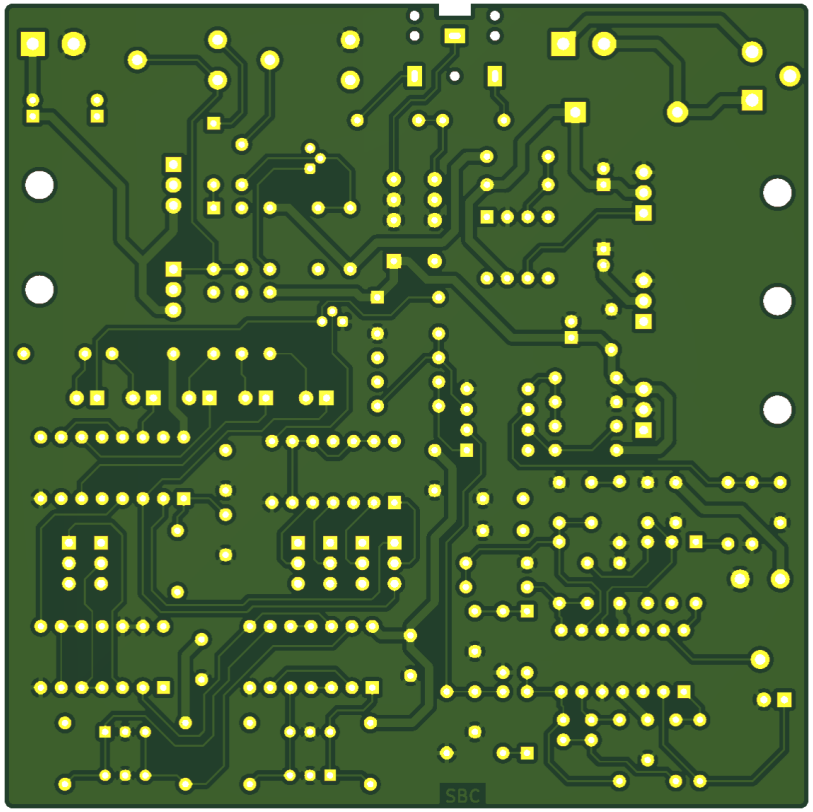
Project schedule

- **Lab 3 (1/22 – 1/23)**: Power supply and digital lock
- **Lab 4 (1/27 – 1/28)**: Audio amplifier
- **Lab 5 (1/29 – 1/30)**: IR receiver/transmitter (optional), finish any remaining sections

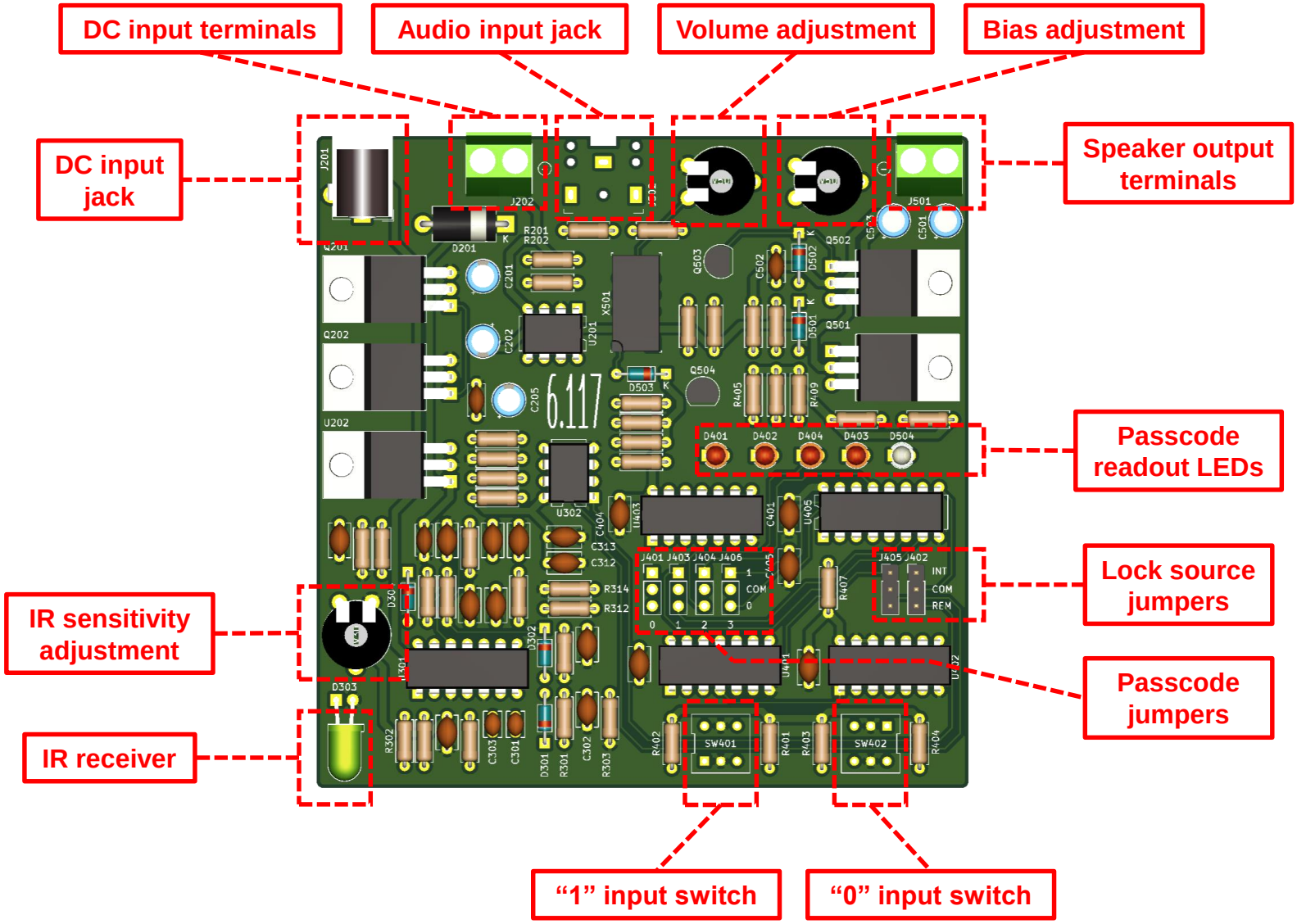
- Project not graded for completeness, only required to attend all lab sessions

Receiver PCB

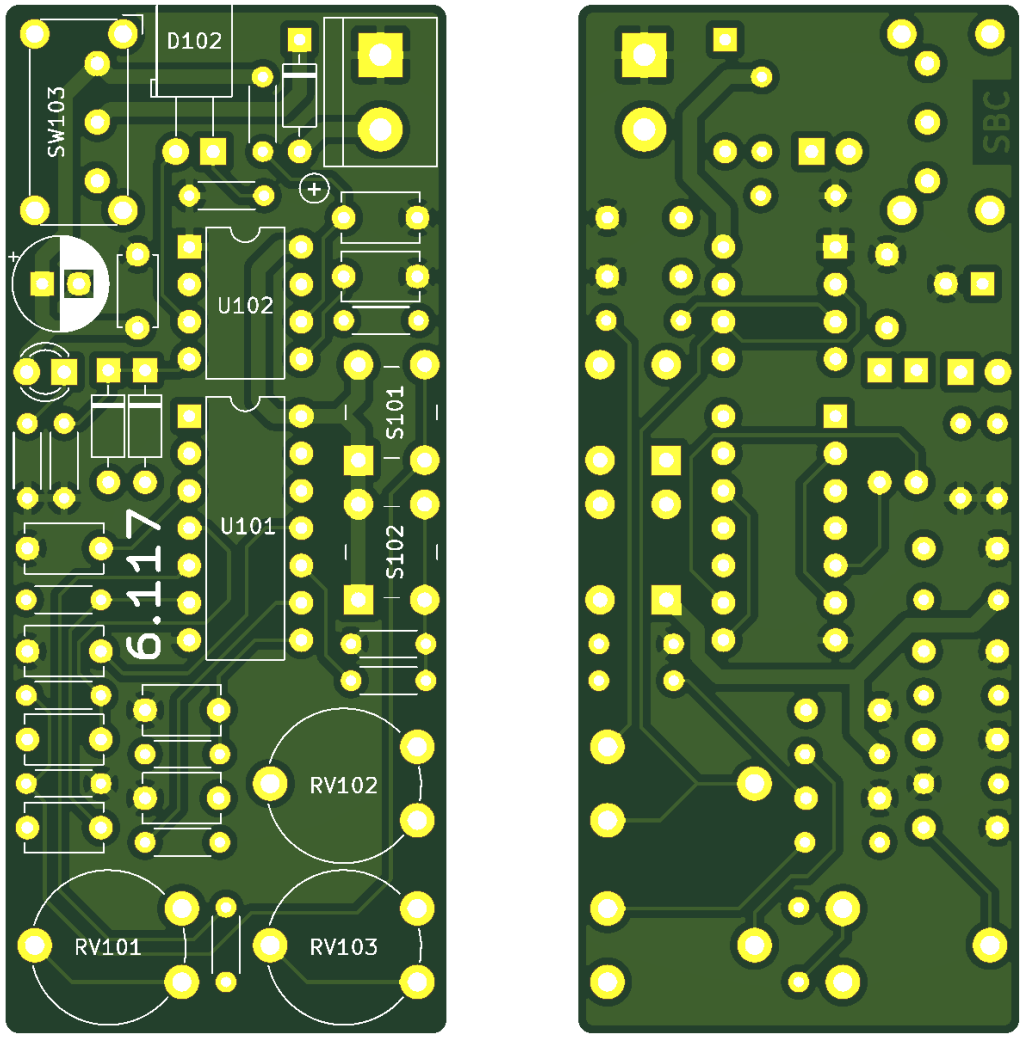




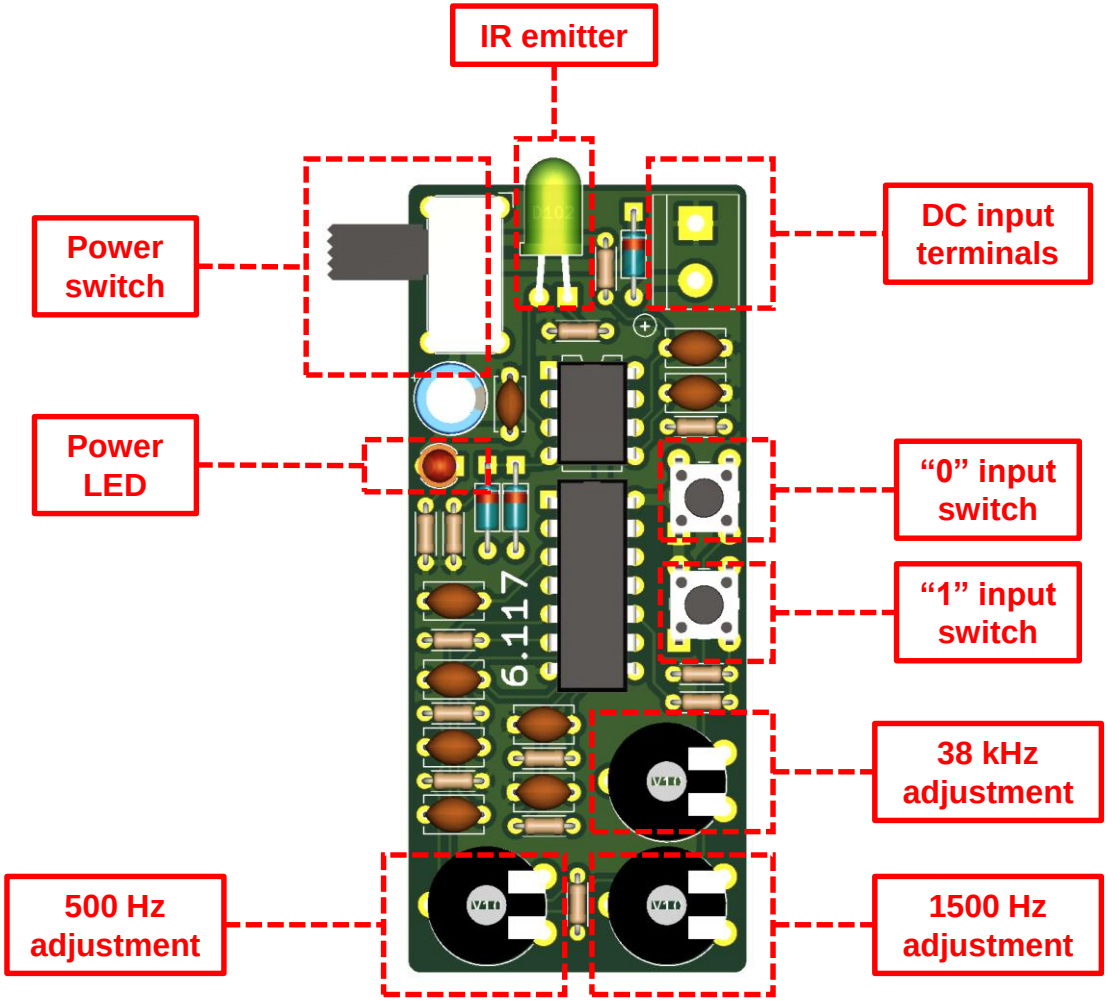
Receiver PCB



Transmitter PCB



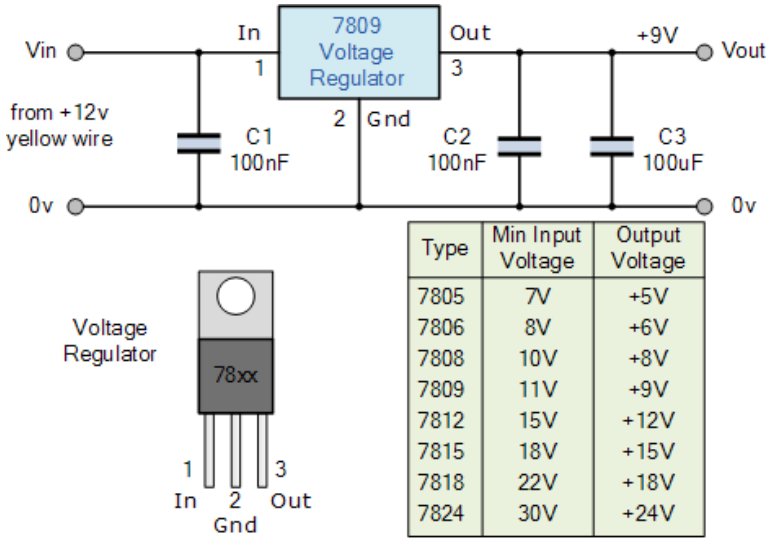
Transmitter PCB



Lab 3 overview

LM7805 Voltage regulator

- **LM7805:** Integrated linear voltage regulator
- Conveniently provides supply regulation in one package
- Use **input/output capacitors** to avoid oscillation

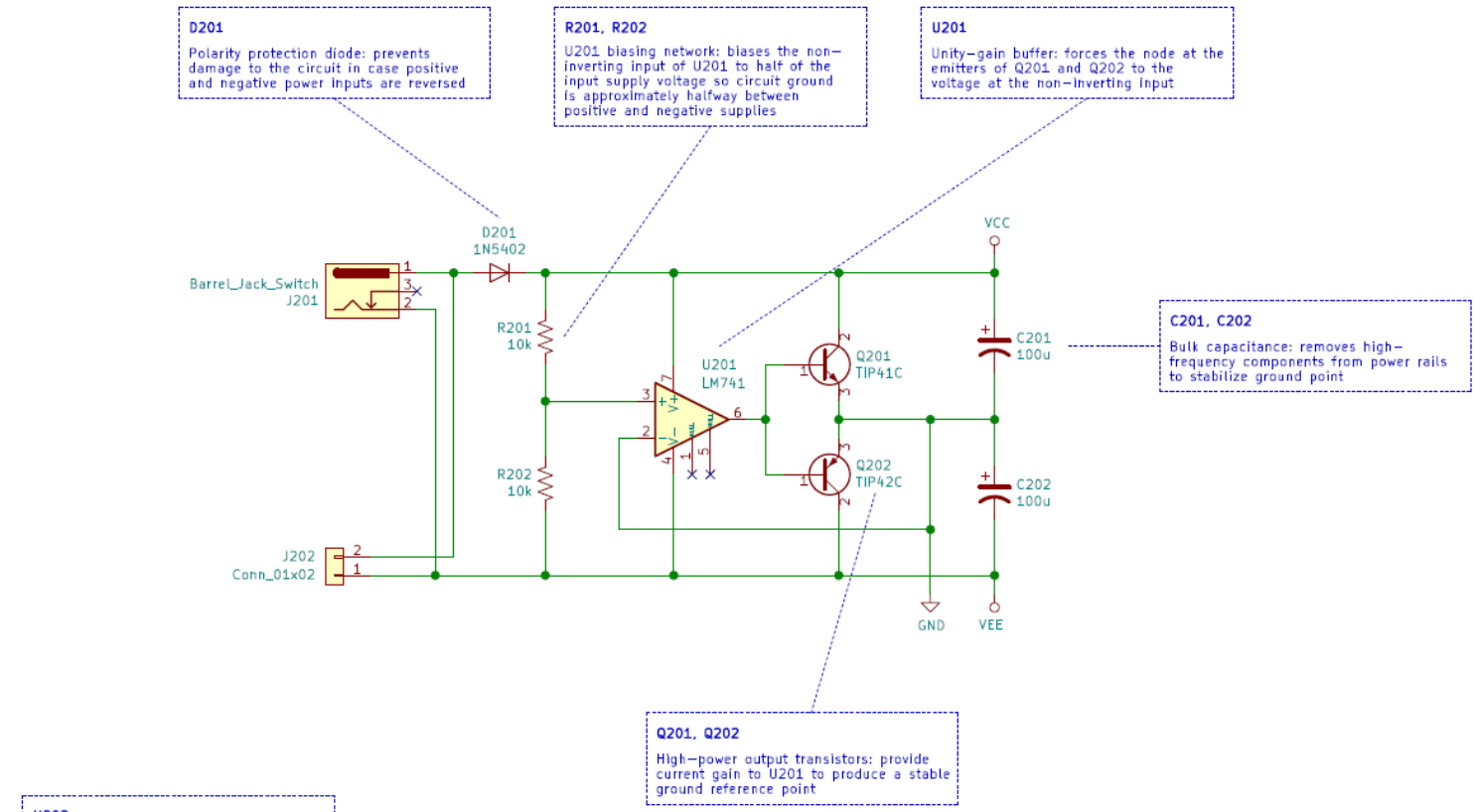


Type	Min Input Voltage	Output Voltage
7805	7V	+5V
7806	8V	+6V
7808	10V	+8V
7809	11V	+9V
7812	15V	+12V
7815	18V	+15V
7818	22V	+18V
7824	30V	+24V

Power supply circuit

- **Function:** Generates ±6V and +5V power supplies

- Uses non-inverting op-amp with push-pull output to generate **floating ground**
- Uses LM7805 to generate **regulated** +5V supply
- Contains **protection diode** to prevent damage



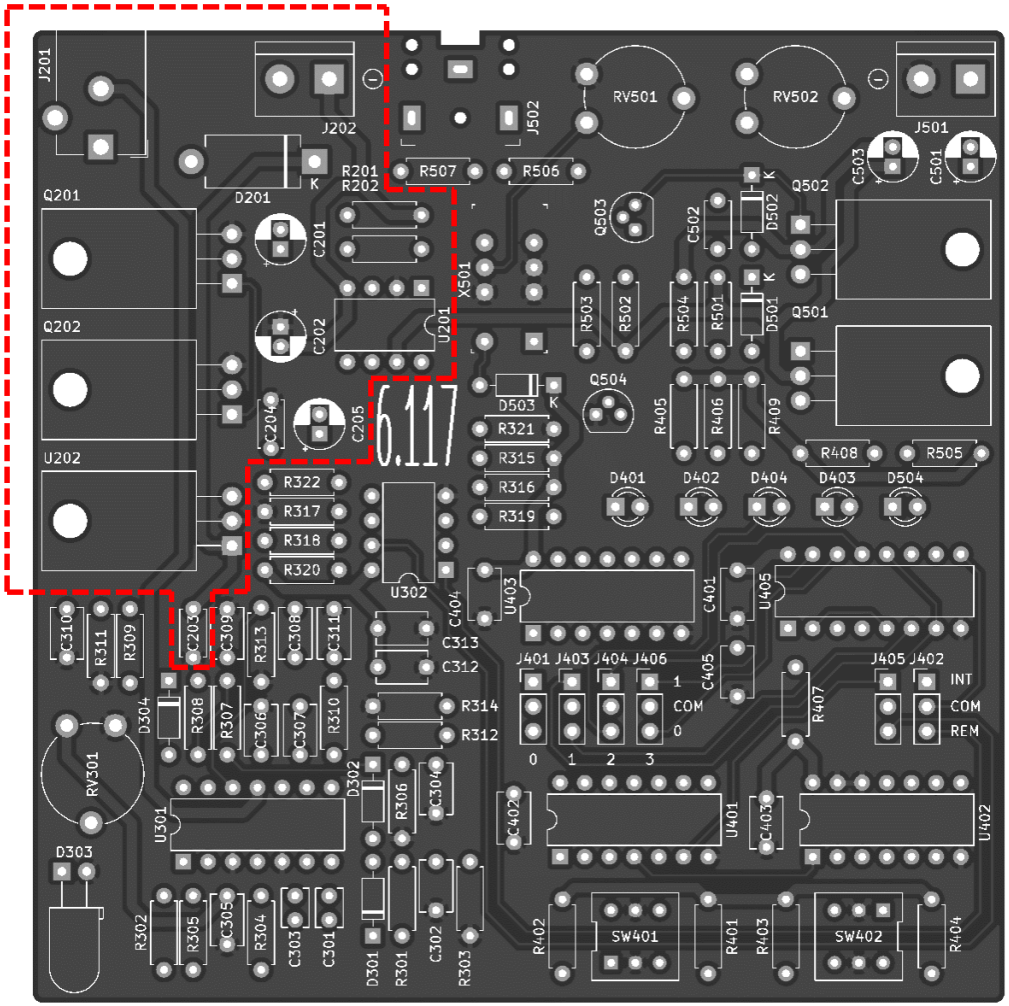
Massachusetts Institute of Technology

Sheet: /Power/
File: power.sch

Title: 6.117 Final Project: Receiver

Size: USLetter	Date: 1/15/2020	Rev: A.0.1
KiCad E.D.A. kicad (5.1.5)-3		Id: 5/5

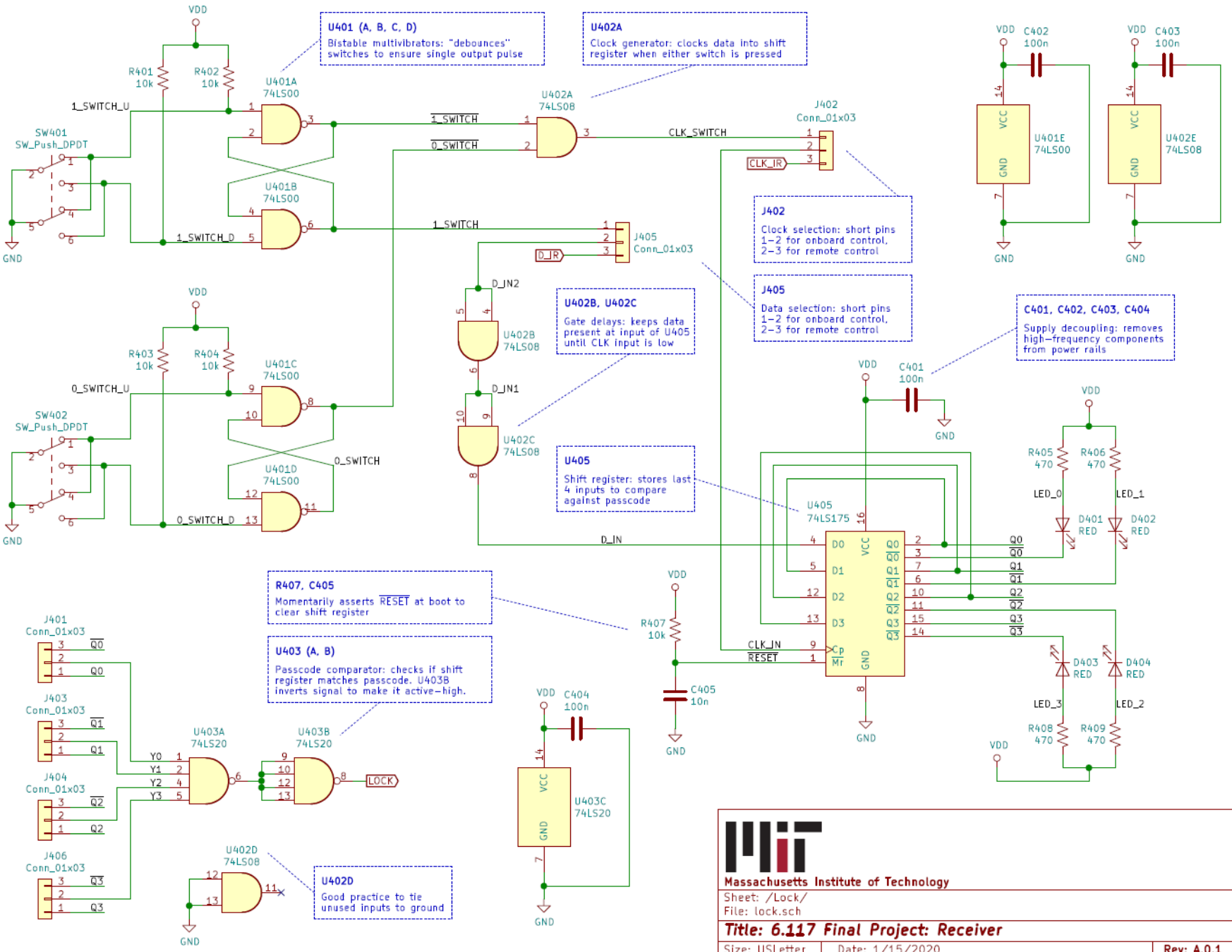
Power supply components



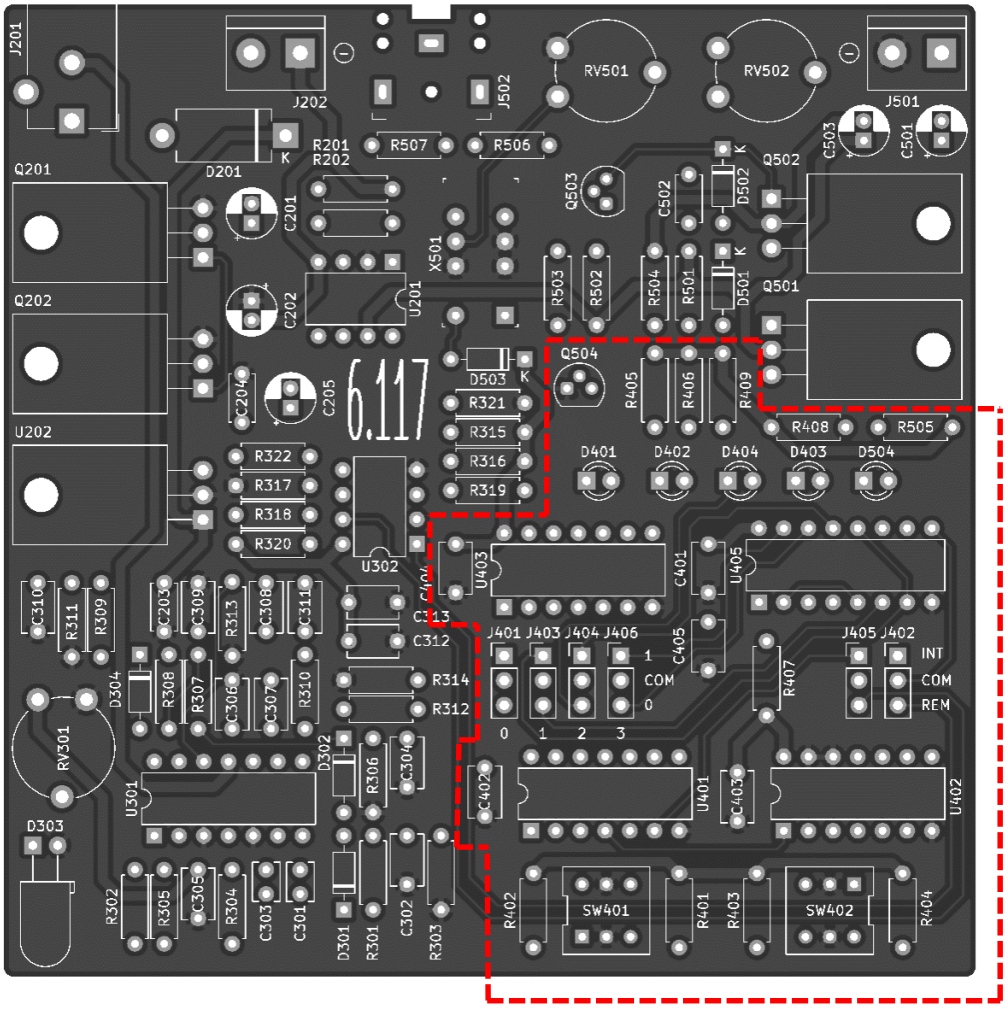
Digital lock circuit

- **Function:** “Unlock” amplifier when passcode is sent

- Uses **shift register** to keep track of received data
- Four-bit passcode hardcoded in via **jumper wires**
- Selectable input source: **buttons or IR receiver** (optional)


U401 (A, B, C, D)
Bistable multivibrators: "debounces" switches to ensure single output pulse
U402A
Clock generator: clocks data into shift register when either switch is pressed
J402
Clock selection: short pins 1–2 for onboard control, 2–3 for remote control
J405
Data selection: short pins 1–2 for onboard control, 2–3 for remote control
U402B, U402C
Gate delays: keeps data present at input of U405 until CLK input is low
C401, C402, C403, C404
Supply decoupling: removes high–frequency components from power rails
U405
Shift register: stores last 4 inputs to compare against passcode
R407, C405
Momentarily asserts RESET at boot to clear shift register
U403 (A, B)
Passcode comparator: checks if shift register matches passcode. U403B inverts signal to make it active–high.
U402D
Good practice to tie unused inputs to ground
VDD
GND
R401 10k
R402 10k
R403 10k
R404 10k
R405 470
R406 470
R407 10k
R408 470
R409 470
C401 100n
C402 100n
C403 100n
C404 100n
C405 10n
SW401 SW_Push_DPDT
SW402 SW_Push_DPDT
U401A 74LS00
U401B 74LS00
U401C 74LS00
U401D 74LS00
U401E 74LS00
U402A 74LS08
U402B 74LS08
U402C 74LS08
U402D 74LS08
U402E 74LS08
U403A 74LS20
U403B 74LS20
U403C 74LS20
U405 74LS175
J401 Conn_01x03
J402 Conn_01x03
J403 Conn_01x03
J404 Conn_01x03
J405 Conn_01x03
J406 Conn_01x03
D401 RED
D402 RED
D403 RED
D404 RED
1_SWITCH_U
0_SWITCH_U
1_SWITCH_D
0_SWITCH_D
1_SWITCH
0_SWITCH
CLK_SWITCH
CLK_IR
D_IR
D_IN2
D_IN1
D_IN
CLK_IN
RESET
LOCK
LED_0
LED_1
LED_2
LED_3
Massachusetts Institute of Technology
Sheet: /Lock/
File: lock.sch
Title: 6.117 Final Project: Receiver
Size: USLetter
Date: 1/15/2020
Rev: A.0.1
KiCad E.D.A. kicad (5.1.5)-3
Id: 3/5

Digital lock components

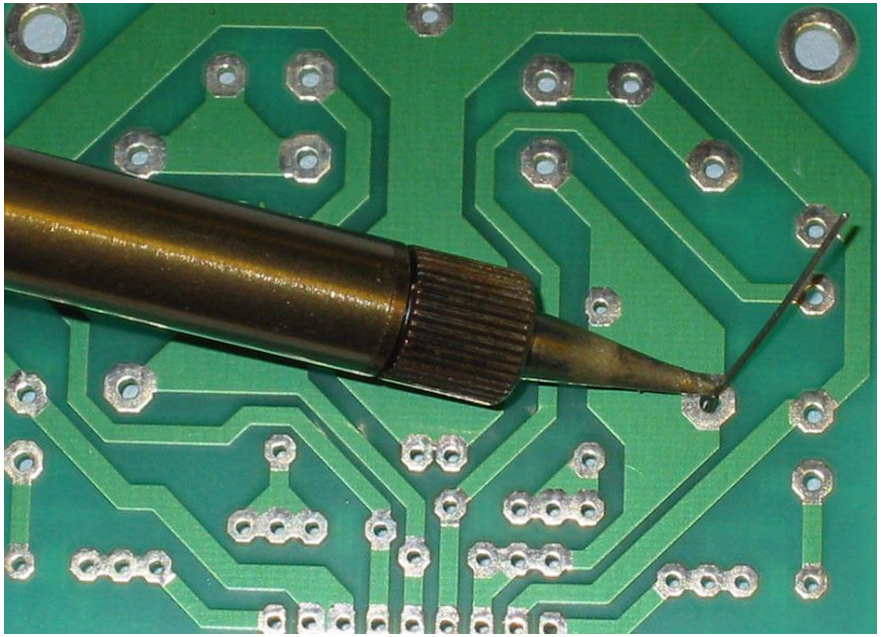


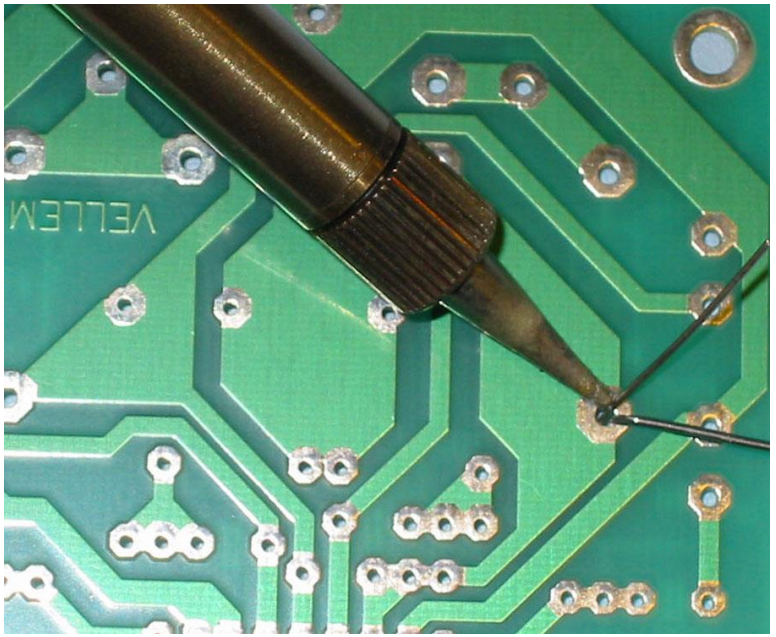
Assembly

- Fold leads **under PCB** so they don’t fall out
- Solder from **short to tall** components
- Attach power transistors with **screws/nuts** and thermal pad before soldering
- Ask someone to check your PCB **before soldering**

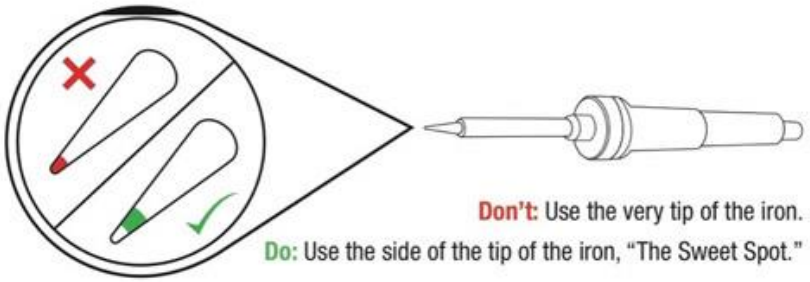
Soldering technique

- Apply the soldering iron to the **component** first, then feed the solder into the joint
- Some components take longer to heat than others





Soldering technique



Don't: Use the very tip of the iron.

Do: Use the side of the tip of the iron, "The Sweet Spot."

Do: Touch the iron to the component leg and metal ring at the same time.

Do: While continuing to hold the iron in contact with the leg and metal ring, feed solder into the joint.

Don't: Glob the solder straight onto the iron and try to apply the solder with the iron.

Do: Use a sponge to clean your iron whenever black oxidization builds up on the tip.

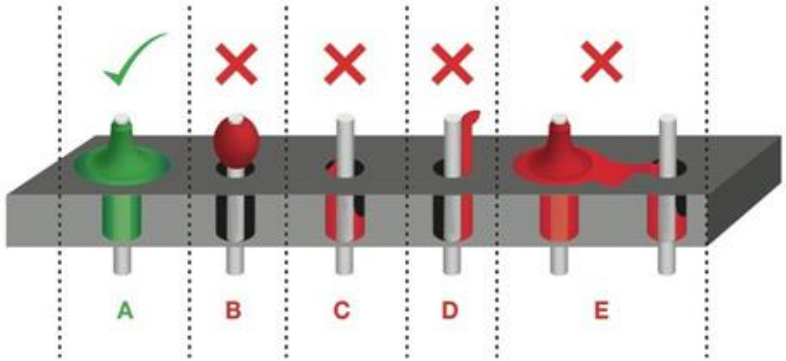
A Solder flows around the leg and fills the hole - forming a volcano-shaped mound of solder.

B Error: Solder balls up on the leg, not connecting the leg to the metal ring.
Solution: Add flux, then touch up with iron.

C Error: Bad Connection (i.e. it doesn't look like a volcano)
Solution: Flux then add solder.

D Error: Bad Connection...and ugly...oh so ugly.
Solution: Flux then add solder.

E Error: Too much solder connecting adjacent legs (aka a solder jumper).
Solution: Wick off excess solder.



Soldering video