Relay Logic

Lab 01A

6.S188 IAP 2026

Labs for 6.S188 are pretty easy-going. It is IAP and you're taking this for general interest. There are no checkoffs. There will tend to be some schematics, descriptions, instructions, and then it is up to you. We've put some good things to try / think about here, but feel free to explore, play, live, laugh, love, and follow wherever your curiosity leads you.

Parts should be in the lab space, though cheapies/passives (like resistors, caps, etc. . .) may need to be harvested from the 5th floor stock room. Adam and Joe are more than happy to help if we're around and free or doing office hours (see calendar on front page). Also. . . if you see another student working on stuff, bug them if you need help or if you want to offer help! The point of these labs is to appreciate electrical phenomena.

The Predicament

As mentioned in lecture, the foundations of ealy digital electronics were established when we didn't have transistors. Instead, early theory and much of our modern foundation was achieved during an era when electromechanical relays were how we did everything. Let's talk a little bit about those.

Inductors

For quite a few decades, most electrical equipment was reliant on electromagnetic sensors and actuators. At the core of these devices were coils, which we usually call "inductors" in circuit land. Once we had figured out how to make wire, it was only a matter of time until inductors were developed (likedly when some nervous fidgety person was twisting wire around their finger and noticed interesting behaviors).

Figure 1: A inductor in real life

Figure 2: A inductor in schematic form

As 8.02/Physics II should have taught you, an inductor is nothing more than wire run in a certain number of loops. When current runs through any wire, you get a magnetic field around it. All a coil is doing is stacking/packing that magnetic field from a lot of wire into a relatively tight space.

If you have a coil, the magnetic field B you can generate is roughly based on the equation:

$$B = \mu_0 \frac{N}{l} \cdot I$$

where μ_0 is the permeability of free space (universal constant related to c of $1.256 \times 10^{-6} \mathrm{T} \cdot \mathrm{A}^{-1}$), N is the number of coils, and l is the length of the coil, and I is the current through the coil.

From the electrical side, you can also get the famous inductor equation:

$$L = \mu_0 \frac{N^2 A}{l}$$

where A is the cross-sectional area of the coil. The inductor equation is less concerned with the magnetic field that is generated from a coil and more how it behaves from the perspective of the electrical circuit that is driving it.

Sometimes, the coil will have a chunk of material through its core since this can greatly increase the amount of field and inductance you can get per unit volume (the inductor in the first figure above is a cored inductor, for example). In that case, the field equation becomes:

$$B = \mu_r \mu_0 \frac{N}{l} \cdot I$$

and the inductor equation will be modified like so:

$$L = \mu_r \mu_0 \frac{N^2 A}{l}$$

In both cases, the μ_r term is greater than unity (often much, much greater than unity) based on the core material. A coil/inductor with a core will usually be drawn like this:

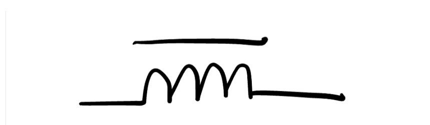

Figure 3: The schematic symbol of an inductor with a ferrous core

The fact that the inductance L of a coil can be based off of this core material, also provides a means of conveniently adjusting the indutance by varying the fraction of the core that is air/free-space or the ferrous core material. If you ever get the chance to open up a pre-2000's-era radio up and seen the little metal boxes with weird flathead screw tops like shown below, these are tunable inductors (inductors with adjustable values of L...the complement of trimmer capacitors):

Figure 4: Variable Slug-Tuner inductors

But we are digressing a little bit. We're thinking of coils as linear circuit elements (which can be useful), but is beside the point for right now. . . we will return to inductors and transformers in the future. . . instead let's get back to thinking of them as devices that allow us to induce a magnetic field based off of applied current.

If you have a coil, just by making electric current run through it you can make a strong and reliable magnetic field, you can set things nearby to be influenced by that magnetic field. We've all played with magnets. They can make things move.

What could you move? Probably a piece of metal. And we can do a lot by moving pieces of metal: we could ring a bell, or stamp a piece of metal, or unlock/lock a door, or push a diaphram that moves air (audio speaker). All of these are coils inducing motion towards various ends. The possibilities are endless.

Figure 5: A solenoid, an example of a coil with a job to do (move a metal piston on the inside)

A Reed Switch

Another thing you could do with the magnetic field is open or close an electrical switch. Think about it. . . a switch is just metal. . . I mean it has to be in order to conduct. Metal can be ferrous. What if you set up your switch so that you could open or close it based on the magnetic field? Reed switches are switches designed just for this purpose:

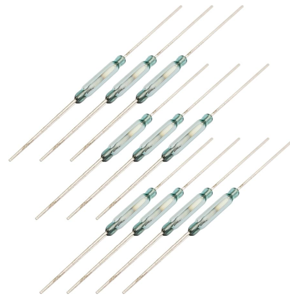

Figure 6: A bunch of reed switches

The symbol for a reed switch usually looks something like this:

Figure 7: The schematic symbol for reed switch

A reed switch is comprised of a pair of ferromagnetic contacts that have just a little bit of overlap longitudinally and are separated by just a little bit of a gap transversally. The pieces of metal have a bit of flex in them. In their normal state, they do not touch, and therefore from one contact to the other there is no conduction. However, if a small magnetic field is exposed along the axis of reed switch, the two flexible contacts straighten out and make contact. In the process, from one electrode to another, you can get conduction.

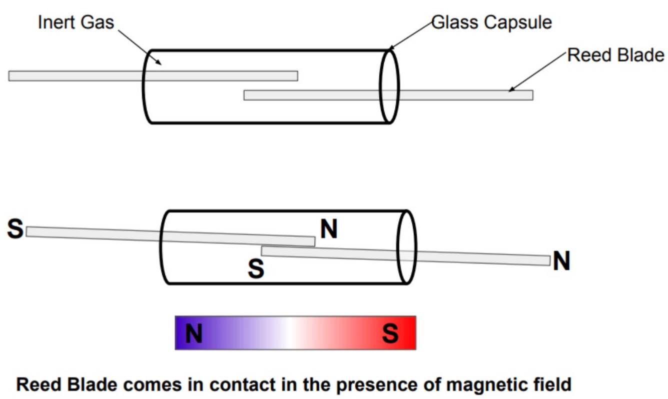

Figure 8: How a reed switch works. Taken from here

There is also a cool video showing a reed switch under a microscope at https://en.wikipedia.org/wiki/File:Reed_switch.ogv

We have some reed switches in class if you want to look at them. They look slightly different from the ones above since they're in a more durable plastic and aren't transparent, but same difference. Feel free to grab one and hook it up to the ohm/continuity setting on one of the multimeters. Normally you'll see them open, but if you put a magnet near it, it'll close (and the meter will read 0 Ohms/short). Very cool. Believe it or not, reed switches and magnets are still used to this day in some low-cost door/window sensors (though they are getting to supplanted by devices that rely on the Hall Effect more and more).

Relays

OK, so now we have two cool things. We have a way to create a small magnetic field (an inductor), and we have a thing that will make potentially meaningful mechanical movement in the presence of a small magnetic field (a reed switch). Those two things sound like they were born to be put togther, so let's do that, and ta-da, we make a new thing called a **relay**:

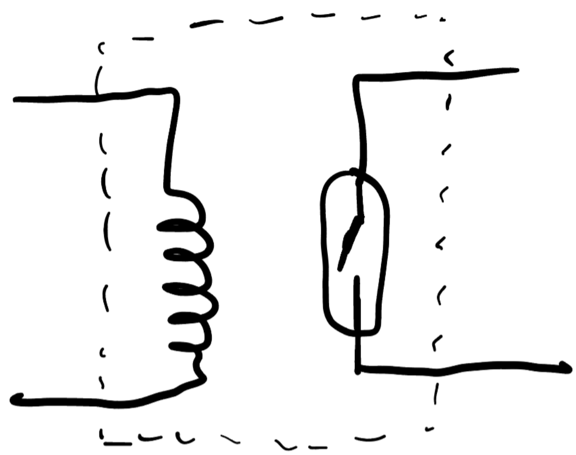

Figure 9: Put a coil and a reed switch next to one another…

Now when current flows through the coil, it makes a magnetic field that causes the switch to close. You can have different types of relays, but among the simplest is a **Single-Pole-Single-Throw (SPST)** relay, which is what's shown above, and here's what it looks like in real life:

Figure 10: A modern relay

Another very common (and arguably more convenient) relay, which we'll be using in our labs, is a **Single-Pole-Double-Throw (SPDT)** relay. Instead of opening and closing a switch, an SPDT relay causes our little switch to move between two separate contacts. This gives us a three-terminal output, where one terminal (the "Common" terminal) is normally connected to a terminal called "Normally Closed" (or "NC" for short), but in the presence of a magnetic field, the Common terminal instead gets connected to a terminal called "Normally Opened" (or "NO" for short).

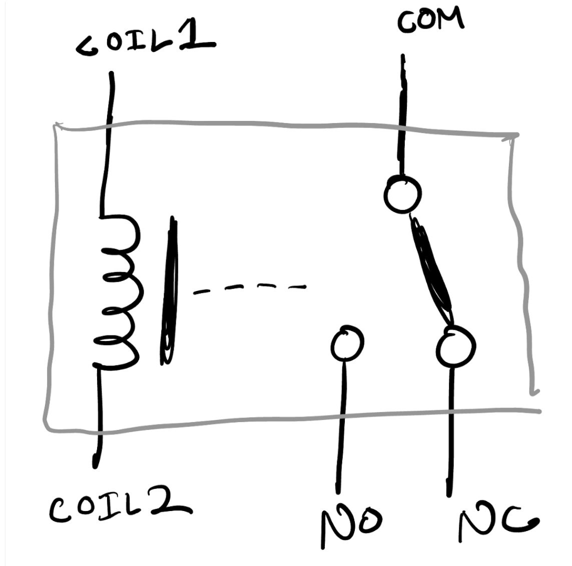

Figure 11: The SPDT Relay

We have a pile of these type of relays on conveniently labeled breakout PCBs for easy usage in a breadboard. You can use them to build the circuits at the end of the lab.

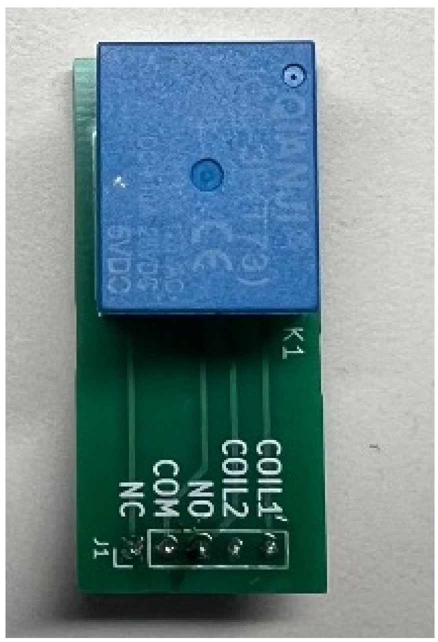

Figure 12: A SPDT relay on a breakout board

There are, of course, more complicated relays, Double-Pole-Double-Throw (DPDT), Triple-Pole-Triple-Throw (TPTT), etc. . . we'll just use SPDT ones for now.

The Behavior

On paper, a relay seems useless… we have an electrical signal… we convert it to magnetic/mechanical and then go back into electrical. But this is misleading. That decoupling allows us to manipulate physics so that the incoming electrical signal can influence the outgoing electrical signal in ways that are advantageous.

Let's take the relays we have in lab… if we call the current through the coil i_{in} and assume we have 10A flowing into the common terminal, like so:

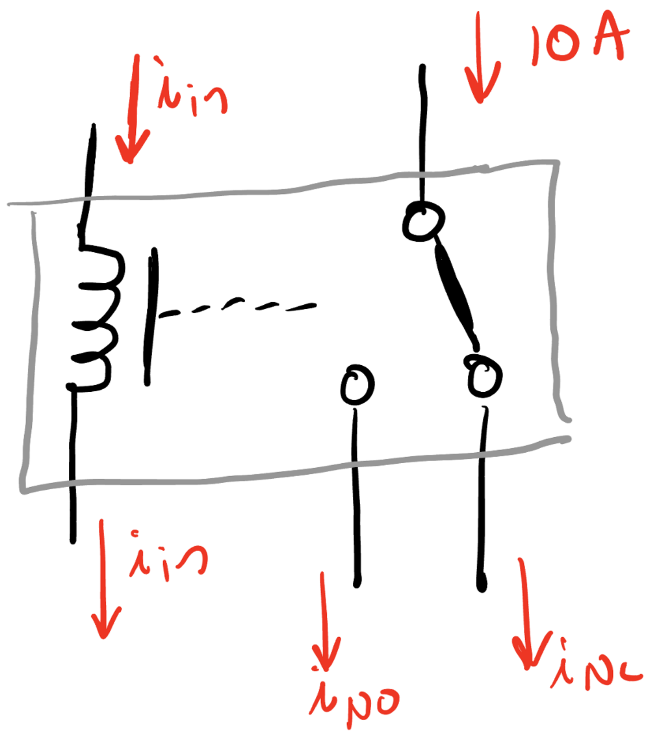

Figure 13: The SPDT Relay

Then we can get two types of input-output behaviors from a relay like this. From the Normally Open terminal, you get this:

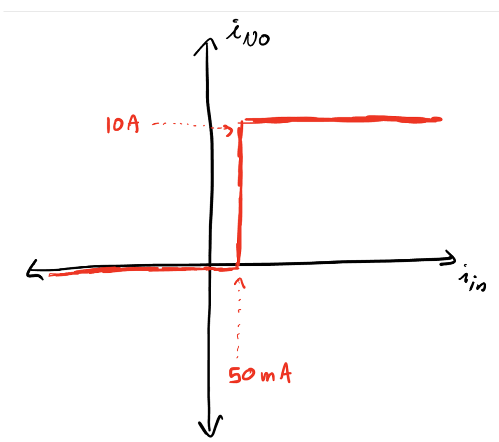

Figure 14: The Input-Output relationship of a relay via the normally-open ("NO-comm") connection

From the Normally Closed terminal you get this:

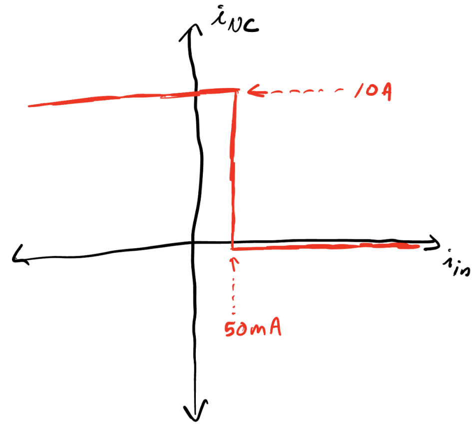

Figure 15: The Input-Output relationship of a relay via the normally-closed ("NC-comm") connection

This is but one way to evaluate and plot the input-output relationship of a relay, but by looking at it as input/output current, the non-linearity and amplification aspect should become readily apparent. These relays have a pull-in current of about 50 mA, meaning when about 50 mA is flowing through the relay's coil, the magnetic field becomes strong enough to flip the switch of the relay. Below that (with some hysteresis mixed in), the switch flips back. This transition point is is sharp, electrically speaking. . . with basically infinite slope as far as we are concerned. This is *highly non-linear*.

Also because the switch can handle a lot of current, but the coil doesn't need too much, the relay is allowing a low-amplitude signal to create/control a high-amplitude signal. This is *amplification*. (with a 10A supply, this circuit could be said to have a current gain of around 200 in fact).

These two attributes (lots of gain and lots of non-linearity), are foundational. Combined, they are enough to do quite a lot.

Three Circuits

Using these relays we could build a historically appropriate telegraph detector, but I think what might be more educational is to use their capabilities to create three important digital circuits:

- A NAND Gate
- A Set-Reset (SR) Latch
- An Oscillator

These three circuits form foundational elements of any useful digital system. We'll talk through building them on the next couple of pages, but maybe first you should look at this other document that talks through some of the basics of putting together circuits. We'll power all of the circuits with the USB connectors described in that document.

Circuit 1: NAND Gate

On a breadboard, using two relays, two button switches, an LED and a resistor, build the following circuit:

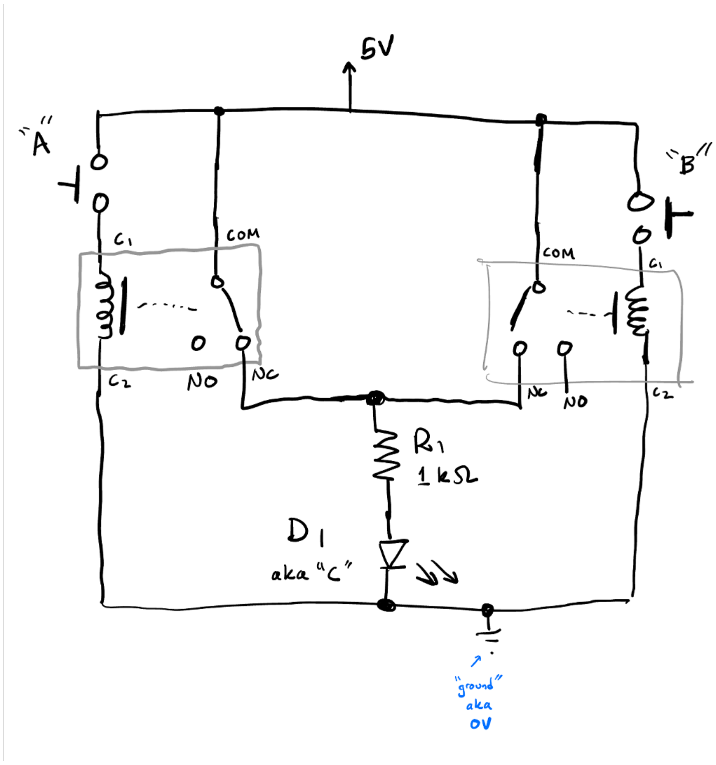

Figure 16: A relay-only NAND gate

If we think of the buttons as digital inputs (not a hard stretch), with unpushed being "0" and pushed being "1", and use the LED as an output "D" (where the LED being on means "1" and off means "0"), we can get a rough pattern of its behavior. We can derive the following:

- $A = 0, B = 0$: $C = 1$
- $A = 0, B = 1$: $C = 1$
- $A = 1, B = 0$: $C = 1$
- $A = 1, B = 1$: $C = 0$

This pattern, is that of a NAND gate. NAND gates (and their siblings NOR gates) are each *functionally complete*. With them, you can build all other digital logic (both combinational and, if you add in feedback paths, sequential). You may think that would be an exhaustively massive undertaking that is only done theoretically, but that's actually wrong. The entire computer that controlled the Apollo moon lander in 1969 was built using on NOR gates for quality control reasons (of course by then logic was being done with transistors, but the fundamental idea remains the same. We'll get to that in a few weeks). So this is a pretty powerful circuit.

Before moving on, let's spend a minute to think and make sure we understand: given the circuit we've built above, why does this pattern of inputs and outputs make sense?

Circuit 2: Set-Reset Latch

The next circuit to build is a slight deviation on the previous. Instead of having the symmetry of the NAND gate, we'll change a few wires (shown in blue) and relable the two button inputs as "S" (for "Set") and "R" (for "Reset"):

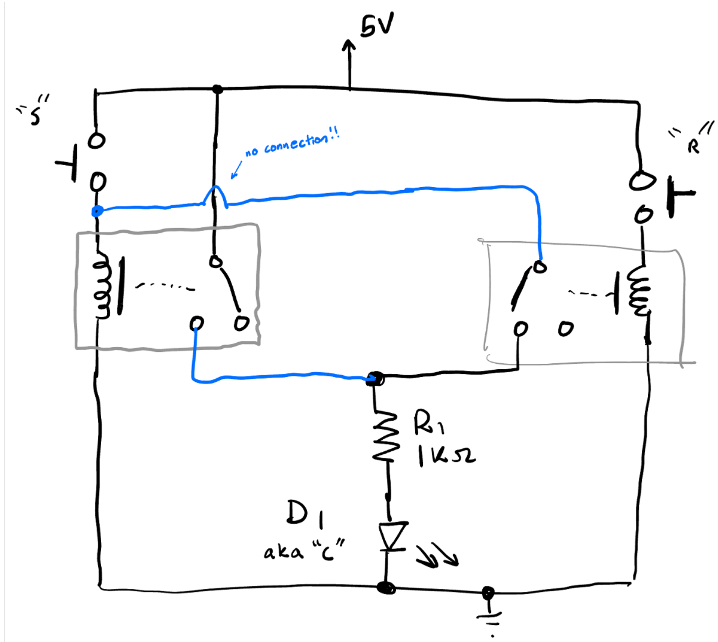

Figure 17: A relay-only Set-Reset Latch

The circuit works by starting as drawn in the "reset" state. If a user comes in and pushes the Reset button, nothing changes. If a user pushes Set, however, the button, allows current to flow through relay K_1's coil, causing its switch to flip towards the Normally Open connection. Since K_1's Common is connected to 5V, the resistor/LED combo connected to the Normally Open terminal will light up. Importantly, the Normally Closed connection of K_2 is also connecte here, and the Common of K_2 connects back to the top side of K_1's coil. This means the coil on K_1 is getting 5V supplied via two paths: The Set button and the path through K_1 and K_2. Now, even if the Set button is releaed, one of these paths is still existing and the circuit will stay "latched" in this state ("Set") until the Reset button is pushed, breaking this path.

The truth table for this circuit is the following:

- $S = 0, R = 0$: $Q =$ Latch
- $S = 0, R = 1$: $Q = 0$
- $S = 1, R = 0$: $Q = 1$
- $S = 1, R = 1$: $Q = 1$

The term "latch" in the truth table here refers to previous value of Q. In other words, this truth table is reliant on history... which means, the circuit is capable of remembering. This... is a big deal. With just two relays, we can have a circuit remember information. This is stateful. Put a few of these together and you can even make a D Latch... then a few more and you could have a D flip-flop, a clock-edge-triggered memory element that is the foundation of all modern sequential digital design. But we'll talk a lot more about those things over the next couple of weeks. For now, let's look at one more cool circuit.

Circuit 3: Oscillator

The previous circuit involved the left relay activating and then holding itself open until a second relay (controlled by a button) breaks the feedback path. Now let's build a circuit that behaves kind of in a similar way, but which kind of pushes its own button, turning itself off and on again in an endless, Kafkaesque loop.

Let's go ahead and build the following circuit, which consists of two little resistor/capacitor ("RC") combinations and a couple of relays. Don't worry about using exactly 27Ω for R_1 and R_2, but something around that range is likely to work.

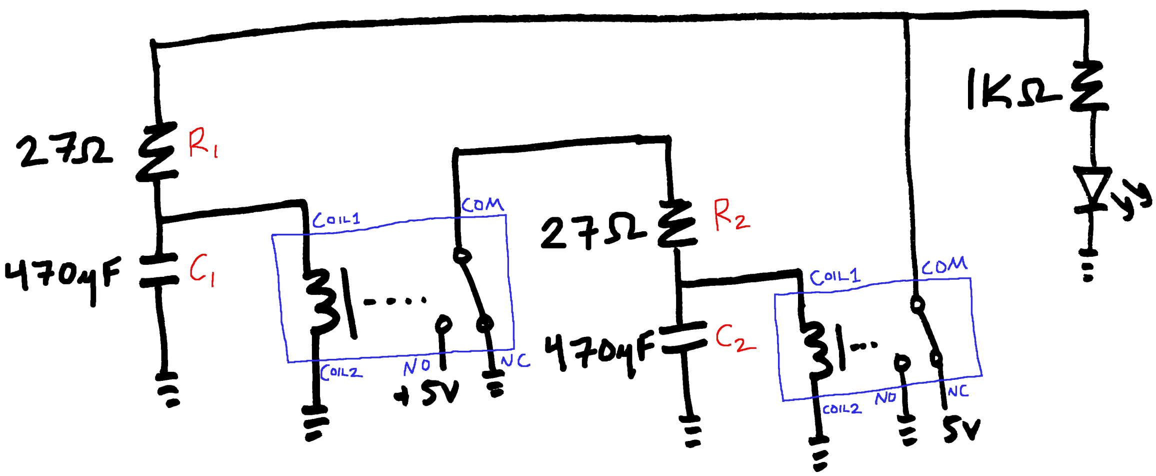

Figure 18: An oscillator circuit

How is this thing working? Well, that's a great question. There are some complicated interactions, but to simplify a bit, when the Common terminal of the right-most relay is connected to 5V (which happens when it's off), the voltage drop across C_1 will start to rise (at a rate predicted by the specific component values of R_1 and C_1). Eventually, this voltage will get big enough to cause enough current to flow through the coil of the left-most relay to turn it on.

When the left relay turns on, the voltage drop across C_2 starts to slowly build up, until it is eventually high enough to turn *that* relay on. When that happens, the voltage across C_1 now starts working its way *down* toward 0; before too long, it gets low enough that the left-side relay turns off, which then causes the voltage across C_2 to go down, until the right-side relay turns off, at which point we're back where we started and we can go around again. And again. And so on forever and ever.

While we're here, try changing the values of C_1 and C_2. As you make them bigger/smaller, what happens to the frequency at which the light blinks on and off?

If the noise gets a little obnoxious, don't worry about it. You've not done anything wrong, that's just how things go. Some people find it soothing, but you don't have to find it soothing if you don't want to.

OK, so that's all of the circuits we're going to ask you to build today, but hey, if you want to play around some more, feel free to go ahead build any of the other gates that we talked about in lecture, which might also be fun, or really any other circuits you feel like.

Reflections

A few big things to take away from these circuits. Relays on their own can provide us amplification and non-linearity. Those two attributes, together and in parts, can be used to create sufficient computation to build basically any modern electronic system. Literally you could build a neural net with them or an LLM....there's one big caveat...

Relays are slow. This arises from the fact that you're moving mass around to get behavior and that will always take some time, but also more fundamentally, relays rely on building up magnetic field to perform actions and that has a hard time scaling downward (in exponential ways like capacitance did in transistors in the latter half of the 20th century). Looking back on them from our 21st century perspective, the fact that they are stuckin the 100's of Hz response time really precluded them from doing any high-speed data processing. Their extremely digital nature doesn't actually even preclude them from being used for analog purposes (Class D amplifiers, which are how most modern systems generate audio are built around high speed digital switching and spectral filtering)...it really is just a matter of speed.

Just like neanderthals coexisted with early *Homo sapiens* for at least several thousand years, relays were widely used in many systems well into the 1950's, even though vacuum tubes were well-established by 1920. Part of this was because relays could be made to function for a very long time whereas vacuum tubes often had limited lifetimes of hundreds or thousands of hours (due to filament burnout and other things). Relays also could run on lower voltages and, amazingly, be more power efficient, which sounds comical given how we think of them today, but relative to the hungry hungry hippos that were vacuum tubes, they could be a good deal..

Interestingly enough, while not the focus of our class, relay-based logic like that shown above was how almost all computers were built up until about the end of the Second World War (~1945). In fact vacuum tubes were really only used heavily in computers for about ten to fifteen years (1945ish to 1955ish) since the sheer scale of computers made the high failure rates of tubes a really big problem. ENIAC, was the first "full" modern computer and it still had several thosand relays in some of its logic. Prior to that, however, Turing complete computers based solely off of relays were indeed a thing. The Zuse Z3 made by German scientists before and during World War 2, did all of its operations using 2600 relays, which when you think about it, isn't *that* crazy of a number. The Z3, however had a clock frequency of 5 Hz. That is not a typo. Not 5 GHz or 5MHz or 5kHz... 5 Hz. This of course was because of the compounding of the resistor-inductor time constants of all the relays driving one another... I mean our relay oscillator ran at ~1kHz with nothing in between itself. Think about relays driving relays driving relays...the LR time constants would build up quickly.

After society collapses and you need to rebuild from scratch, relay-based logic will probably be your best bet to getting back to a functioning computer. Make a coil and a magnetic-field-based switch. Hook them up like shown above into a NOR topology, repeat several thousand times, and bingo bango bongo, you'll have a computer. The clock rate will be limited by greatly by the relay inductances and parasitic resistances, but at least it will work.

Further Readings

Claude Shannon's Masters Thesis from 1938, which establashes much of modern computing, uses relays as the bridge between boolean algebra and electrical manifestations of his ideas.

The Zuse Z3 computer was recreated in the early 2000's and documented here. Also this book (which you can get through MIT libraries or email Joe since he has a copy) goes into crazy detail on the designs and operation...all of this was done with relays, keep in mind.

Moving On

In the next lab we'll start to solve the problems of relays. The first issue we'll address is their low speed using an alternative technology that gives us non-linearity but can do so at very high frequencies. Then we'll use a different tech to give us non-linearity and amplification at high frequencies.