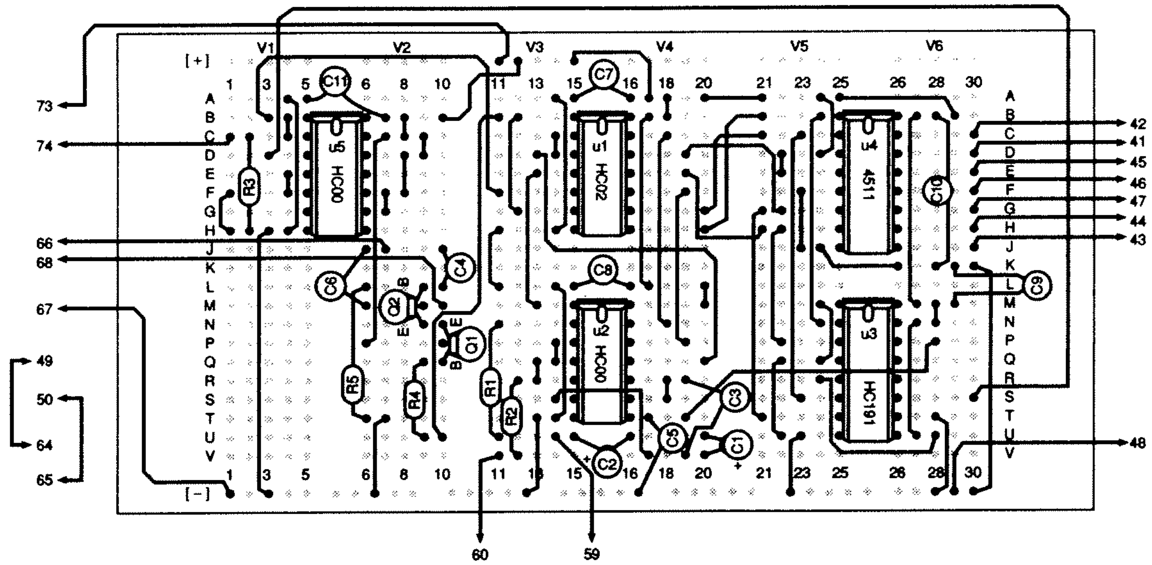
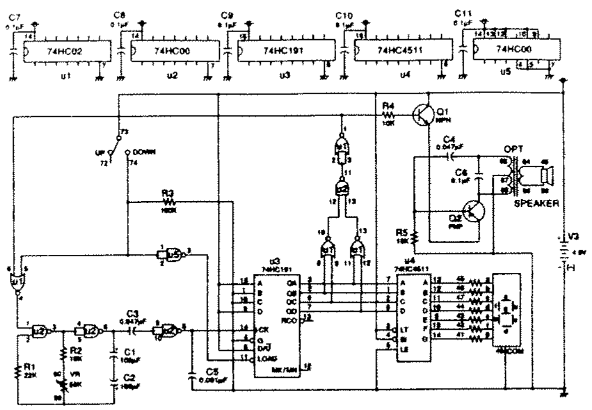
PROJECT 61. COUNT DOWN TIMER



U1	74HC02	Q1	NPN	R1	22KΩ	C1	100μF	C7	0.1μF
U2	74HC00	Q2	PNP	R2	10KΩ	C2	100μF	C8	0.1μF
U3	74HC191			R3	100KΩ	C3	0.047μF	C9	0.1μF
U4	74HC4511			R4	10KΩ	C4	0.047μF	C10	0.1μF
U5	74HC00			R5	10KΩ	C5	0.001μF	C11	0.1μF
						C6	0.1μF		

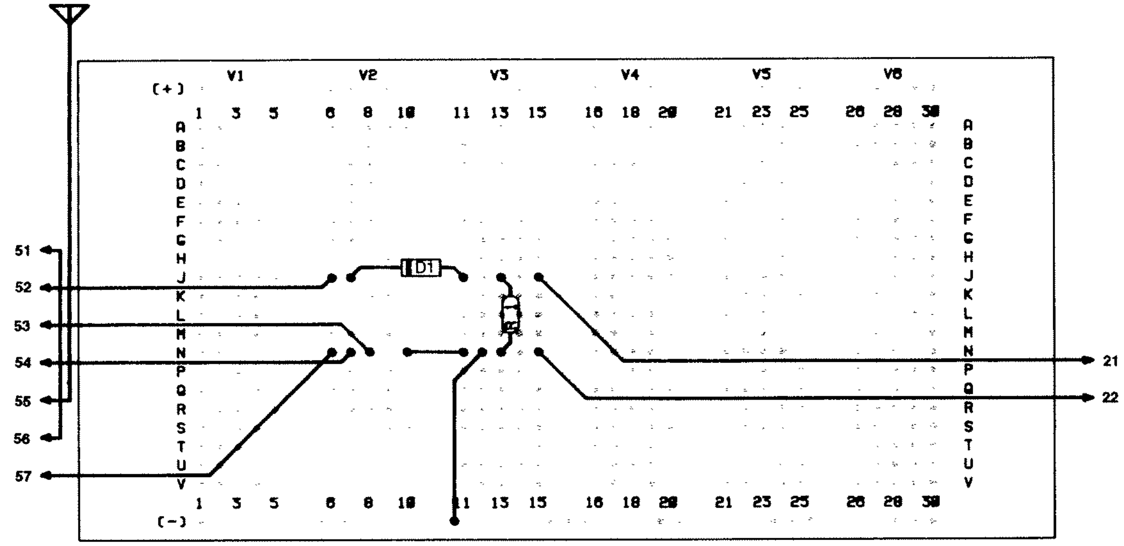
Here's simple count down timer. When you finish wiring the project, slide the **select switch** down. The **LED** display shows **9** then starts counting down. When the timer reaches 0, a beep sounds from the **speaker**.

You can adjust the pitch of counting by the **control volume**. To restart the timer, set the **select switch** up once then back to down.



5) Radio Circuit

PROJECT 62. "CRYSTAL SET" RADIO



R1	470KΩ
D1	Ge

Back in the early days of radio, crystal sets were widely used (your grandparents might remember them). This project is a more up-to-date version of the classic crystal set.

Before using this project, you need to have an outdoor antenna erected. Be sure to have an adult help you put up the antenna!

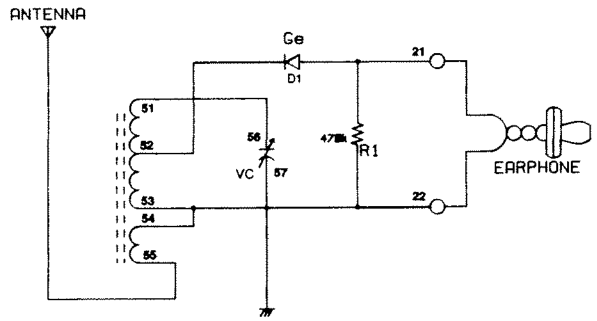
You also need a ground connection. You can run an insulated wire to a cold water pipe or to a ground rod. Have an adult help you make the ground connection.

Notice something funny about the schematic for this circuit? No, we haven't forgotten the batteries - This circuit doesn't use any. The power for this project is supplied by the radio signal itself!

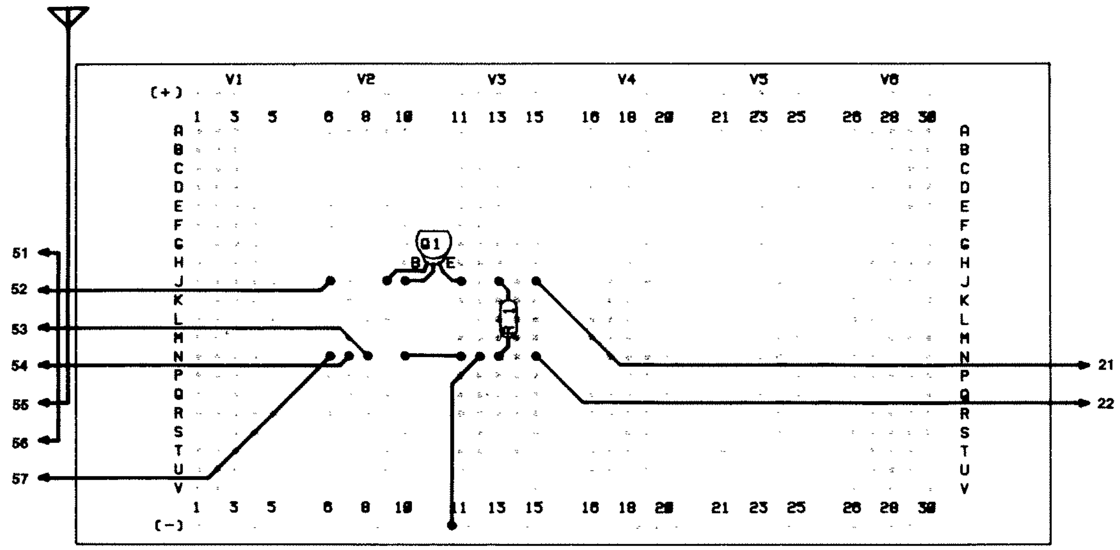
When radio waves strike the outdoor antenna, they set up small electric currents in the antenna. The lead-In wire carries these currents to the antenna coil. These currents are AC and are at different frequencies. The **tuning control** selects one set of frequencies (remember how a variable capacitor work?) and passes the current on to the diode.

The diode (as you should remember!) rectifies AC into DC. You hear the results of all this in the earphone.

This project lets you tune stations on the AM broadcasting band. Adjust the **tuning control** to find the station you want to hear. Don't be too surprised if all you can hear is one loud local station with this circuit. And if you live several miles from the nearest AM station you might not be able to hear any stations. Be sure to use an outdoor antenna for best results.



PROJECT 63. "FUNNY TRANSISTOR" RADIO



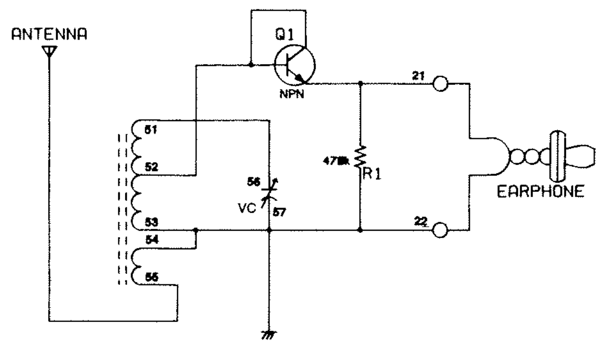
Q1 NPN R1 470KΩ

If you thought something was unusual about the schematic in our last project of "Crystal Radio Set", just take a look at this schematic. There's a transistor ... but no batteries are used!

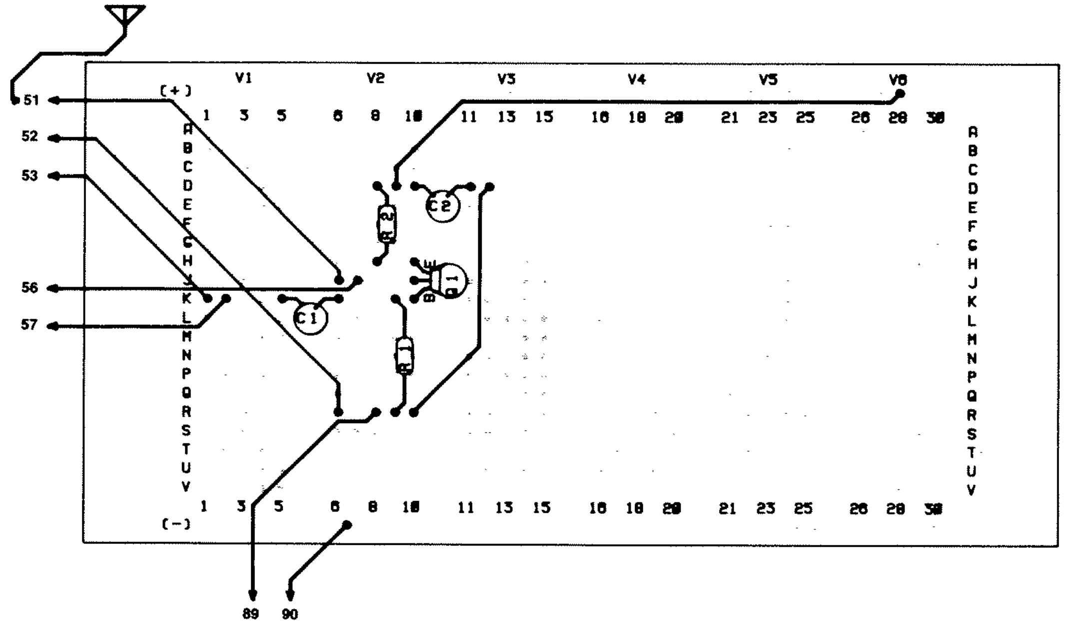
But when you finish making the wiring connection, connect the earphone to terminals **21** and **22** of your kit and listen. Adjust the **tuning control**. Hear anything? If you managed to hear stations on our last project (you have to use an outdoor antenna to make it work), you can hear AM stations on this one too.

Now put on your thinking cap: why does the transistor in this project think it's a diode?

Give up? It's because a transistor is actually two diodes back-to-back. One diode is from the emitter to base and the second is from the collector to the base. That's why in our circuits using transistors we have currents from emitter to base and another from collector to base.



PROJECT 64. WIRELESS CODE TRANSMITTER



Q1	PNP	R1	100KΩ	C1	0.001µF
		R2	4.7KΩ	C2	0.1µF

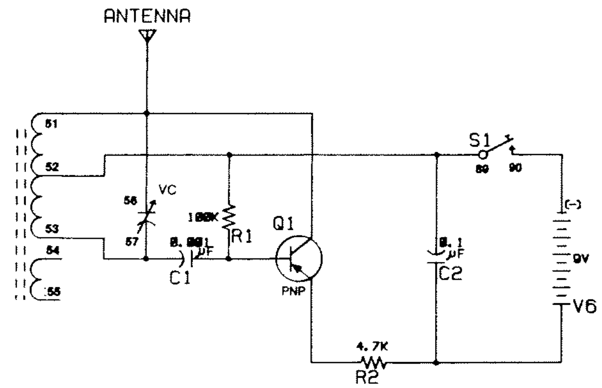
This project is a simple but effective code transmitter. The same principle is used on the communication device for the military and amateur radio operators around the world. You send the codes with the key that turns the transmitter ON and OFF.

You can use the normal AM radio to receive the code sent out by this transmitter. First tune to a weak station. The transmitter signal is mixed with the station's signal to produce an audio tone, called a "beat note." Tune the transmitter using the **tuning control** until you hear a beat note on the AM radio as you press the key.

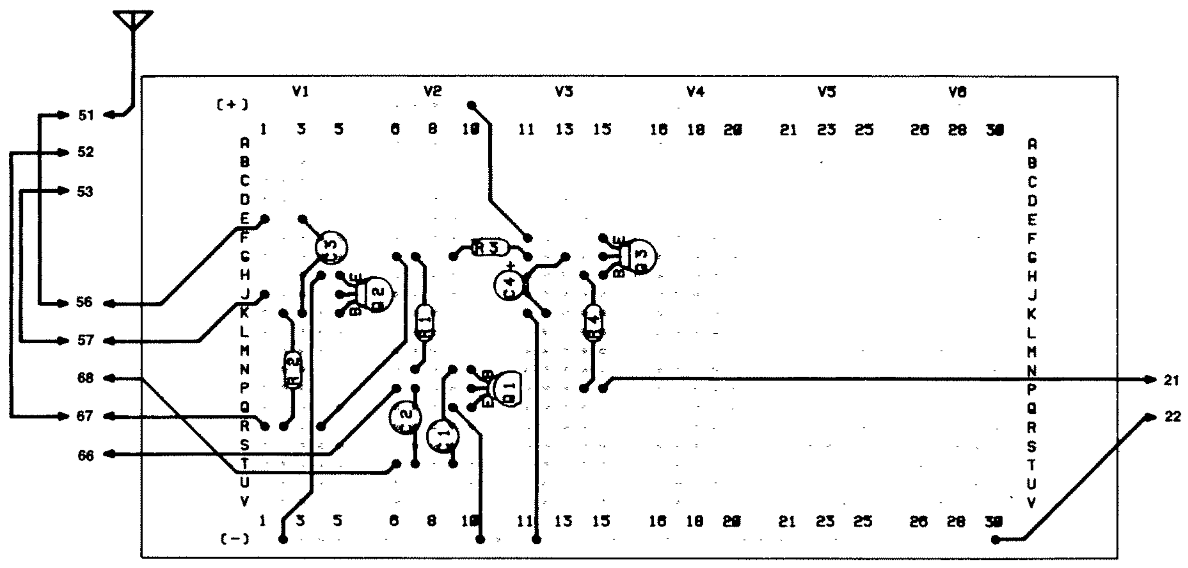
Some communication receivers have a circuit called beat frequency oscillator (BFO). With such receiver you can receive the continuous wave (CW) signal of this transmitter without having to tune to another station. The BFO "beats" with the CW signal from this transmitter and produces the tone.

Like other oscillator circuits in this kit, the quick charge/discharge of the 0.001µF capacitor causes the oscillation as you press the key.

Transmission and reception of CW signals is very efficient (much more than voice signals). Thus, during times of emergency this is the most reliable transmission. Because of this high efficiency, you might find only two or three feet of wire as an antenna can do the job -- or even with no antenna! Have fun!



PROJECT 65. REMOTE WATER LEVEL DETECTOR



Q1	NPN	R1	47KΩ	R4	10KΩ	C3	100pF		
Q2	NPN	R2	22KΩ	C1	0.047μF	C4	100μF		
Q3	PNP	R3	1KΩ	C2	0.1μF				

Here's a circuit similar to project 251, Rain Detector but with a twist: it includes a radio transmitter. This means you can check for high water levels or rain even if you're not in a place where you could hear an alarm. Just take along an AM radio and listen!

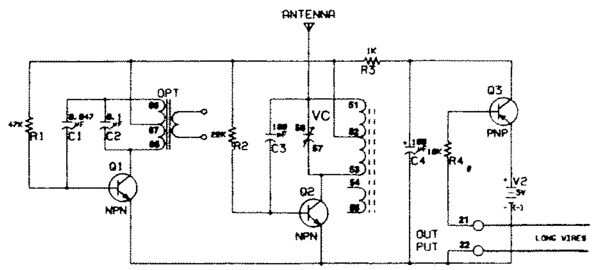
In this project, we send out a radio wave at a frequency where there are no stations, so we need to find such a spot in the AM band of 530 - 1600 kHz.

As you can see from the schematic, there's no power switch to turn this project ON or OFF. Whenever water (or some other conducting substance) allows electric current to flow from the base to the emitter of the PNP transistor, the circuit operates. You hear a tone from the radio when this circuit operates.

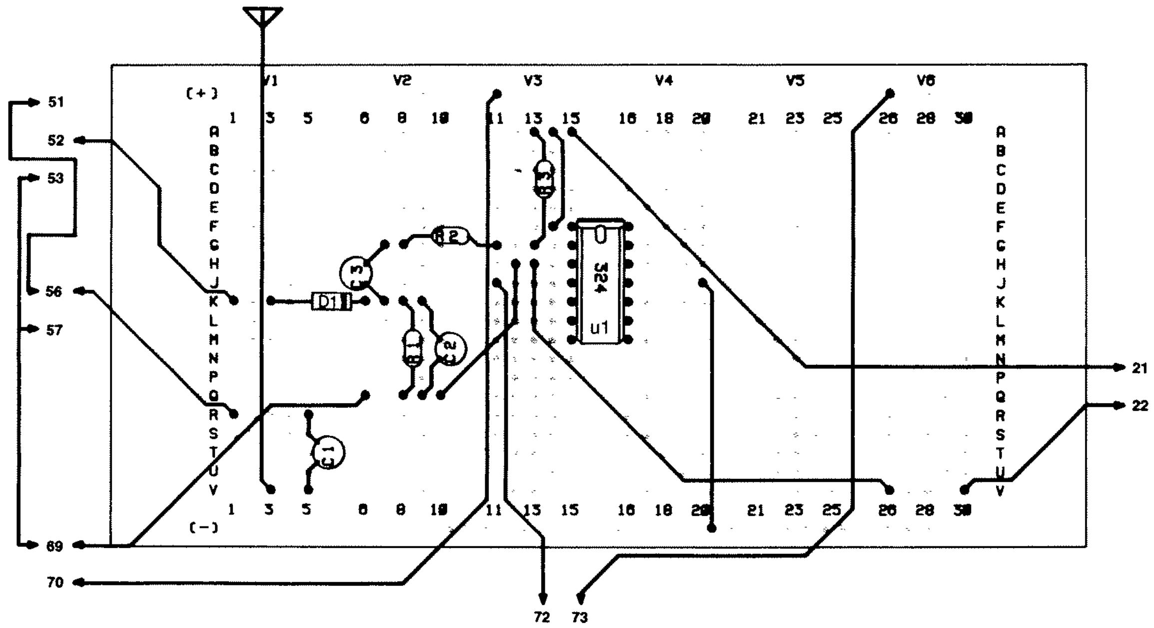
When you finish wiring up the circuit, erect an antenna with a length of about 1 m, set the **tuning control** in the 12 o'clock position. Try attaching the long wire to water and turn the dial of the AM radio. If necessary rotate the **tuning control** of the kit so you can hear the sound clearly.

Of course, you can come up with some interesting applications for this circuit on your own. You might try adding the **CdS cell** between the PNP transistor's emitter and base. Or you could add the 50K **control volume** and key in the emitter base circuit or ... well, you get the idea. Be sure to keep notes of what you manage to come up with.

See--you're starting to design your own circuits!



PROJECT 66. IC RADIO

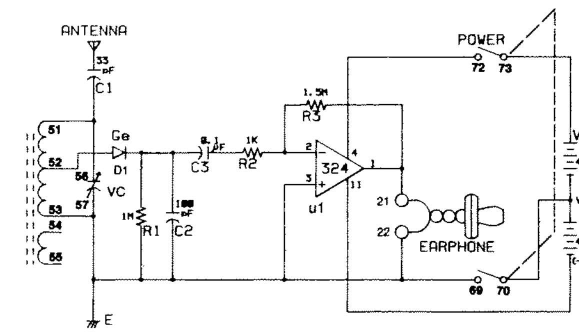


U1	324	R1	1MΩ	C1	33pF	D1	Ge
		R2	1KΩ	C2	100pF		
		R3	1.5MΩ	C3	0.1μF		

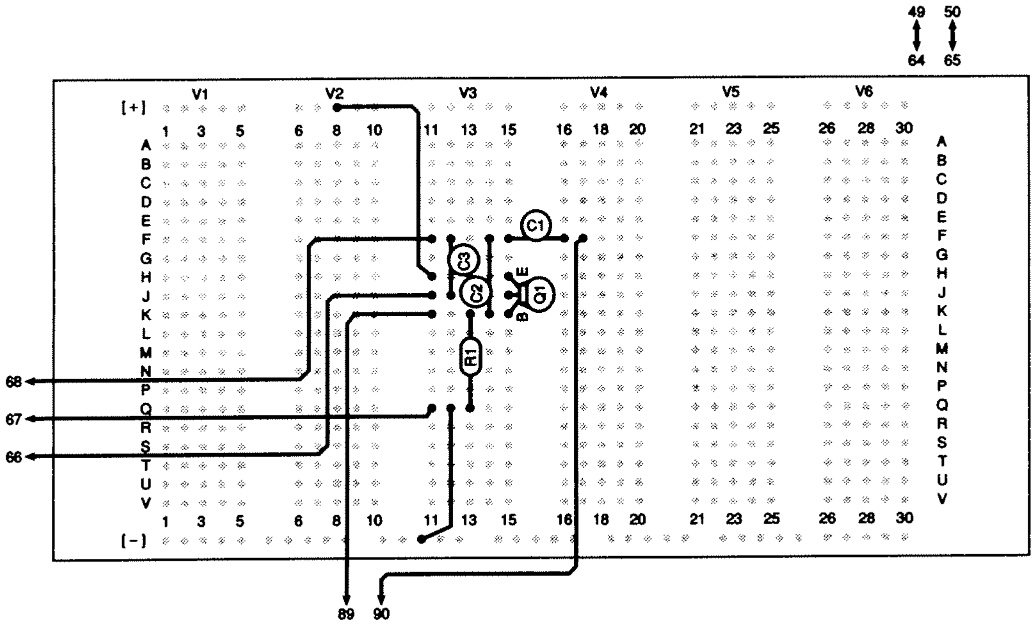
This is an IC radio: it uses the operational amplifier to amplify the output of a germanium radio and produce a loud sound from the earphone. You can receive an AM broadcasting in range of 530 - 1600 kHz.

The coil and the **tuning control** form a tuning circuit. When radio waves strike the antenna and a low current is applied to the antenna coil, this tuning circuit selects one radio frequency. The selected radio wave is detected by the germanium diode (Ge diode) and converted to an audio signal. This audio signal is amplified by the Operational Amplifier so you can hear it through the earphone.

When you finish the wiring ground the antenna, turn power ON, and turn the **tuning control** until you receive the station you want to hear. Did you receive it all right? Now, change the antenna and ground connection arrangement so you can tune in other stations on the AM band.



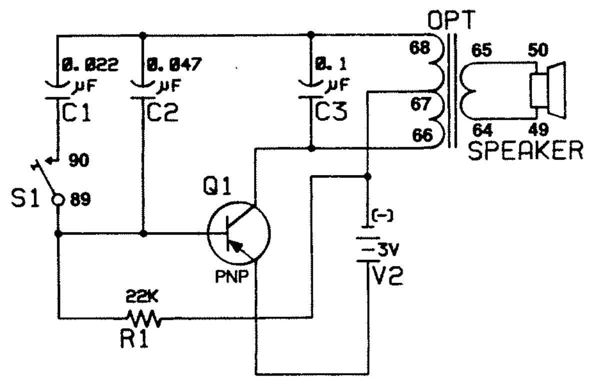
PROJECT 67. TWO-TONE PATROL CAR SIREN



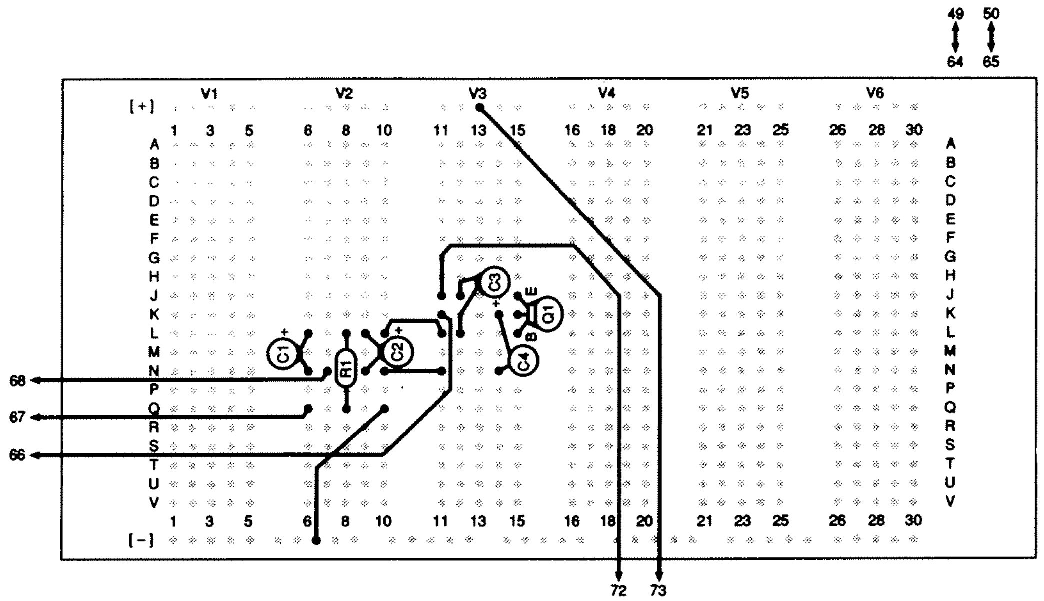
Q1	PNP	C1	0.022µF
R1	22KΩ	C2	0.047µF
		C3	0.1µF

Here is a loud siren which is so much like the real sirens on some police cars or ambulances. You have to be careful you don't confuse people! The initial tone is at a high pitch, but when the key is closed the pitch decreases. You can control the cycling of the pitch the same as the police and ambulance drivers do.

As you can see from the schematic, this is the same type of oscillator circuit we've used in many other projects. Current to the transistor's base is supplied through the 22K resistor. Feedback to keep the oscillator going comes from the 0.047µF capacitor as they charge and discharge. When you press the key, another 0.022µF capacitor is inserted into the circuit to slow down the switching action as its charge hold the transistor on and off.



PROJECT 68. PLANT GROWTH STIMULATER

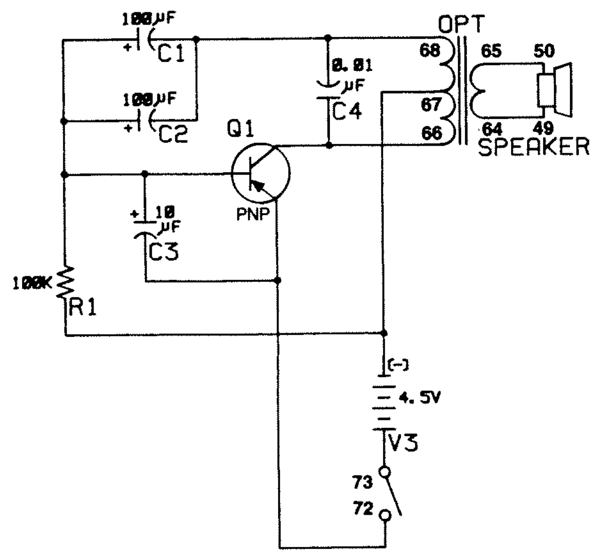


Q1	PNP	C1	100µF	C3	10µF
R1	100KΩ	C2	100µF	C4	0.01µF

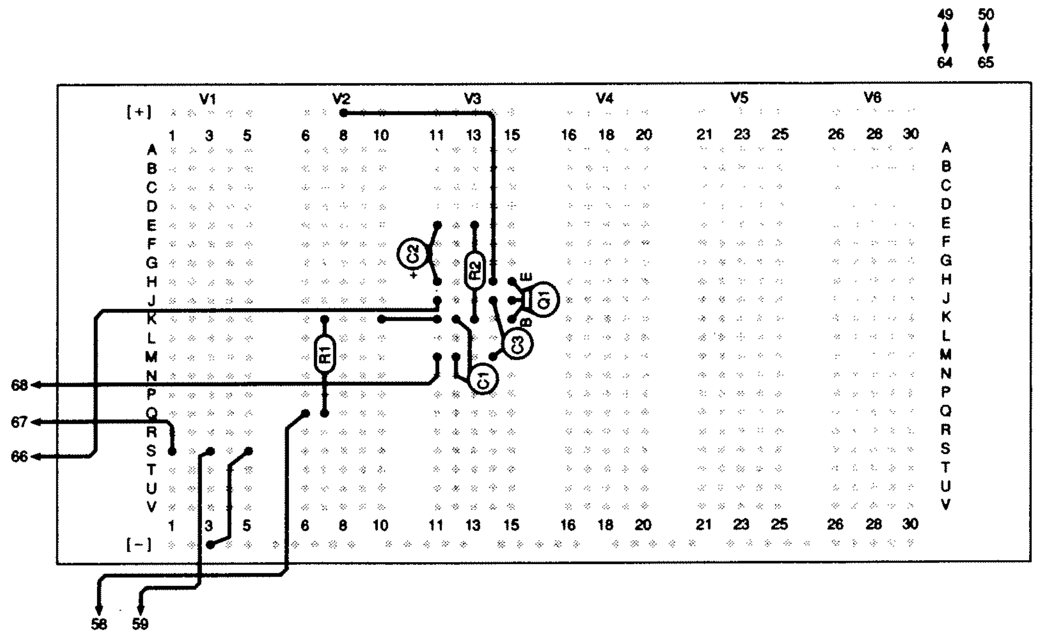
Some persons claim that sound can affect how fast plants grow! Here's a project that lets you test this idea for yourself!

The circuit must now be familiar to you... yes, it is the oscillator.

To see if this project can actually speed up the growth of plants, you need to set up an experiment just like a scientist does. Take a dozen budding plants (such as beans) and divide them into two groups. Make sure each group gets the same amount of sunlight, water, fertilizer and keep both groups at the same temperature. Let one group "hear" sound from the project for a few minutes each day. At the end of two weeks compare the two groups. Has one grown faster than the other? Is the difference large or small? (Small difference in the growth rate might be due to chance.)



PROJECT 69. ELECTRONIC WOODPECKER



Q1	PNP	C1	0.022µF
R1	47KΩ	C2	100µF
R2	1KΩ	C3	0.047µF

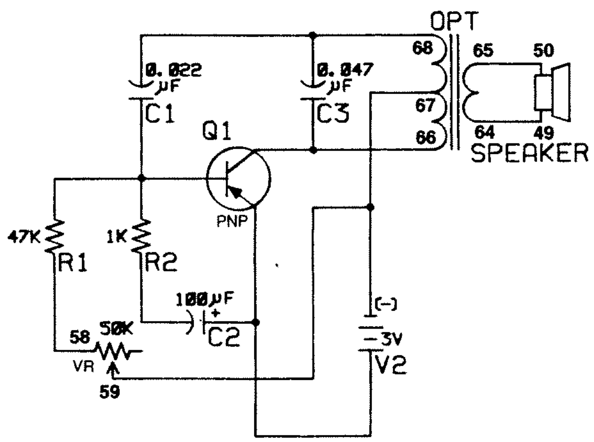
Have you ever heard a red-headed woodpecker chirping? Here is an electronic bird capable of reproducing the sounds of the red-headed woodpecker. If you have them around your house they might fly near by to try to see this electronic relative!

The basic circuit shown does not have a switch or key but you can wire one in yourself. Simply replace one of the wires connected to the battery with leads to the key or switch. The key provides more convenient control when carrying the kit around outside as you try to attract birds with your bird calls.

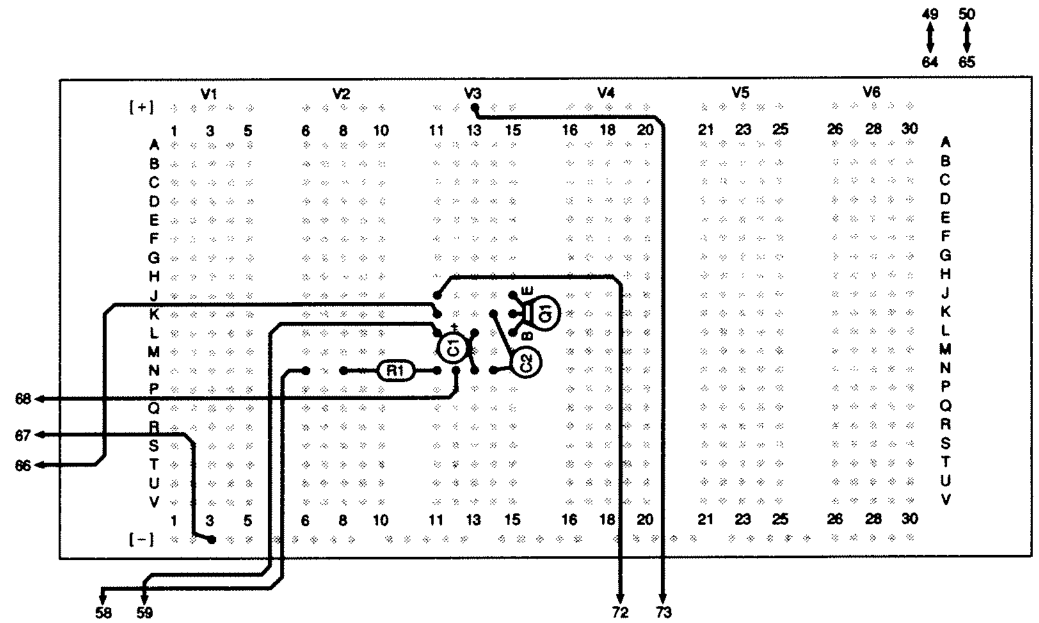
You can also try the 9V supply. The output is louder and resembles even more the scolding chirps of the red-headed woodpecker. The chirps with the 3V supply circuit resemble more the English sparrow.

When experimenting with this circuit you can change almost anything without causing damage. However, do not decrease the 47K resistor to below about 10K or the transistor can be damaged.

Some combination of resistance and capacitance in place of the 1K and 100µF results in some interesting sounds from crickets to bears! Don't forget to record your results for later use, like a good scientist does.



PROJECT 70. FISH CALLER



Q1	PNP	C1	3.3µF
R1	4.7KΩ	C2	0.1µF

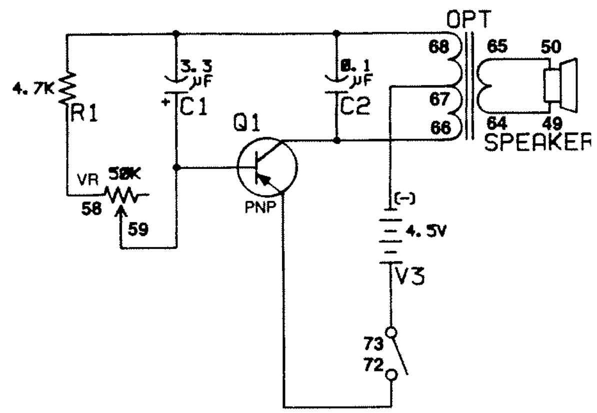
Plants are not the only things in nature which respond to sound. Did you know that some marine animals communicate with each other by sound? You've probably heard that whales and porpoises communicate with sound, but they're not the only ones. Research indicates that some fish are attracted by certain sounds. This circuit lets you see for yourself.

First try building this project on dry hand. When you slide **select switch** to turn power ON, you hear sound pulses from the **speaker**. You can control the rate of the pulses with the **control volume**. Look at the schematic you find this to be a variation of the audio oscillator circuit we've used in other project in this Manual.

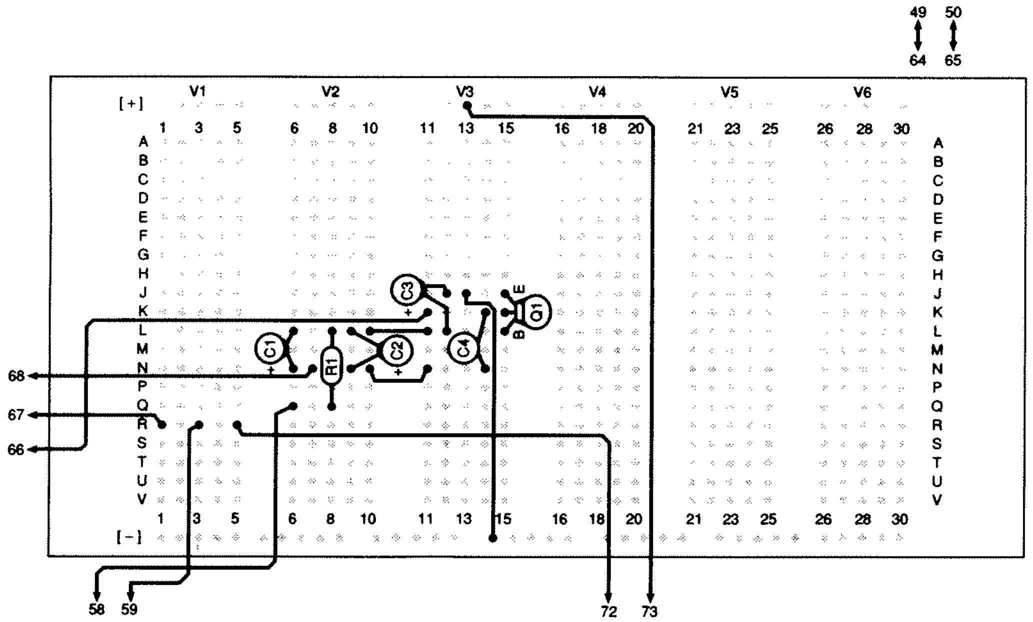
How well does this work in attracting fish? If you have an aquarium at home or at your school, you can place your kit near the aquarium glass and watch to see if fish are attracted to the sound. Or you can actually try it out while fishing. Get another speaker and attach it to terminals **65** and **64** using long lengths of insulated wire.

Carefully wrap the **speaker** in a waterproof plastic bag or seal it inside of a jar. Make sure no water can reach the **speaker**. Now lower it into the water. Then cast a line in the water and wait for the results.

If you don't have much luck with this project, try altering a few parts values for a different pulse sound. Be sure to keep notes of your results - and good fishing.



PROJECT 71. ELECTRONIC RAINDROPS

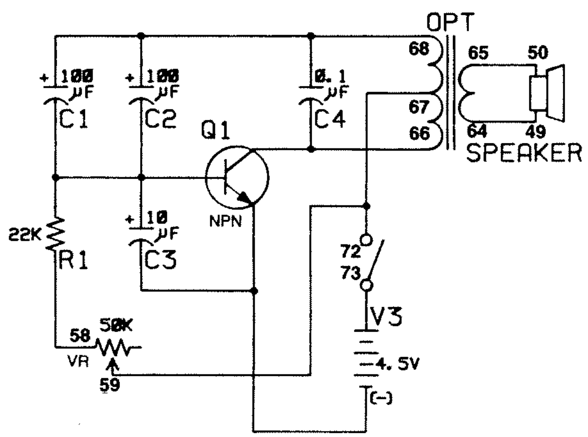


Q1	NPN	C1	100μF	C3	10μF
R1	22KΩ	C2	100μF	C4	0.1μF

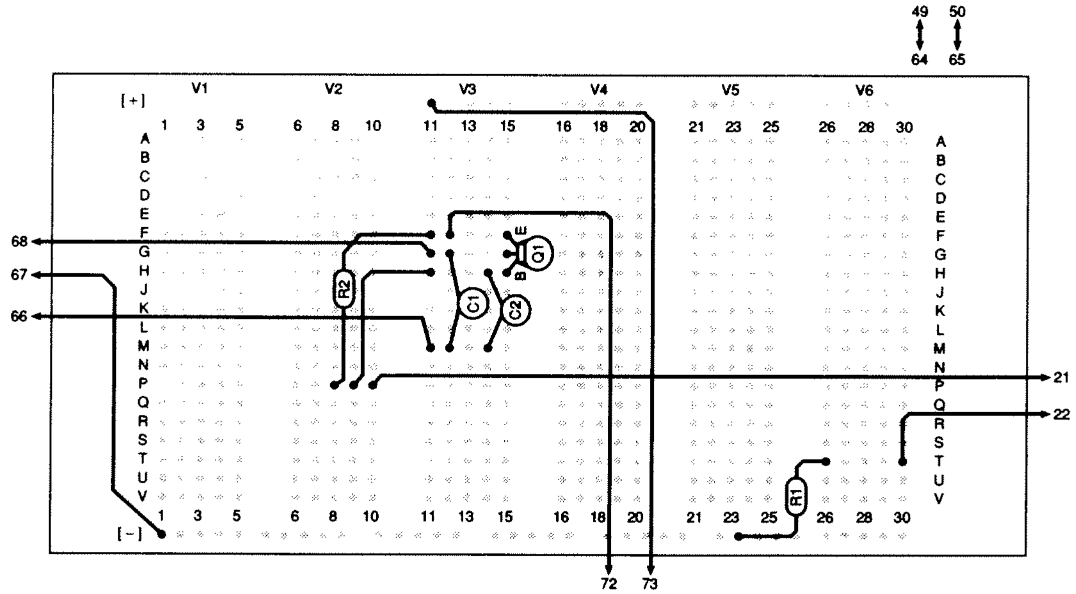
This circuit produces a sound like raindrops falling. By adjusting the **control volume** you can change it from a drizzle to a downpour.

As you can see on the schematic, this project uses two 100μF capacitors in parallel. Their charging and discharging rates are controlled by the 22K resistor and 50K **control volume**. When you turn power ON, you hear the "falling raindrops" sound from the **speaker**. Moving the **control volume** causes the sound to "speed up" or "slow down."

Try substituting different values of capacitors and resistors in this project, just like you've done in previous circuits. Be sure to keep notes about what you discover.



PROJECT 72. PENCIL LEAD ORGAN



Q1	PNP	R1	4.7KΩ	C1	0.047μF
		R2	22KΩ	C2	0.1μF

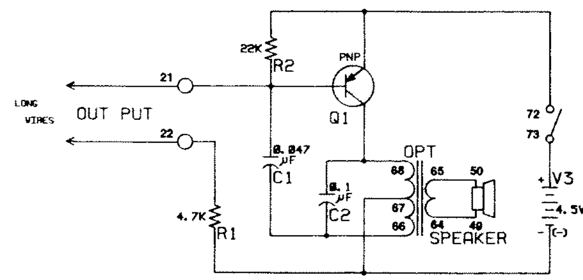
You know about electronic organs that have keyboard... but one with a pencil lead? It's possible with this project!

To use this project, you need to draw a box four inches long and a quarter-inch wide on a sheet of paper. Fill in the box using a soft pencil. Make sure that a heavy coating of pencil lead is left in the box.

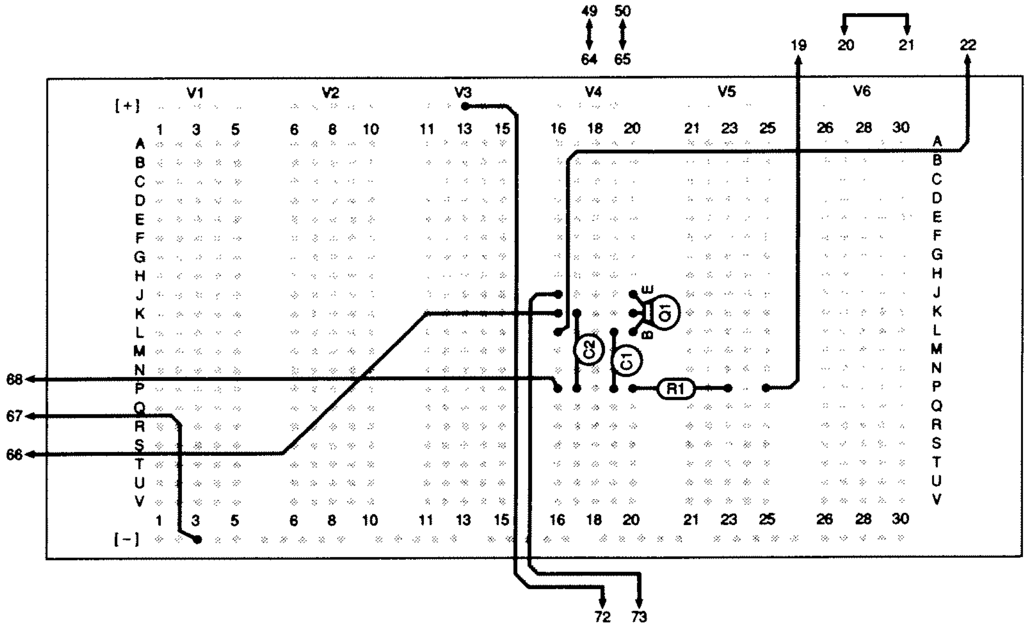
We know what you're thinking... "why on earth did I do that??" Believe it or not, that penciled-in box is the keyboard you're going to use to play the organ!

Turn power ON and place the exposed end of one of the long wires at one end of the penciled-in box. Take the exposed end of the other long wire and place it at the opposite end of the box. Carefully move the two long wires toward each other. You'll discover that you hear sound from the **speaker** when the two long wires are close enough. As you get more skilled in moving the wires, you'll be able to play simple tunes on your Pencil Lead Organ.

Don't be surprised if you have tried several different pencils before you find one that gives good results. A very soft lead (such as the kind in a #1 pencil) will probably work best.



PROJECT 73. ELECTRONIC MOTORCYCLE



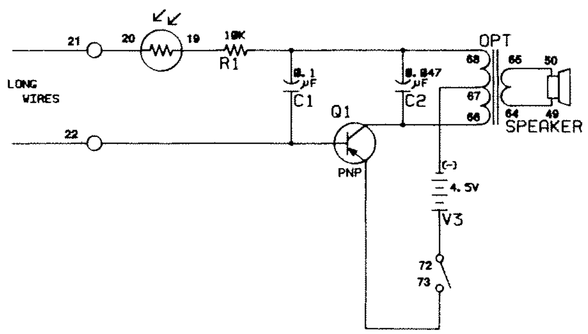
Q1	PNP	C1	0.1μF
R1	10KΩ	C2	0.047μF

Ever try steering a motorcycle with just four fingers? That's dangerous on a real motorcycle, but it's a lot of fun on this electronic version.

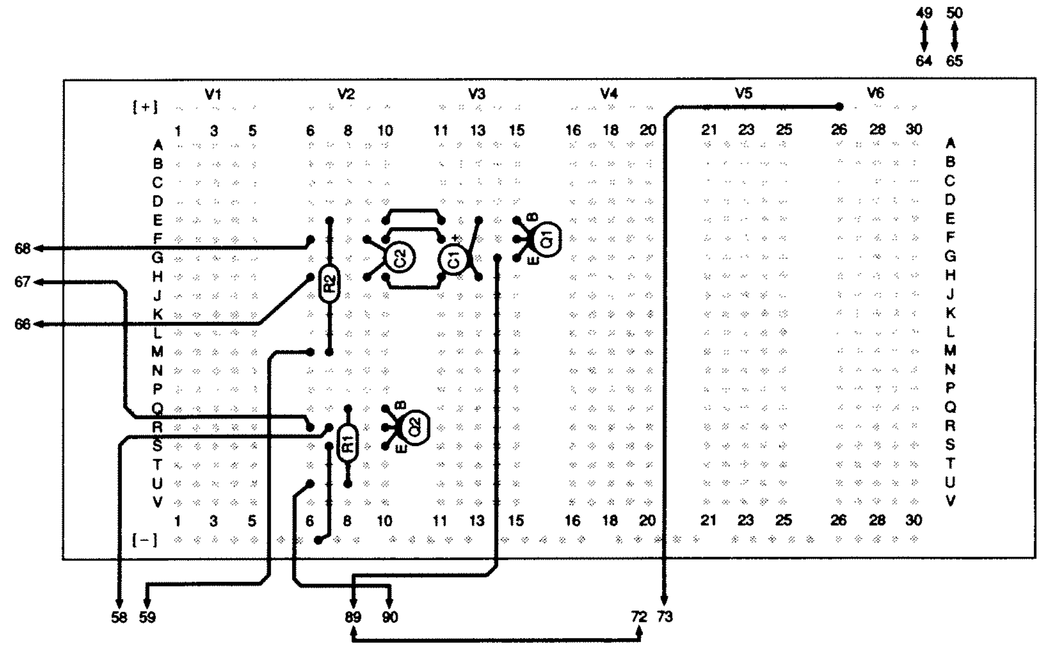
This circuit is similar to the one we played with in project 34. You notice that we've added a **CdS cell** to this circuit to give you even more control over its operation.

To use this project, turn power ON. Grasp the exposed metal ends of each long wire with the thumb and index finger of your hands. Now vary the grip as you listen to the sound from the **speaker**. With a little practice you can make the sound of a speeding motorcycle. You can also get different sounds by controlling the amount of light that falls on the **CdS cell**.

Try using this circuit without the **CdS cell** and with a different capacitor in place of the 0.1μF one (remember to carefully observe the polarity of the electrolytic capacitors - see the description of different parts at the front of this manual if your memory needs refreshing).



PROJECT 74. MACHINE GUN PULSE DETECTOR

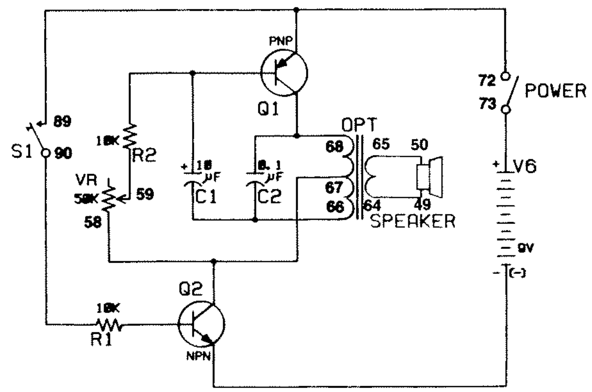


Q1	PNP	R1	10KΩ	C1	10μF
Q2	NPN	R2	10KΩ	C2	0.1μF

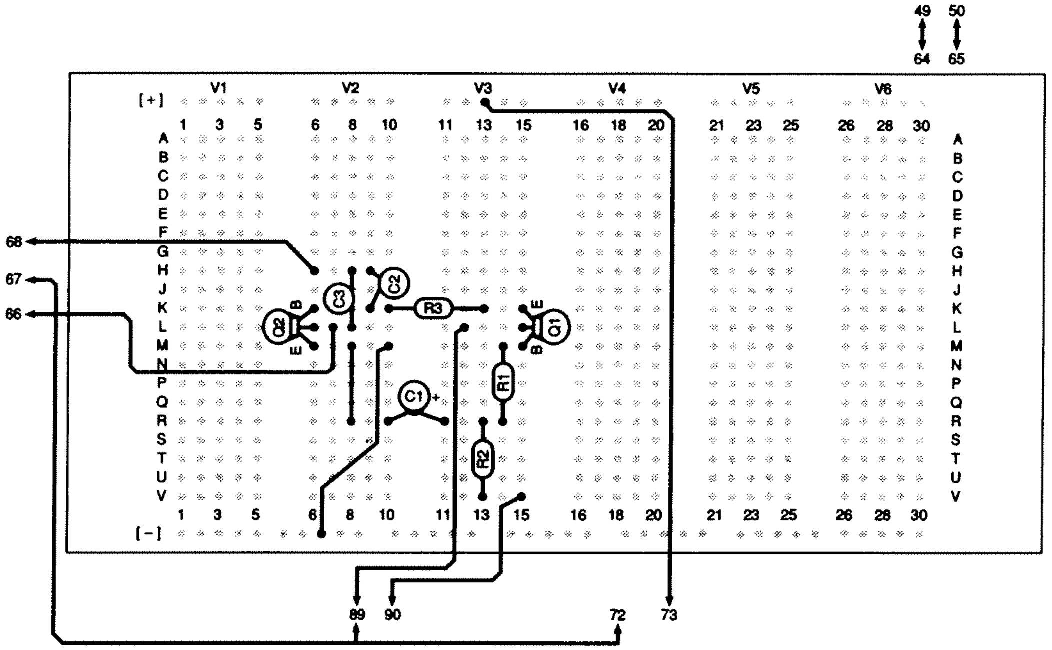
This project is a pulse oscillator that has a sound like a machine gun or a one cylinder motorcycle engine. Adjust the **control volume** to change the sound from the **speaker** from a few pulses per second to a dozen or so per second.

After wiring this project, turn power ON and press **S1**.

You can experiment and change the frequency of this oscillator by trying other capacitors in place of the 10μF capacitor. Be sure to observe the + and – connection (polarity) on the capacitors marked with a + sign. Don't forget you can change the frequency by adjusting the **control volume**, too!



PROJECT 75. ELECTRONIC SIREN

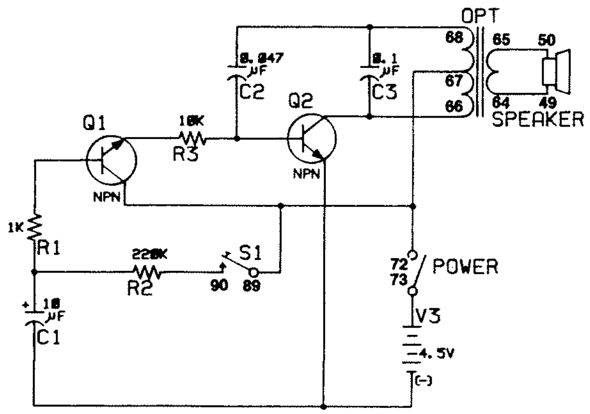


Q1	NPN	R1	1KΩ	C1	10µF
Q2	NPN	R2	220KΩ	C2	0.047µF
		R3	10KΩ	C3	0.1µF

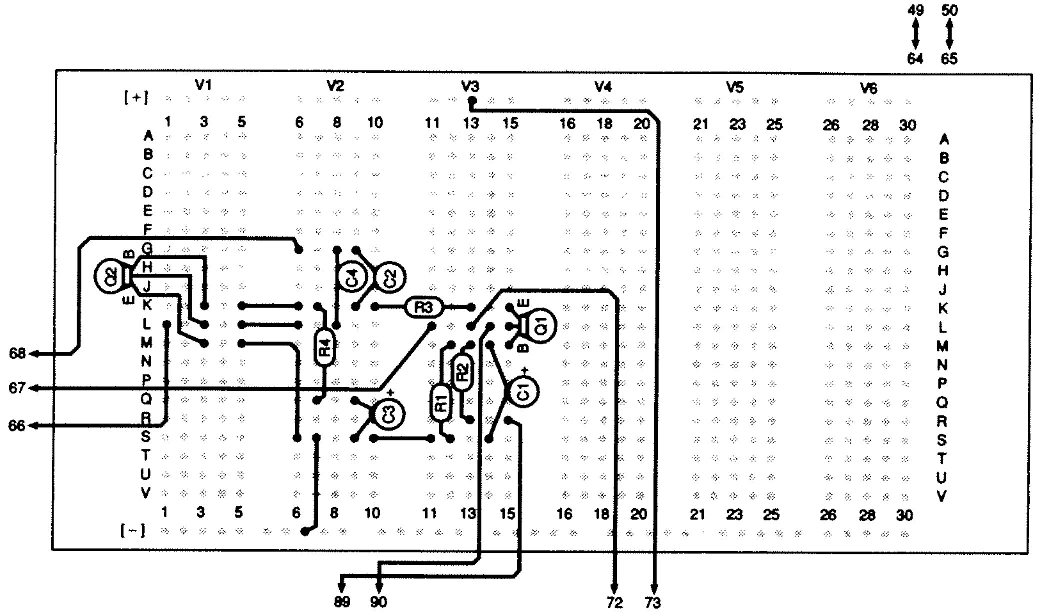
Turn power ON, and press **S1**. You'll hear the sound from the **speaker** slowly build step-by-step until it reaches a mindboggling peak. Release **S1**, and the volume from the **speaker** gradually diminishes and eventually stops.

Carefully compare the schematic of this project with that of our next project. What's the biggest difference you can spot? Try some other ways of changing the current to the base of the right transistor and see what other sounds you can come up with.

You can adapt this circuit for other applications... wouldn't this make a terrific door or other alarm?



PROJECT 76. CHIRPING BIRD



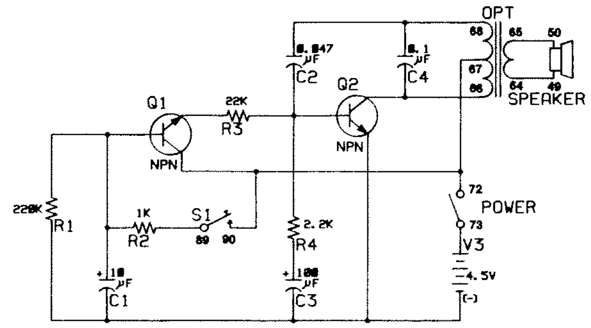
Q1	NPN	R2	1KΩ	C1	10µF	C4	0.1µF
Q2	NPN	R3	22KΩ	C2	0.047µF		
R1	220KΩ	R4	2.2KΩ	C3	100µF		

Here's a project that imitates our feathered friends - you could say it mocks the mockingbird!

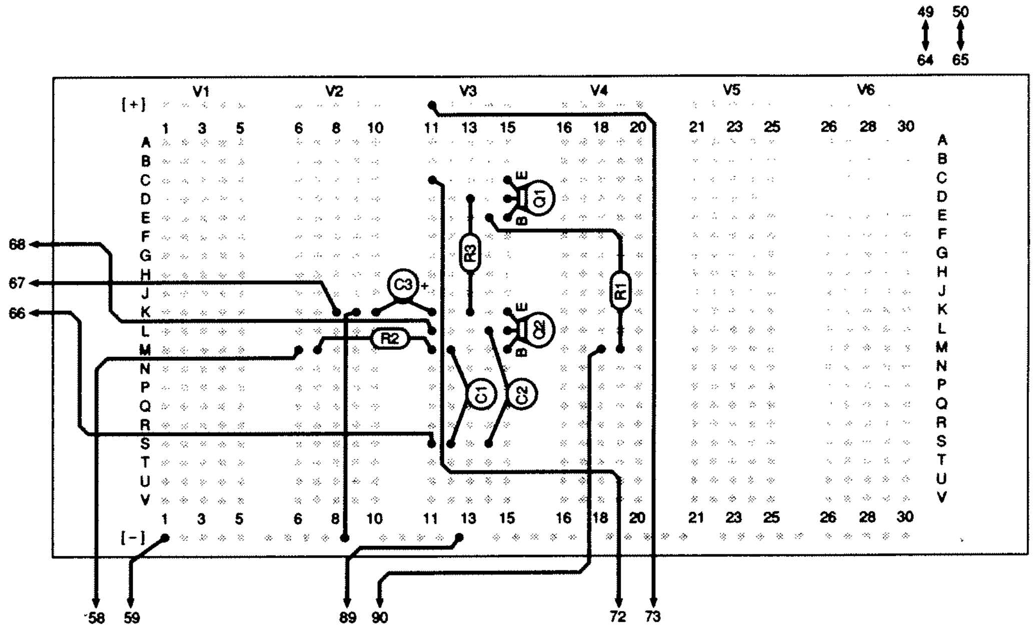
When you turn power ON, you won't hear any sound from the **speaker**. Now press **S1**. You'll hear a chirping sound from the **speaker**. Release the key, and you still hear the chirping sound for a few moments until it stops. Take a look at the schematic and see if you can tell why this happens.

You see that when you press **S1** a current is supplied to the base of the left transistor. This allows the transistor to operate and supplies current for the right transistor. The 10µF capacitor charges when **S1** is pressed and starts to discharge when **S1** is released. When it is completely discharged, the circuit no longer works.

Try a different value of capacitor in place of the 10µF and 100µF ones and see what happens.



PROJECT 77. ELECTRONIC CAT



Q1	PNP	R1	10KΩ	C1	0.047µF
Q2	PNP	R2	22KΩ	C2	0.1µF
		R3	100Ω	C3	100µF

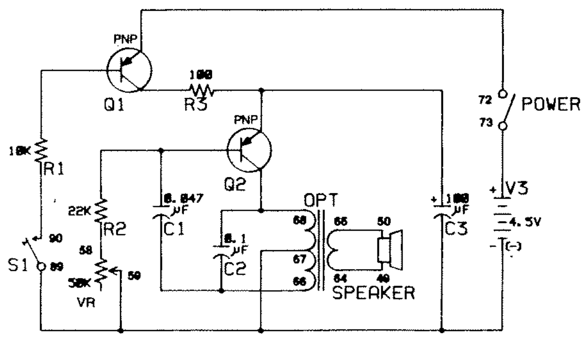
Bothered by mice? And you don't have a mousetrap? Try this project instead - see if the sound of the electronic feline can keep them away. (And you don't have to bait it with any cheese!)

This project is another variation of the audio oscillator circuit we've used in other projects. Look at the schematic for this project and notice the location of 100µF capacitor. This is why this circuit operates even if you only briefly press the key and then release it.

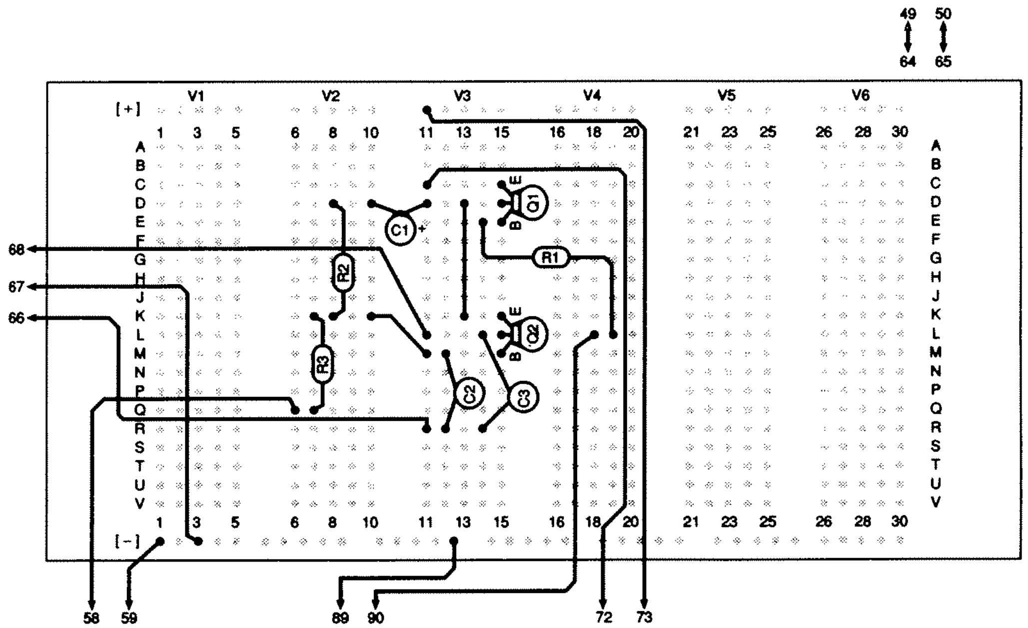
When you press **S1** after turning power ON, you hear the "cat's meow" from the **speaker**. Adjust the **control volume** - what effect does it have on the circuit's operation?

You can experiment with this circuit and produce a variety of other sounds. Just don't change the value of 0.047µF capacitor to more than 10µF or reduce the value of the 100 ohm resistor - otherwise the transistor might be damaged.

Have fun with this project - and we hope you're not bothered by mice anymore!



PROJECT 78. ELECTRONIC BIRD

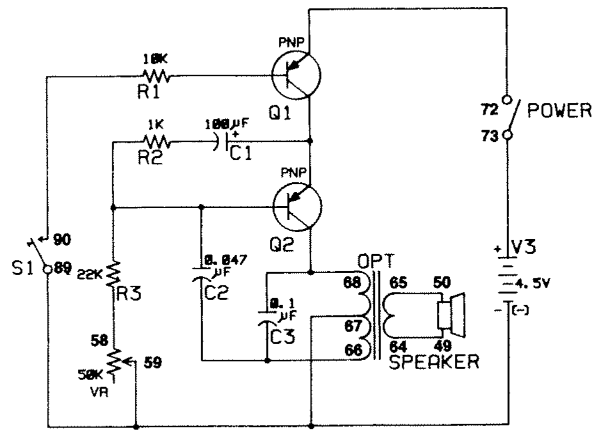


Q1	PNP	R1	10KΩ	C1	100µF
Q2	PNP	R2	1KΩ	C2	0.047µF
		R3	22KΩ	C3	0.1µF

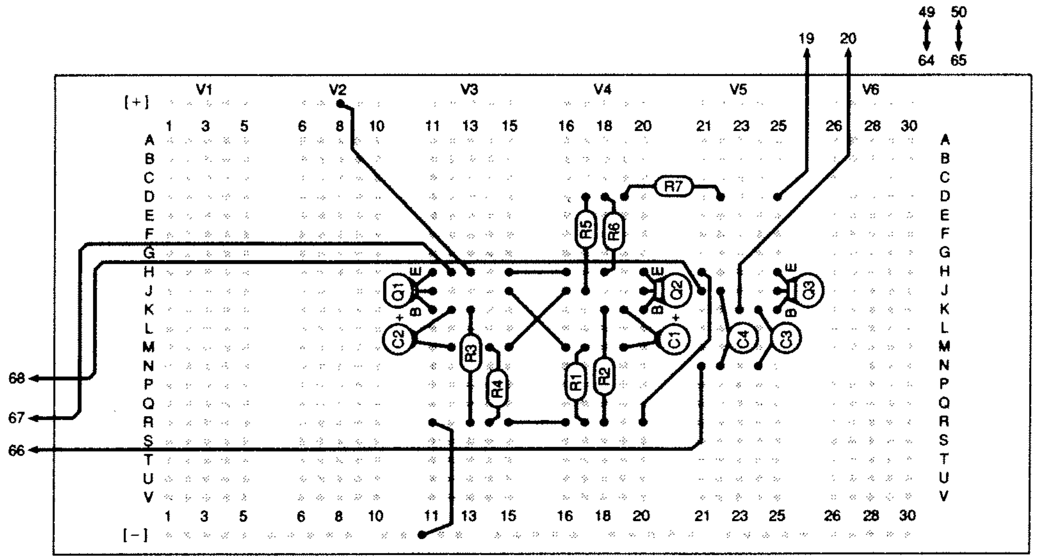
Have you heard the old saying about the "cat that ate the canary?" We use that expression when we mean someone looks guilty of having done something they shouldn't have. (So what's that got to do with electronics? Simple - we've decided to make an electronic bird for our electronic cat!)

This circuit is another variation of an audio oscillator circuit. When you press **S1** after turning power ON, you hear a "chirping" sound from the **speaker**. You can make the "bird" chirp faster or slower by adjusting the **control volume**.

If you like to make changes to these projects (of course you do!) you'll love this circuit! You can change just about any part and get a different result. Just don't decrease the value of the 22K resistor below 10K (this helps prevent damage to the transistor).



PROJECT 79. "HORROR MOVIE" SOUND EFFECT



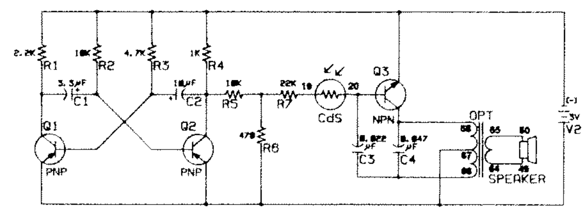
Q1	PNP	R1	2.2KΩ	R5	10KΩ	C1	3.3µF
Q2	PNP	R2	10KΩ	R6	470Ω	C2	10µF
Q3	NPN	R3	4.7KΩ	R7	22KΩ	C3	0.022µF
		R4	1KΩ			C4	0.047µF

The sound that it produces might remind you of the "scary" music you hear in horror movies.

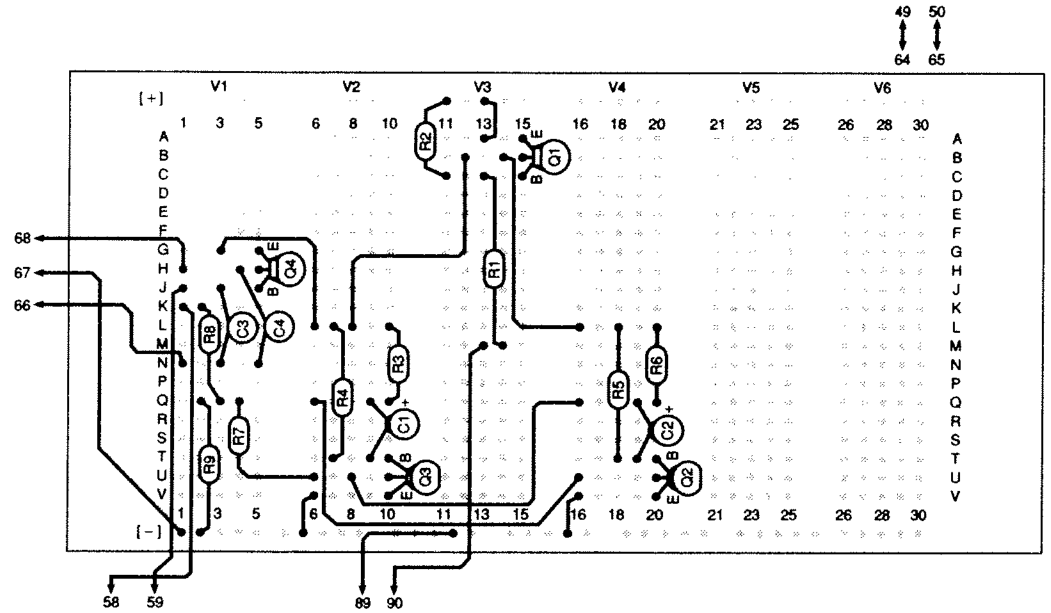
After wiring the project, you'll hear "scary" sound. Why don't you try to change the light that falls on the **CdS cell** with your hand?

The pulse type oscillator on the right side of the schematic produces the basic sound. The base bias of the NPN transistor is supplied through two sources: the multivibrator circuit on the left side of the schematic and the **CdS cell**. The multivibrator provides a pulsating current flow, which causes the tremolo (wavering tone) effect, and the resistance of the **CdS cell** controls how much of that current reaches the transistor base. This determines the charge/discharge rate of the capacitor and therefore the frequency of the pulse oscillator.

When the frequency of an oscillator is controlled by another circuit, it is called FM or frequency modulation. An FM radio signal is something like this, except at much higher frequencies.



PROJECT 80. ELECTRONIC ORGAN



Q1	PNP	R1	10KΩ	R5	4.7KΩ	R9	470Ω	C4	0.047µF
Q2	NPN	R2	22KΩ	R6	1KΩ	C1	10µF		
Q3	NPN	R3	1KΩ	R7	47KΩ	C2	3.3µF		
Q4	PNP	R4	10KΩ	R8	22KΩ	C3	0.1µF		

Electronic organs have revolutionized the music world in recent years. This project shows you the principles of electronic organs and let you play some simple tunes as well.

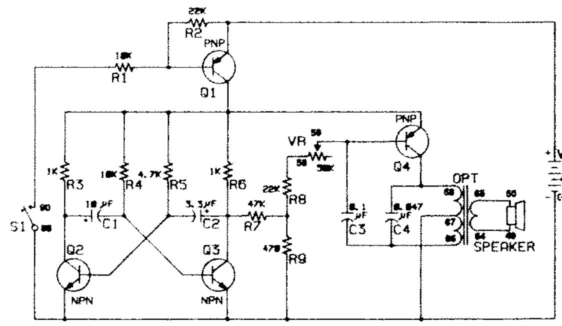
As you can see from the schematic, this project is an oscillator circuit coupled to an amplifier circuit. Remember how we told you complex electronic devices are actually composed of simple circuits?

After you finish wiring connections, press the key down. You'll hear a tone from the **speaker**. You can vary this tone from the **speaker**. You can vary this tone with the 50K **control volume**. Try pressing the key and rotating the **control volume** so that you can produce simple tunes.

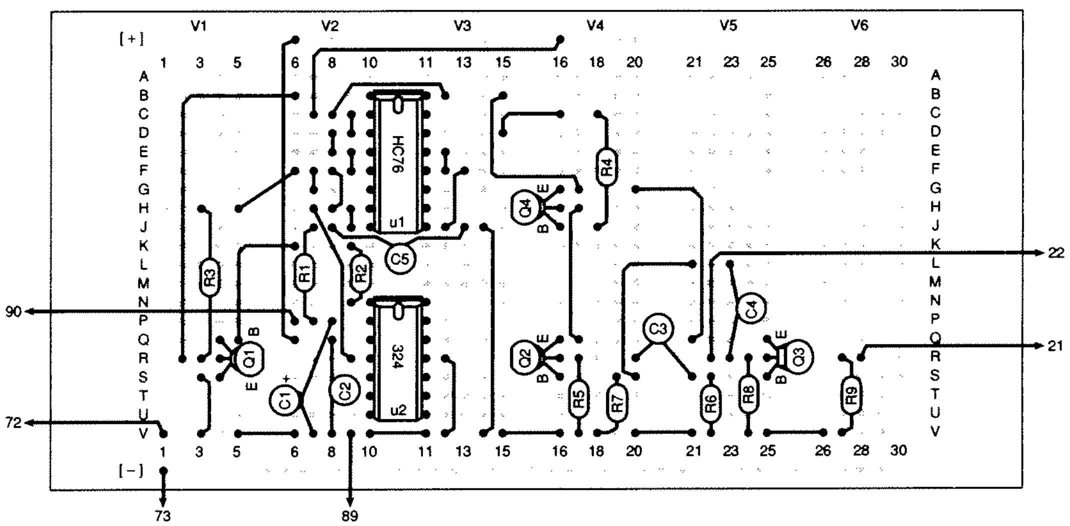
This circuit is a "natural" for experimentation! You can alter the tonal range by changing the 10µF and 3.3µF capacitor to different values. Try adding the **select switch** in place of the key - and see if you can use the key to add another component to the circuit (like an extra capacitor in parallel to the 10µF or 3.3µF ones). Be sure to keep notes of what you do.

Since this is our last audio oscillator project for a while, it might be good to review your notes. What have you discovered about oscillators so far? Were you able to come up with some interesting circuits on your own? Try making up your own "Electronic Zoo and Sound Factory" like we've done here.

Now here's a challenge for you - think you can make up your own audio oscillator circuit using just your notes and what you've got stored in your head? Try it, and don't peek at this Manual. We think you can do it. (And if you do, you're designing electronic circuits just like we promised you would do!)

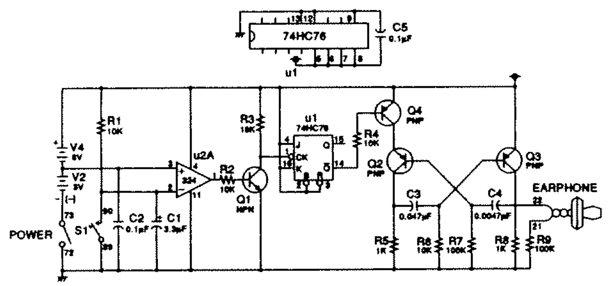


PROJECT 81. SOUND MACHINE I



U1	74HC76	Q1	PNP	R1	10KΩ	R6	10KΩ	C1	3.3μF
U2	324	Q2	PNP	R2	10KΩ	R7	100KΩ	C2	0.1μF
		Q3	PNP	R3	10KΩ	R8	1KΩ	C3	0.047μF
		Q4	PNP	R4	10KΩ	R9	100KΩ	C4	0.0047μF
				R5	1KΩ			C5	0.1μF

Here's a marriage of digital and analog electronics. Take a good look at the schematic - can you guess how this circuit works even before you build it? Make a mental note of your guess.

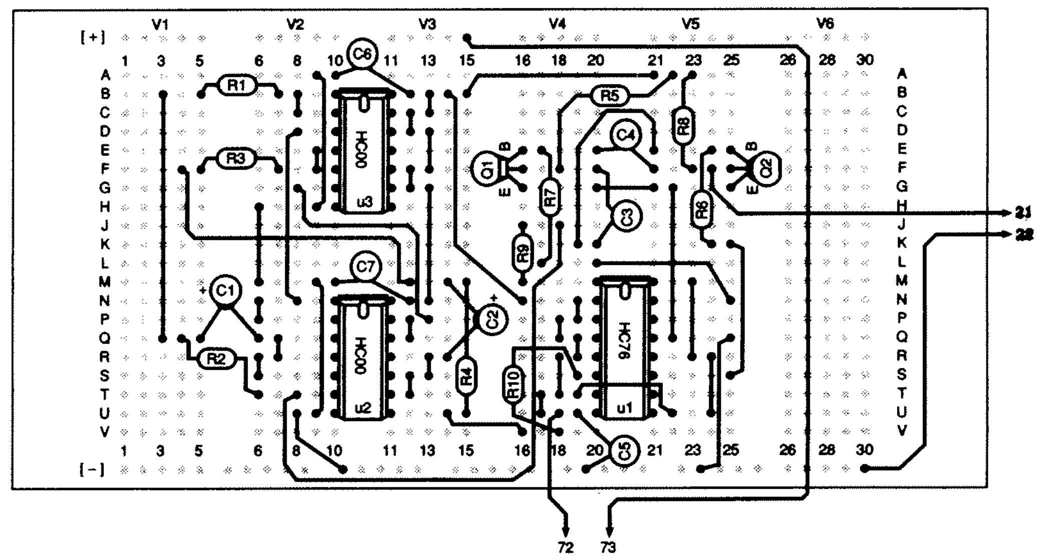


To listen to this project, connect the earphone to Terminals **21** and **22** and turn the **select switch** ON. Can you guess what happens when you press the **S1** key and then release it? What happens when you press it again? If you guessed right about how this circuit works, those two questions should be a cinch.

We don't want to keep you in suspense, so press the key and release it. Listen for a few seconds and then press and release the key again. Now what happens?

By now you probably had no problem recognizing the multivibrator circuit. But, It's the first time for you to use the J-K flip-flop. If has many functions. However, first study one basic function that "the J-K flip-flop changes its state every time a pulse is input to the CK (clock) terminal if both J and K terminals are at 1. "Press the key once, and the J-K flip-flop sets. You heard sounds in the earphone. When you pressed the key a second time, the sounds stopped. This was because the circuit reset.

PROJECT 82. SOUND MACHINE II



U1	74HC76	R1	22KΩ	R6	4.7KΩ	C1	1μF	C6	1000pF
U2	74HC00	R2	47KΩ	R7	10KΩ	C2	1μF	C7	0.1μF
U3	74HC00	R3	22KΩ	R8	1KΩ	C3	0.1μF	C8	0.1μF
Q1	NPN	R4	150KΩ			C4	0.047μF	C9	0.1μF
Q2	NPN	R5	1KΩ			C5	1000pF		

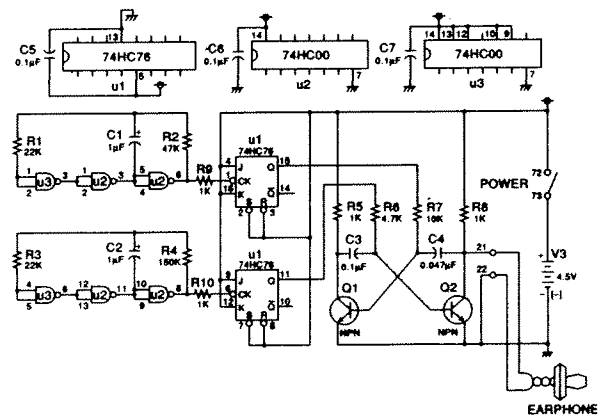
Look at the schematic for this project. You'll see that the NAND gates are connected to the J-K flip-flops at a point known as CK. The output from the NAND gates control the set/reset functions of the J-K flip-flops. Notice how the output of the J-K flip-flops go to the bases of the transistors in the multivibrator circuit. What effect do you suppose this has upon the operation of the multivibrator?

Take care as you make the wiring connections for this project - it's easy to make a mistake with so many connections. Connect the earphone to Terminals **21** and **22** and turn power ON. What kind of sounds do you hear?.

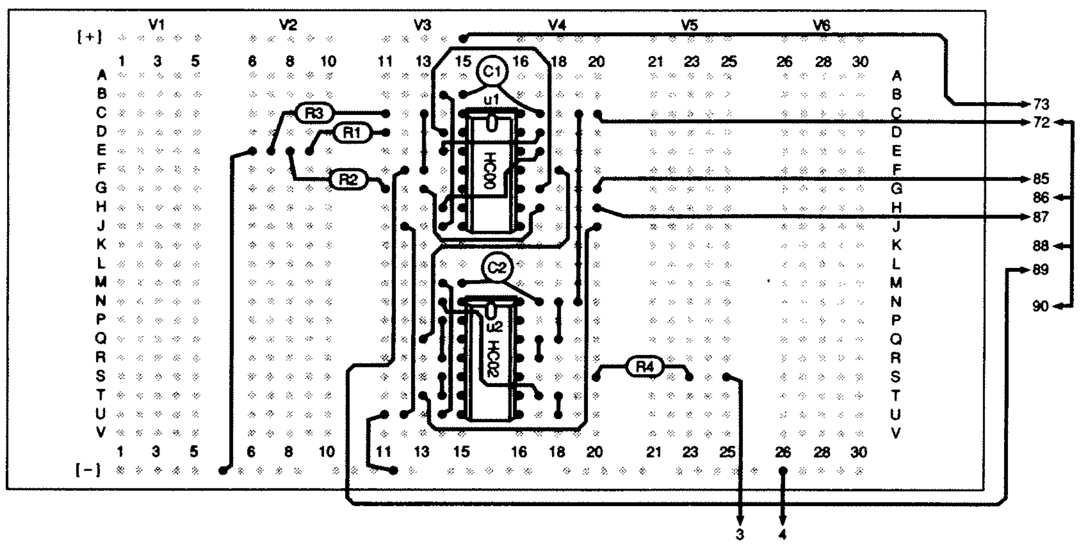
The strange sounds you hear are produced by the digital portion of the circuit controlling the operation of the multivibrator. You can also alter the operation of the NAND gates (and their effect on the rest of the circuit) by substituting different values of capacitors in place of the 1μF one.

As you might have suspected, the 1μF capacitor control the NAND gates by charging and discharging, which turns the NAND gates on and off. The NAND gates set or reset the J-K flip-flops, whose outputs then go to the bases of the transistors, which then .....

See how everything tits together? Like we said earlier - even complex devices are made up of a few basic circuits.



PROJECT 83. MAJORITY LOGIC GATE

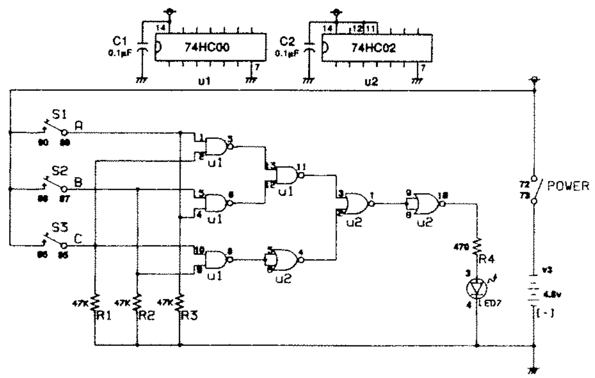


U1	74HC00	R1	47KΩ	R3	47KΩ	C1	0.1μF
U2	74HC02	R2	47KΩ	R4	470Ω	C2	0.1μF

The circuit is called "Majority Logic Gate." It has an odd number of input terminals (three in this project), and judges the input level (1 or 0) of each terminal. Then it judges which of the two kinds of inputs, 1 or 0, are larger in number. Why odd number inputs? Because when you set even number of terminals, there can be a case of a tie.

This circuit is made up of seven NAND gates. See in figure 1: the output is 1 when two or more inputs are at 1, and is 0 when two or more inputs are at 0.

When you finish the wiring for the project, try pressing and releasing for various key combinations while looking at Figure 1, and see how this counter works. A table like this is called "Truth table". It shows the output logic levels against the combinations of the input levels.

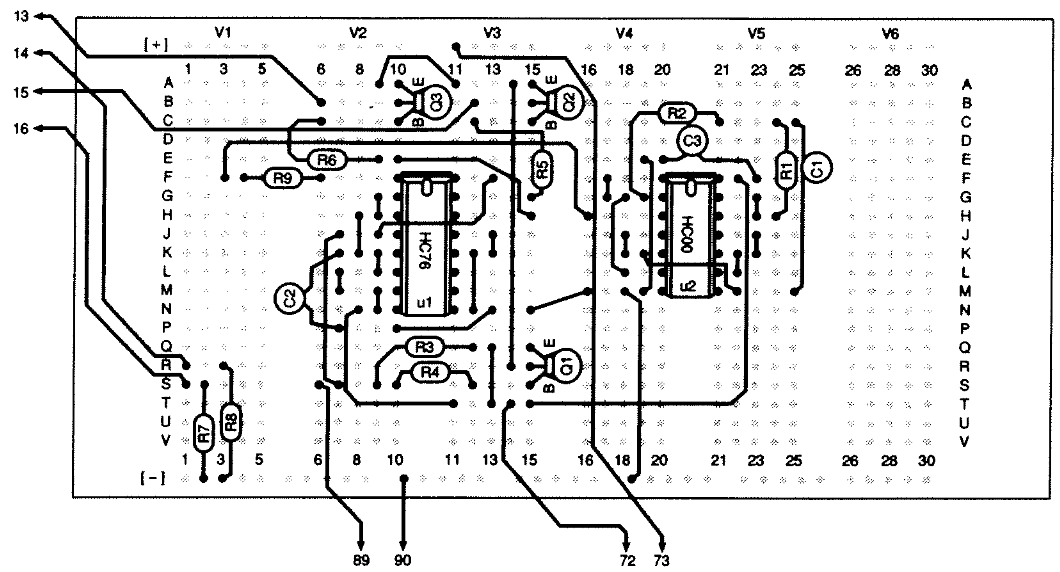


A	B	C	OUTPUT
0	0	0	0
1	0	0	0
0	1	0	0
1	1	0	1
0	0	1	0
1	0	1	1
0	1	1	1
1	1	1	1

Switch ON : 1
Switch OFF: 0
LED ON : 1
LED OFF : 0

Figure 1

PROJECT 84. ELECTRONIC COIN TOSS



U1	74HC76	Q1	PNP	R1	22KΩ	R6	10KΩ	C1	0.01μF		
U2	74HC00	Q2	PNP	R2	22KΩ	R7	470Ω	C2	0.1μF		
		Q3	PNP	R3	10KΩ	R8	470Ω	C3	0.1μF		
				R4	10KΩ	R9	10KΩ				
				R5	10KΩ						

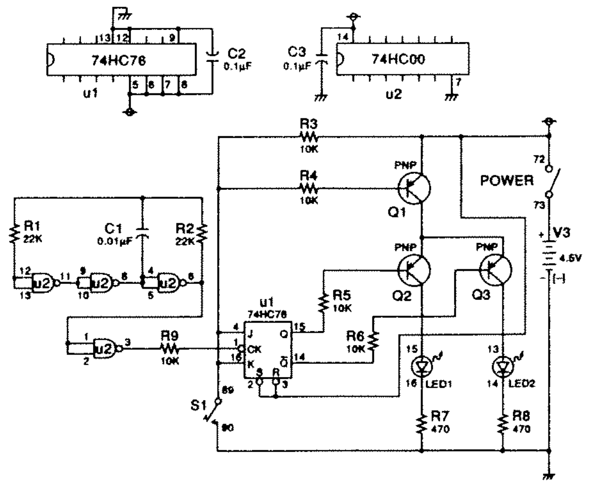
Do you hate to make decisions? If so, you'll love the next several projects! This group of "electronic decision makers" can be used in games in place of dice. And you can make up your own games using these circuits.

Since these projects all "make decisions", you might expect them all to involve digital electronics. And you're right! Take a look at the schematic diagram for this project. You'll see that it involves both NAND gates and the J-K flip-flop. Can you guess how this circuit works before you build it?

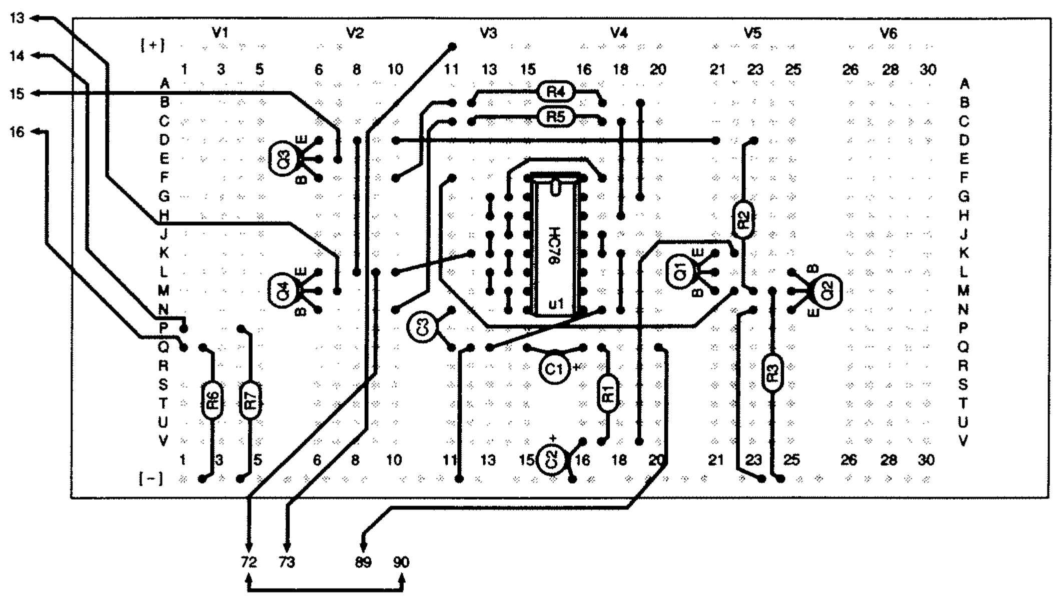
When you finish building this project, turn power ON. Press **S1** key once. Observe what happens to **LEDs 1** and **2**. Press the key several times. What pattern, if any, can you detect in the lighting of the two **LEDs**?

As you probably suspect, while the key is open, the J-K flip-flop is rapidly setting and resetting. This does not affect the **LEDs**, since the transistor Q1 is "off" while the key is open. By pressing the key, you do two things; turn the Q1 on (and thus provide a current path for the **LEDs**), and stop the J-K flip-flop's continuous setting and resetting. One **LED** is set while the other is reset. Does this happen at random or is there a pattern? Try this experiment to find out : press the key fifty times and record which **LED** lights. Does each **LED** light up 25 times? It not, are the results almost equal?

A truly random result would be each **LED** lighting exactly 25 times. It's not unusual for this to vary a bit, such as one **LED** lighting 27 times and the other 23 times. But if there's a big difference - such as one **LED** lighting 40 times and the other only lighting 10 times - that indicates the circuit isn't really random. Would you say this project is or isn't random?



PROJECT 85. ELECTRONIC COIN TOSS II



U1	74HC76	Q4	PNP	R4	10KΩ	C1	100µF
Q1	PNP	R1	33KΩ	R5	10KΩ	C2	1µF
Q2	NPN	R2	470Ω	R6	470Ω	C3	0.1µF
Q3	PNP	R3	1KΩ	R7	470Ω		

Here's a different version of our last project. If you aren't happy with the decisions the last circuit made, then try this one!

You use this project almost like last project. Turn power ON. Now press **S1**. The two **LEDs** take turns lighting and going off. When you release **S1**, only one **LED** remains lit.

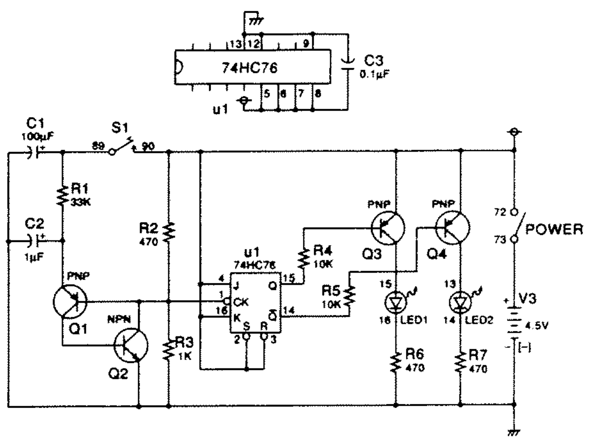
You can see by the schematic that this circuit uses the J-K flip-flop IC. Look at how the **LEDs** are connected - one is connected to a point called Q while the other is connected to a point called $\overline{Q}$. Electronics engineers call Q the set output while the $\overline{Q}$ output is called the reset output.

Remember what we mean when we say a digital device is 0 or 1 ? (If you don't, look back at project 24.) With a J-K flip-flop, we say the flip-flop is set if Q is 1 and $\overline{Q}$ is 0. But if Q is 0 and $\overline{Q}$ is 1, then the flip-flop is reset.

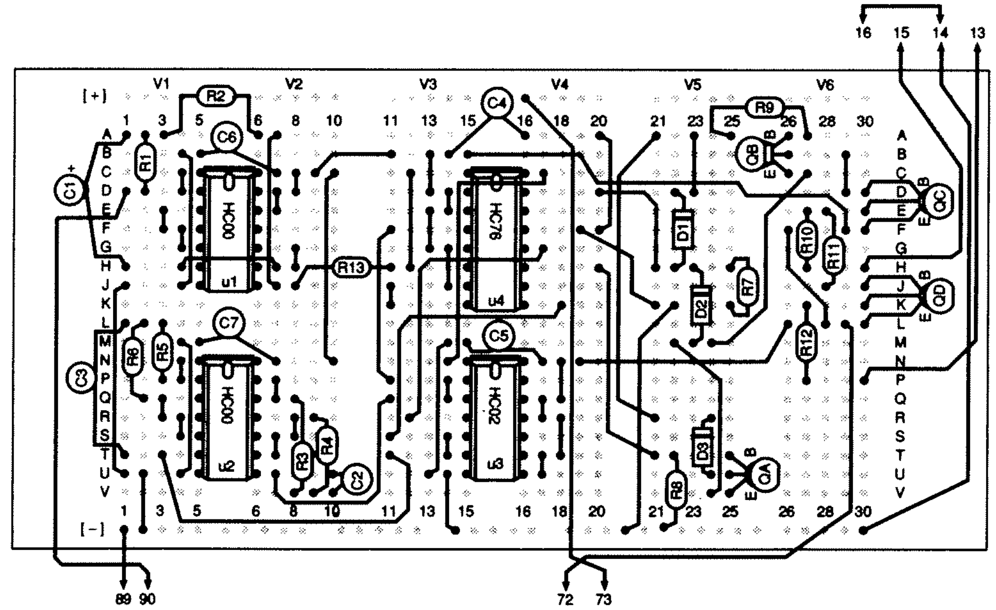
Armed with that bit of knowledge, try your hand at this : which **LED** is lit if the flip-flop is set? Which one is lit if it's reset?

Can you also figure out how or why the J-K flip-flop sets and resets? Do you suppose it has anything to do with the connection point marked CK? (Look back at project 81.)

Try the same experiment you did with project 84 to see if this is a random circuit. Is it more or less random than the previous project? Or are both the same? (If you can't decide, try letting one of the two projects decide for you!)



PROJECT 86. ELECTRONIC COIN TOSS III



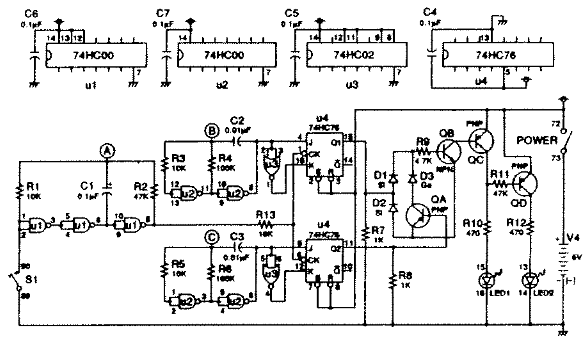
U1	74HC00	QD	PNP	R4	100KΩ	R11	47KΩ	C6	0.1µF
U2	74HC00	D1	Si	R5	10KΩ	R12	680Ω	C7	0.1µF
U3	74HC02	D2	Si	R6	100KΩ	C1	0.1µF	C8	0.1µF
U4	74HC76	D3	Ge	R7	1KΩ	C2	0.01µF		
QA	PNP	R1	10KΩ	R8	1KΩ	C3	0.01µF		
QB	NPN	R2	47KΩ	R9	4.7KΩ	C4	0.001µF		
QC	PNP	R3	10KΩ	R10	680Ω	C5	0.1µF		

Still you can't make decision? OK, then let's try this third version of the coin toss.

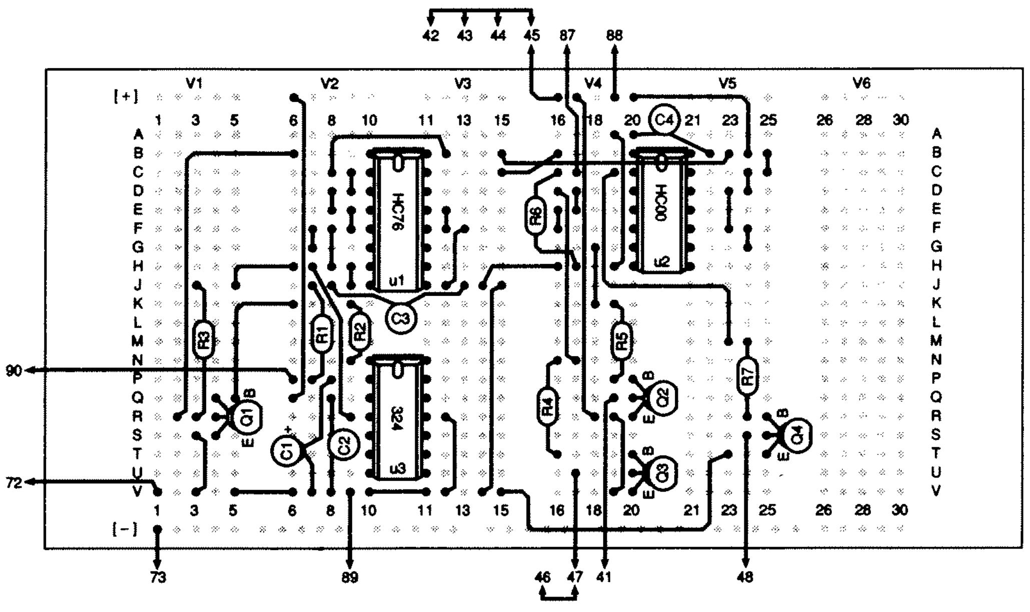
As you see in the schematic, this circuit uses three astable multivibrators, A, B, and C. A generates a clock pulse, and B and C generates pulses that are sent to two flip-flops. Outputs from the flip-flops (Q1, Q2) are sent to an XOR circuit to light up **LEDs 1** and **2**.

When you press **S1**, the clock pulse generation stops, and a random output is produced in Q1 and Q2, lighting up either **LED**.

When you've wired the project, switch power ON and see what happens to the **LEDs**. They take turns lighting, don't they? Now press **S1**, and you notice that just one of the **LEDs** stays ON. Try the same experiment to check if this circuit is random or not.



PROJECT 87. EVEN OR ODD



U1	74HC76	Q1	NPN	R1	10KΩ	R5	10KΩ	C1	3.3μF
U2	74HC00	Q2	PNP	R2	10KΩ	R6	10KΩ	C2	0.1μF
U3	324	Q3	PNP	R3	10KΩ	R7	4.7KΩ	C3	0.1μF
		Q4	NPN	R4	10KΩ			C4	0.1μF

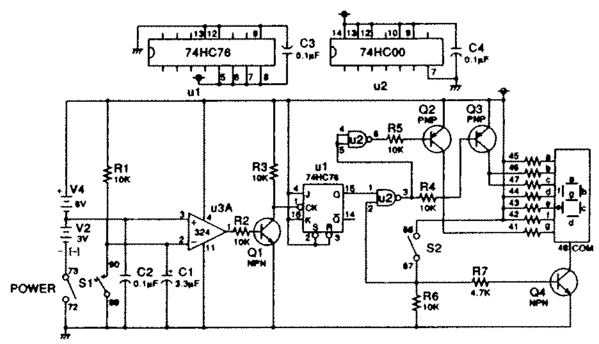
If you have a decision that comes down to either "even" or "odd", let this project decide for you!

After building this project, turn power ON and set the slide switch **S2** ON. You might see the letter **E** on the **LED display**. If not, press the key **S1** so the **E** appears on the **LED display**. Now set the slide switch **S2** OFF and press **S1** (not rapidly) a few times. Set the slide switch **S2** ON and watch the **LED display**. You see either **E** for even or **O** for odd appear.

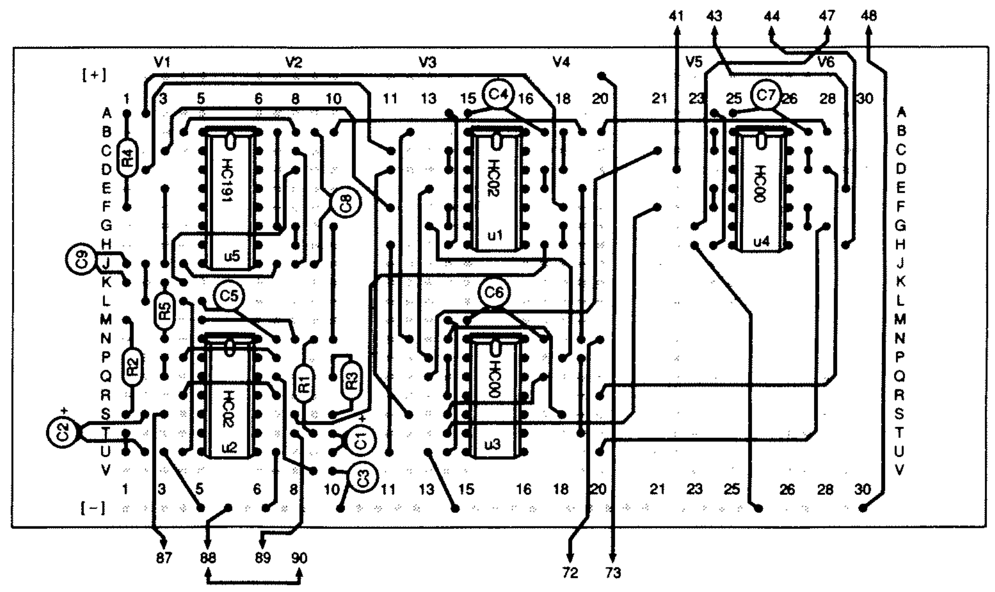
Look at the schematic for this project. You see that when you press **S1** and then release it you send and then stop current flow to a point called CK on the J-K flip-flop. CK stands for clock. It's the portion of the J-K flip-flop that controls whether the flip-flop sets or resets.

In most cases the clock signal is provided by an other electronic circuit (multivibrator, AND gate etc.) but here you provide it by pressing the key. When you set the slide switch **S2** ON, current can flow to one of the inputs of the NAND gates. The other input comes from Q of the J-K flip-flop. Depending on whether these inputs are 0 or 1 (remember how a NAND gate works?), certain segments of the **LED display** turns on. In turn this spells out **E** or **O** on the **LED display**.

Since you are providing the clock signal instead of an electronic circuit, repeat the same experiment you did in previous projects. Is this circuit any more or less random than the previous two circuits?



PROJECT 88. QUICK DRAW GAME



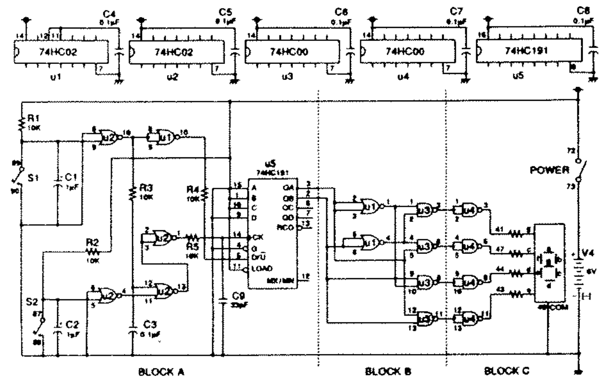
U1	74HC02	U5	74HC191	R4	10KΩ	C3	0.1μF	C7	0.1μF
U2	74HC02	R1	10KΩ	R5	10KΩ	C4	0.1μF	C8	0.1μF
U3	74HC00	R2	10KΩ	C1	1μF	C5	0.1μF	C9	33pF
U4	74HC00	R3	10KΩ	C2	1μF	C6	0.1μF		

Here's a game circuit with two switches; two players compete to be the first press of the key. Which key is pressed first is indicated by the rotation direction (clockwise or counterclockwise) of the display.

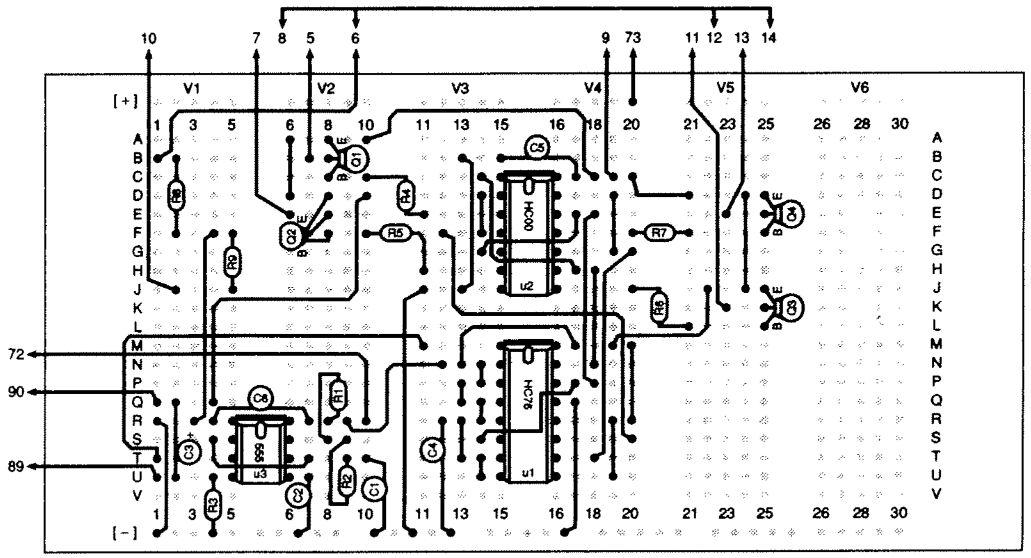
See the schematic. **S1** and **S2** in block A are the keys for the two players. IC 74HC191 is a 4-bit up/down counter. It works as an up or down counter, depending on which key is pressed first. (Don't worry, we'll explain you more about this IC later.) The output of IC is sent out from QA and QB and applied to the block B, which is what is known as decoder. Can you guess how it got this name? (Do you suppose it could have anything with to do with "decoding" the output of the counter?) The output of the decoder is sent to the **LED display** in block C.

Assemble the project, turn power ON, and see what happens to the **LED display** when you repeatedly turn **S1** ON and OFF. Do you see the display on the **LED display** turning clockwise? Now repeat turning **S2** ON and OFF, and the display turns counterclockwise.

Now you can play the game with a friend. Tell him to choose his switch and press it at a "Go" sign. If your key is **S1** and you pressed and released it faster than he did his (**S2**), the display turns clockwise, and that means you've won the game.

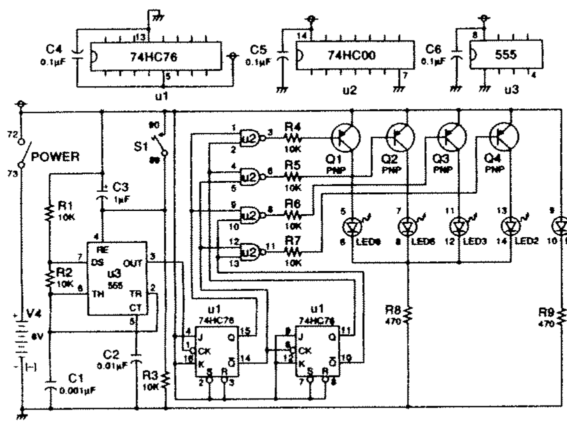


PROJECT 89. CLOSE-IN



U1	74HC76	Q3	PNP	R4	10KΩ	R9	470Ω	C5	0.1µF
U2	74HC00	Q4	PNP	R5	10KΩ	C1	0.001µF	C6	0.1µF
U3	555	R1	10KΩ	R6	10KΩ	C2	0.01µF		
Q1	PNP	R2	10KΩ	R7	10KΩ	C3	1µF		
Q2	PNP	R3	10KΩ	R8	470Ω	C4	0.1µF		

This is a project that's a bit different from the last few: This one too helps you make a decision... like whether to go to the left or right.



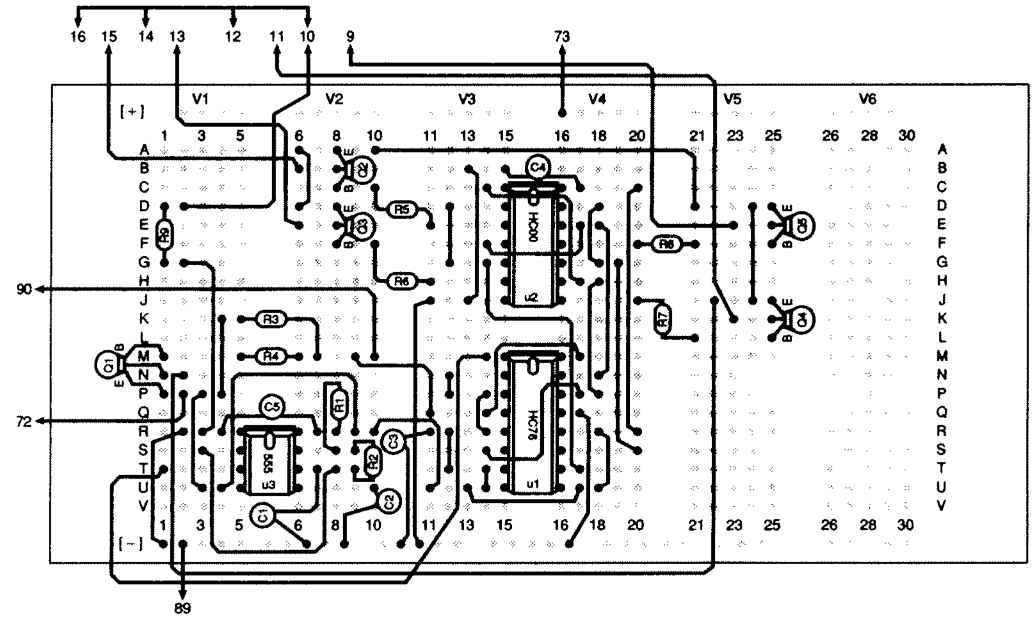
After you finish the wiring connections, turn power ON. You notice that **LED 4** lights up. Now press **S1**. You see **LEDs 2, 3, 5** and **6** light at what seems to be "half brightness". Now release **S1**. What happens?

You see **LED 4** and one more **LED** lights up. Which one was it? Now press and release the key..... and try it again..... and again. Does a different **LED** light up each time or does the same one light up over and over?

Look at the schematic. You notice that **LED 4** is a separate circuit of its own, not affected by the rest of the project. That's why **LED 4** stays on all the time. The rest of the project is the oscillator U3/counter/decoder arrangement we've seen in other projects. Again, can you figure out how the outputs of the flip-flop are combined in the NAND gates to light just one of the **LEDs**? Be sure to make a note of your guess.

You can also use this project in game. The winner could be whoever can get a **LED** to light closest to **LED 4**, for example.

PROJECT 90. ESP TESTER



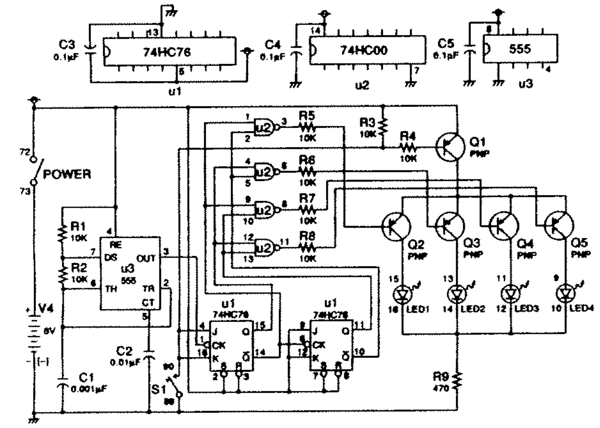
U1	74HC76	Q3	PNP	R3	10KΩ	R8	10KΩ	C4	0.1µF
U2	74HC00	Q4	PNP	R4	10KΩ	R9	470Ω	C5	0.1µF
U3	555	Q5	PNP	R5	10KΩ	C1	0.001µF		
Q1	PNP	R1	10KΩ	R6	10KΩ	C2	0.01µF		
Q2	PNP	R2	10KΩ	R7	10KΩ	C3	0.1µF		

Do you believe in ESP (extrasensory perception)? Before you answer, why not investigate the subject a little more using this project?

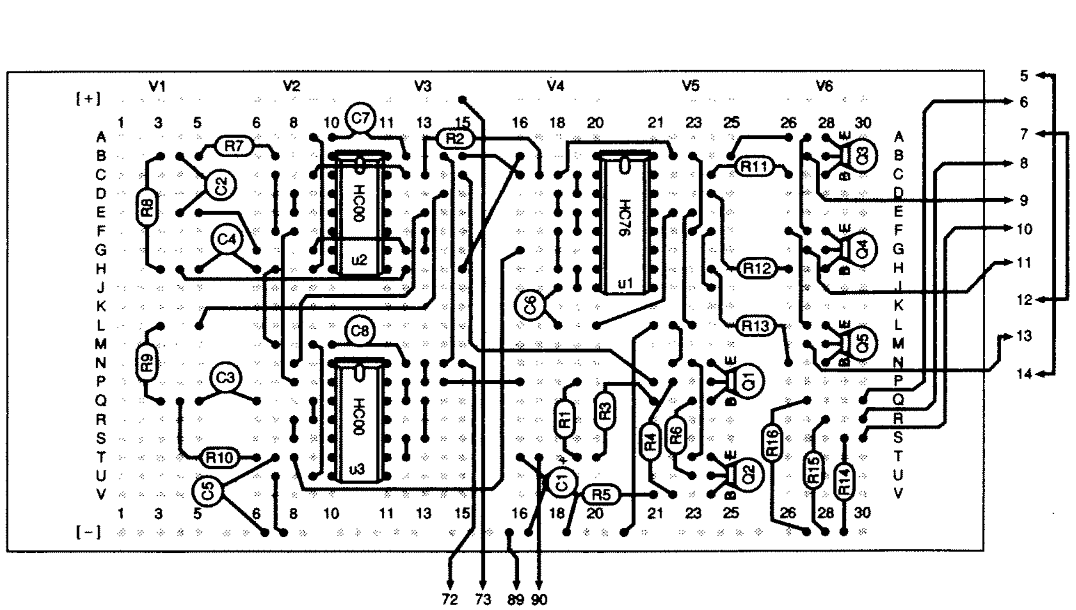
To use this project, turn power ON. Make a guess as to which **LED (1, 2, 3,** or **4)** lights up. Then press **S1**. Were you right? Try this 100 times, keeping a record of your guess and actual result.

You should have no trouble by now recognizing the different sections that make up this project. The oscillator U3 feeds its output into the flip-flops, which together make up a counter circuit. The outputs of the flip-flop go to the four NAND gates and flow to the transistors, which are connected to the four **LEDs**. The four NANDs make up the decoder.

What kind of results did you get when you tried the experiment 100 times? If the circuit is truly random, each of the four **LEDs** should have lit 25 times each. Don't be surprised if some lit more often than others; this is normal for only 100 times. How often were you able to guess which **LED** was going to be lit? Sheer luck would let you guess correctly about 20 or 30 times. You should not decide you have ESP until you can correctly guess 50 out of each 100 trials..... in several tests. Ask your teacher or librarian for some good books about ESP. Read some of these books before making a final decision on whether you have ESP!



PROJECT 91. THE LIGHT FANTASTIC



U1	74HC76	Q5	PNP	R7	22KΩ	R14	470Ω	C5	0.001μF
U2	74HC00	R1	4.7KΩ	R8	47KΩ	R15	270Ω	C6	0.1μF
U3	74HC00	R2	4.7KΩ	R9	22KΩ	R16	270Ω	C7	0.1μF
Q1	NPN	R3	22KΩ	R10	100KΩ	C1	100μF	C8	0.1μF
Q2	NPN	R4	4.7KΩ	R11	10KΩ	C2	0.1μF		
Q3	PNP	R5	2.2KΩ	R12	10KΩ	C3	0.1μF		
Q4	PNP	R6	33KΩ	R13	10KΩ	C4	0.001μF		

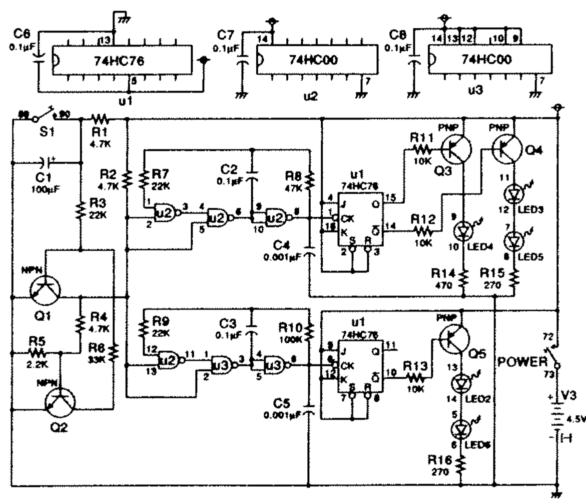
In the last projects we used NAND gates to combine the outputs of flip-flops. There's no law that says NAND gates must always follow flip-flops in a circuit - just look at the schematic for this project and see.

You notice that this project uses a two transistor circuit to drive the NAND gates. Have you seen this particular circuit in other projects? (Hint- it's a type of oscillator.) The output is combined by the NAND gates and then goes to the flip-flops.

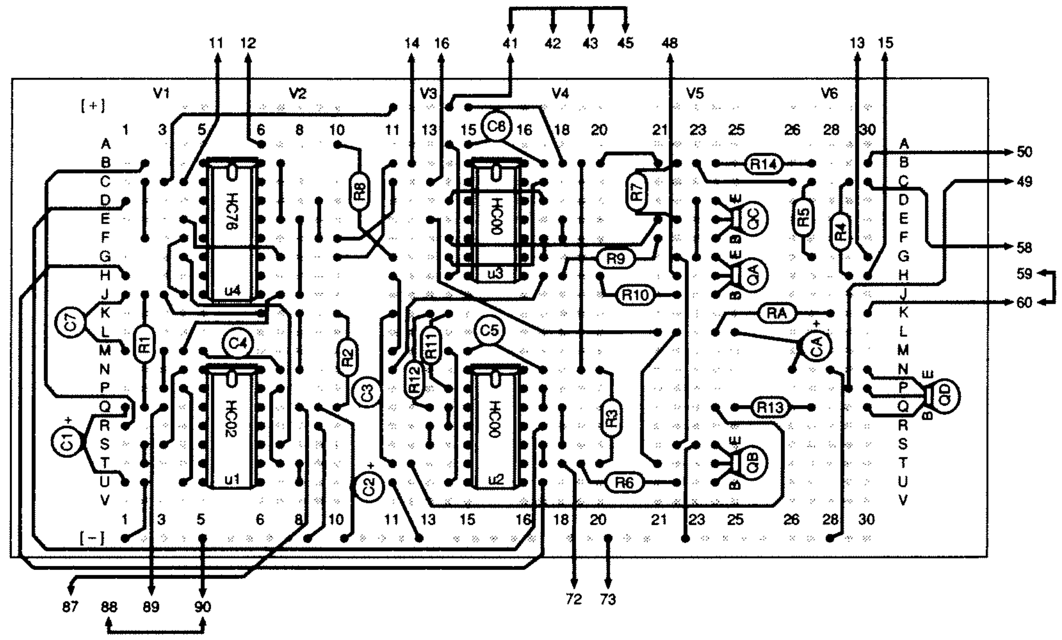
Note that the NAND gates control the clock signal for each flip-flop, meaning that the NAND gates really control when the different **LEDs** light up.

After wiring this project, turn power ON and press **S1**. The **LEDs** begin lighting up in certain patterns. Now release **S1**. Notice which **LED** or **LEDs** remain lit. Try pressing and releasing **S1** several times - is the result truly random?

Using what you've found out about flip-flops so far, can you tell which outputs are 0 and which are 1 when various **LEDs** are lit (now that should be easy!). Once you figure that out, put your thinking cap on and try to figure out how the output of the oscillator circuit combines in the NAND gates to produce the clock signal for the flip-flops! (Don't feel bad if you have trouble doing that ..... it isn't easy to keep track of what goes on inside those NANDs. But give it your best shot and make a note of what you think happens.....)

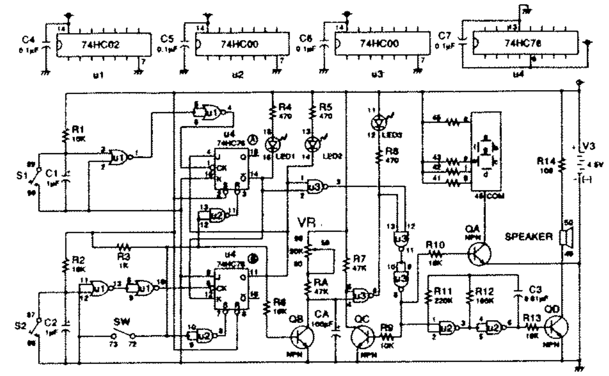


PROJECT 92. SHOOTING GAME



U1	74HC02	QD	NPN	R6	10KΩ	R13	10KΩ	C5	0.1μF
U2	74HC00	RA	47KΩ	R7	47KΩ	R14	100Ω	C6	0.1μF
U3	74HC00	R1	10KΩ	R8	470Ω	CA	100μF	C7	0.1μF
U4	74HC76	R2	10KΩ	R9	10KΩ	C1	1μF		
QA	NPN	R3	1KΩ	R10	10KΩ	C2	1μF		
QB	NPN	R4	470Ω	R11	220KΩ	C3	0.001μF		
QC	NPN	R5	470Ω	R12	100KΩ	C4	0.1μF		

Here's a shooting game that can sharpen your sense of speed. You can play this game with a friend. It has three functions, the function to judge the player's speed, the function to judge the start signal, and the function to judge the foul signal.

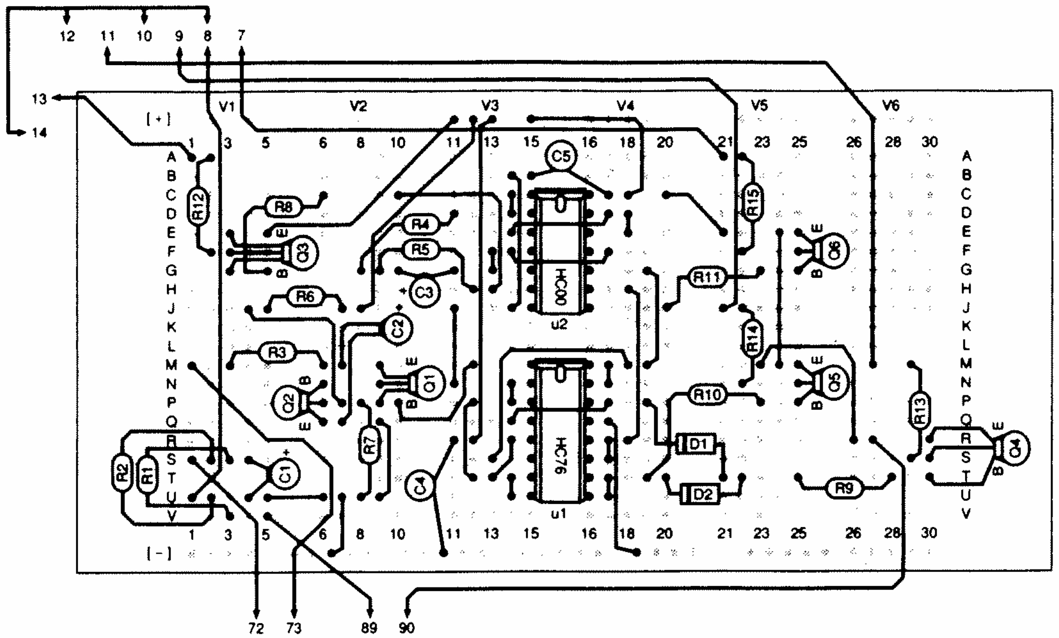


Let's take a look at the schematic and see how to play this game. The two flip-flops judge the speed that determine which key is pressed first. If **S1** is pressed first, **LED 1** lights up. If **S2** is pressed first, **LED 2** lights up.

Turn on the **slide switch**. After a short period, **LED 3** lights. As soon as you see it lights, press **S1** or **S2**.

If a player press **S1** or **S2** before the **LED 3** lights up, the **LED display** indicates the letter **F** (meaning "foul") and the **speaker** gives the foul sound. When this happens, release the key, turn power OFF, and then switch it ON again. The game re-starts. You can adjust the time lag between power on to the start signal generation (**LED 3** ON) by rotating the **control volume**.

PROJECT 93. MARCHING LEDS



U1	74HC76	Q5	PNP	R5	150KΩ	R11	10KΩ	D2	Si
U2	74HC00	Q6	PNP	R6	470Ω	R12	470Ω	C1	100μF
Q1	PNP	R1	10KΩ	R7	2.2KΩ	R13	470Ω	C2	1μF
Q2	NPN	R2	470KΩ	R8	10KΩ	R14	470Ω	C3	1μF
Q3	PNP	R3	100KΩ	R9	10KΩ	R15	470Ω	C4	0.1μF
Q4	PNP	R4	1.5MΩ	R10	10KΩ	D1	Si	C5	0.1μF

Make way for the parade of marching **LEDs**! This project lets you send **LEDs** "moving" across your kit as you press **S1**.

When you finish the wiring, **LED 2** lights and starts flickering (you might also have other **LEDs** light; this is normal). Set the **select switch** up and keep pressing **S1**. First **LED 3** lights, followed by **LEDs 4** and **5**. As you keep pressing **S1**, you'll have **LEDs 3, 4** and **5** lighting in turn. It looks just like a parade of **LEDs** marching down the street.

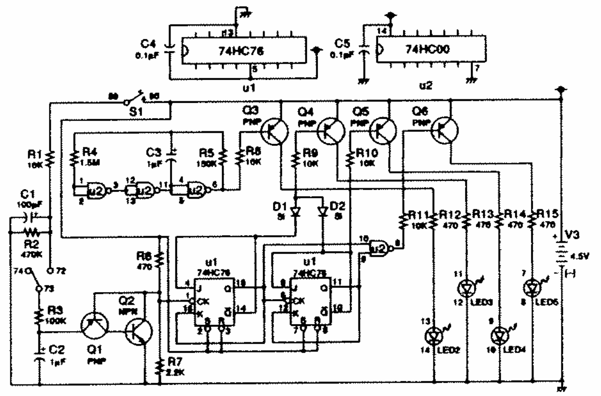
While pressing **S1**, try quickly setting the **select switch** down. What happens?

The **LEDs** come to a screeching halt. The **LEDs** are either lit or off just like they were the instant you set the **select switch** down. Now set the **select switch** again up. The **LEDs** again starts "marching."

While you have the **select switch** at the up position, stop pressing **S1** and watch what happens to the **LEDs**.

Can you guess why this takes place?

This project also uses the "flip-flop" IC. You can easily see why this device is so useful in electronics!



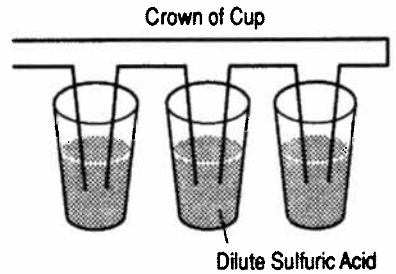
Coffee Break

Frog legs led us to Electric Current Flow

The static electricity that the ancient Greek, Thales, discovered still could not be used in the daily lives of mankind. That was because static electricity does not flow constantly. So, how was this energy form harvested as a "constant current electricity" that can be used?
Well, in the 18th century, an Italian anatomist called Luigi Galvani noticed that the muscles of frog legs contract when they are exposed to electric sparks from charged metal objects. He also found that the muscles twitch when the frog's spinal cord and nerves are touched by the tips of two different kinds of metal that are connected to each other. When he saw this happen, he thought electricity was generated in the frog's body, and named this electricity "animal electricity." Was it really generated inside the frog's body, though?

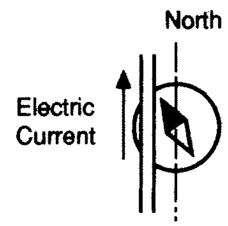
Another Italian, a physicist and chemist called Allessandro Volta, carried out Galvani's work further, and invented the "Crown of Cup," which became the basis for today's batteries. So, it was the doctors and chemists who originally pioneered the science of electricity. Of course, there were no electricians back then, not even electric lamps. In fact, the basis for direct currents was not established until only 200 years ago. Moreover, even then, it took another 80 years until Edison invented the first carbon filament lamp.

Volta correctly suspected that the secret of electricity was in the two different kinds of metal instead of the frog's body, and conducted the following experiment. First, he connected a piece of tin foil to a silver coin. Then, as soon as he placed the tin foil on his tongue, and stuck the silver coin under his tongue, he sensed a strong acidic taste. In other words, an electromotive force (voltage) was generated, and an electric current ran through his tongue (so, it was not the frog that generated electricity after all).
Volta poured dilute sulfuric acid into a cup, and dipped copper and zinc plates into the cup. This was the renowned "Crown of Cup," the first battery in history (we now have constant electric current, at last).

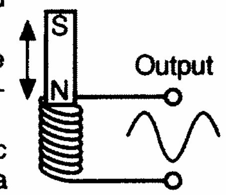


♦ Electricity and magnetic force are like good friends; always together

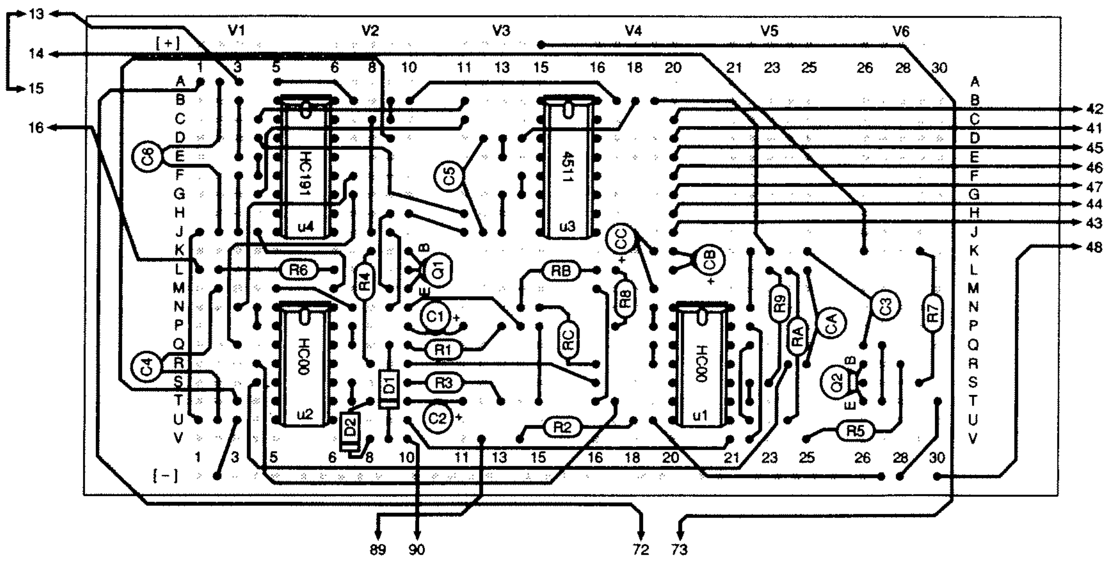
In 1819, Hans Christian Oersted of Denmark discovered that a compass needle shakes when an electric current runs through it. The consequent attention that focused on the relationship between electricity and magnetic force developed into today's science of electromagnetism. You probably also witnessed the needle shaking when running a current through it when you conducted an electricity experiment using a battery, light bulb, and a compass. The result of Oersted's experiment (a magnetic force was generated when an electric current ran through) led to a bright idea by an Englishman called Michael Faraday. What do you think his idea was?



Think about it for a moment. Your hint is "electricity and magnetic force are like good friends; they are always together."
If you guessed right, and you were living in the first half of the 19th century, your name would have gone down in history instead of what is now called "Faraday's Electromagnetic Induction Law."
This is how Faraday thought it out. "If a magnetic force is created when an electric current runs through something, does it not follow that electricity is produced when a magnet is moved?" So, he made a coil and moved a magnetic rod in and out of it. Sure enough, an electric current ran through the coil, and moreover, the faster he moved the magnet, the larger the induced current became. In addition, the induction current also became larger when he increased the number of windings of the coil. (So, now you know that electricity and magnetism are intricately related.)



PROJECT 94. ELECTRONIC DICE



U1	74HC00	RA	330KΩ	R4	10KΩ	CA	0.1μF	C4	0.1μF
U2	74HC00	RB	10KΩ	R5	10KΩ	CB	10μF	C5	0.1μF
U3	74HC4511	RC	4.7KΩ	R6	470Ω	CC	10μF	C6	0.1μF
U4	74HC191	R1	47KΩ	R7	470Ω	C1	100μF	S1	Si
Q1	NPN	R2	100Ω	R8	22KΩ	C2	100μF	S2	Si
Q2	NPN	R3	22KΩ	R9	22KΩ	C3	0.1μF		

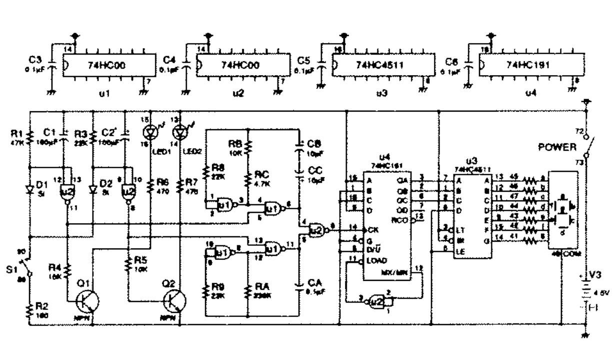
Here's electronic dice that can display the numbers **1 - 6** using the display.

As you see in the schematic, the electronic dice use a counter IC to count the number of pulses, and displays the numbers **1 - 6** on the display with the aid of a decoder IC.

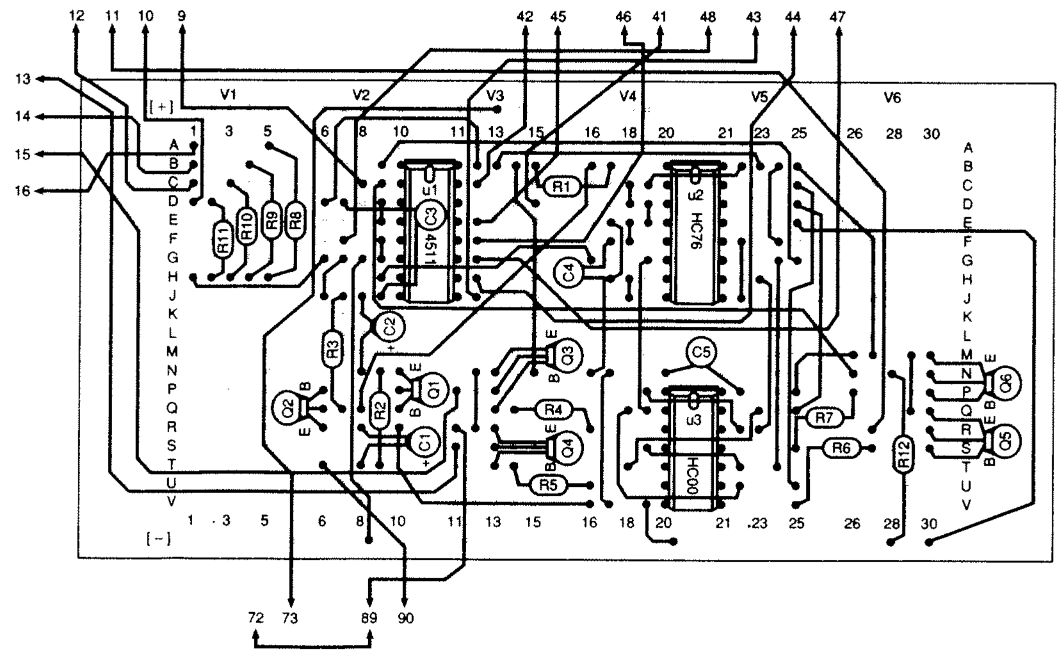
Two circuits are used to generate pulses. They begin generating pulses from the moment when you release **S1** after pressing it, and continue generating pulses during the time determined by C1 and R1 and by C2 and R3.

These two oscillators have different frequencies, and you can reproduce the natural movement of ordinary dice by staggering the pulse generating time of the two oscillators.

When you finish building this project, turn power ON and press **S1**. The numbers on the **LED display** changes swiftly and one of the numbers **1 - 6** is finally displayed.



PROJECT 95. ELECTRONIC ROULETTE

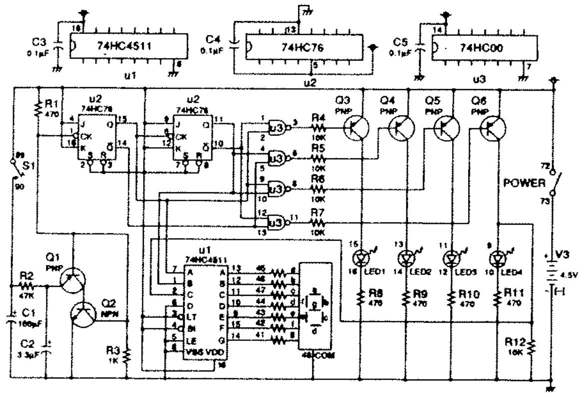


U1	74HC4511	Q4	PNP	R4	10KΩ	R10	470Ω	C1	100μF
U2	74HC76	Q5	PNP	R5	10KΩ	R11	470Ω	C2	3.3μF
U3	74HC00	Q6	PNP	R6	10KΩ	R12	10KΩ	C3	0.1μF
Q1	PNP	R1	470Ω	R7	10KΩ			C4	0.1μF
Q2	NPN	R2	47KΩ	R8	470Ω			C5	0.1μF
Q3	PNP	R3	1KΩ	R9	470Ω				

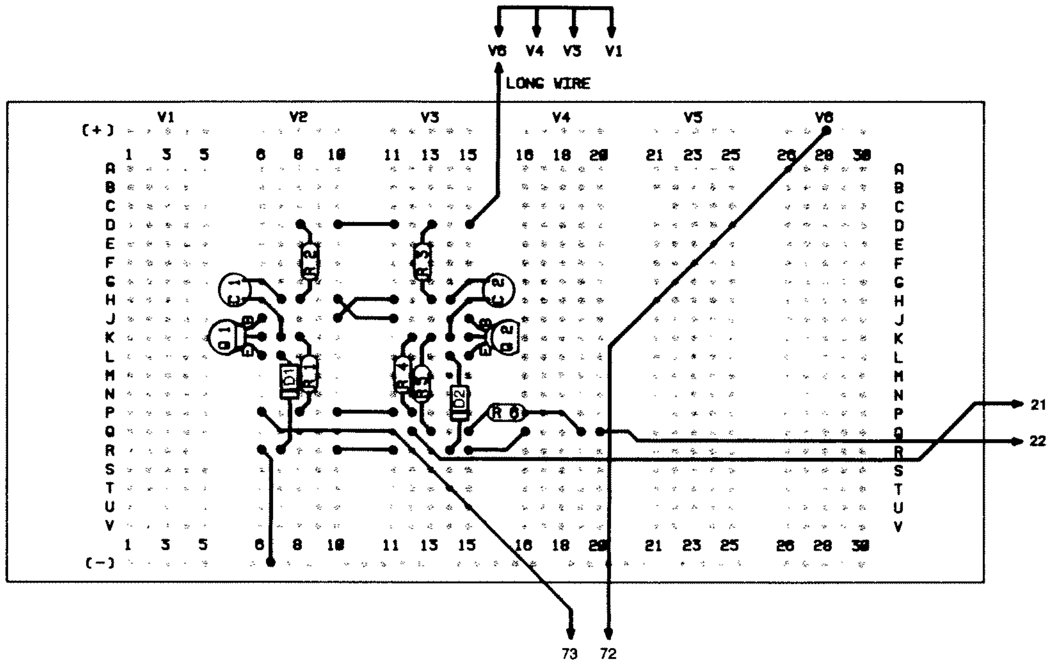
You'll find the general principle behind this circuit the same as the previous three. Be careful in constructing this one - there are many wiring connections. Take your time and double-check your work. When you're finished, turn power ON and press **S1**. Hold it down for a few seconds. You'll see the four **LEDs** flash off and on rapidly. On the display you'll see the numbers **1** through **4** indicated in order very rapidly. Now release **S1**. The **LEDs** and **LED display** "slow down" and finally stop at one **LED** and its corresponding number.

As you might suspect, the two transistors generate the clock signal for the J-K flip-flops. Note from the schematic that the clock signal for the second J-K flip-flop comes from output $\overline{Q}$ of the first flip-flop. The combined outputs of the J-K flip-flops then are applied to the NAND gates, which in turn light and turn off the **LEDs**. As you might also suspect, the discharge of the 100μF capacitor controls how long the clock signal is generated.

Notice how many J-K flip-flop ICs and NAND gate ICs were used in this circuit. You see, there are actually two J-K flip-flops in the Dual J-K Flip-Flop IC and four NAND gates in the Quad 2-input NAND IC. That's what the "integrated" means in integrated circuit - there are actually several circuits inside each IC!



PROJECT 96. MEET THE VCO



Q1	NPN	R1	1KΩ	R4	1KΩ	C1	0.047µF	D1	Si
Q2	NPN	R2	10KΩ	R5	100KΩ	C2	0.01µF	D2	Si
		R3	47KΩ	R6	470KΩ				

VCO?? What's that??? No, actually this is not the first time you meet the VCO....

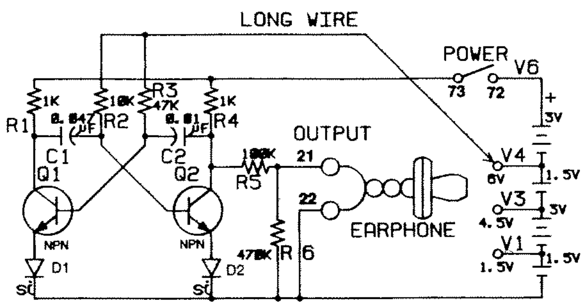
"VCO" is short for "voltage controlled oscillator." The name alone should give you a big hint how this project works.

When you finish the wiring connections, turn power ON. Connect the earphone to terminals **21** and **22**. Take the long wire and attach it one end of 47K resistor as illustrated. Now connect the other end of the wire to various point of the voltage source... You'll find the sound varies by changing the long wires with different value of voltages. What effect does changing the voltage have on the sound you hear?

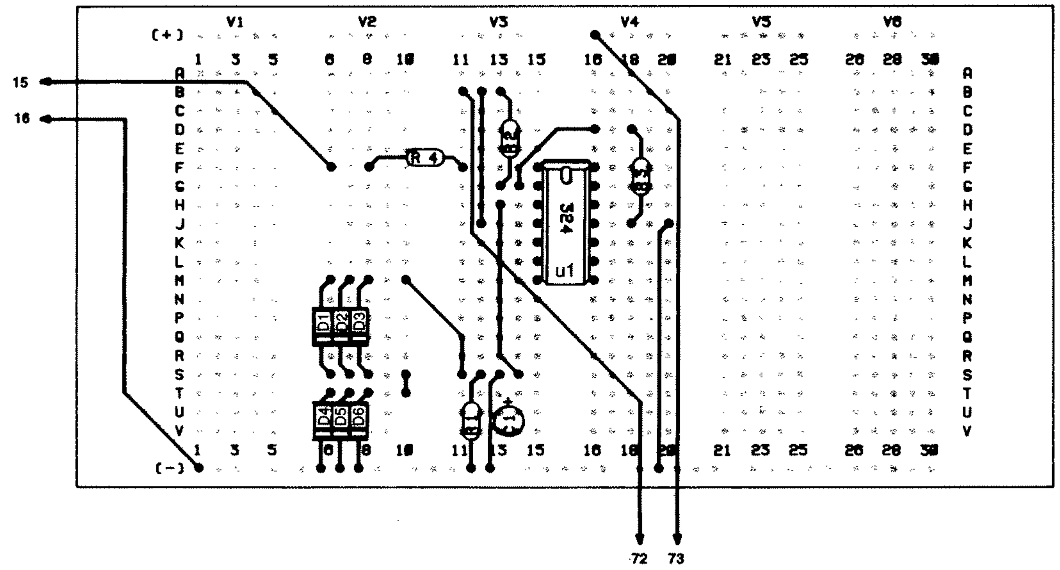
This project use the familiar astable multivibrator that we've used in many of our other projects. The frequency of other types of oscillators is also sensitive to changes in voltages.

VCO circuits have a variety of uses. They can let you hear when the voltage from a circuit or power source changes. They can warn when the voltage from a source gets too high or too low. And they are often found in electronic musical instruments and devices.

Try to think of some other applications for VCO circuits. You might also want to substitute the **LED** in place of the earphone to "see" what you've been hearing.



PROJECT 97. SILICON DIODE SOLAR CELL



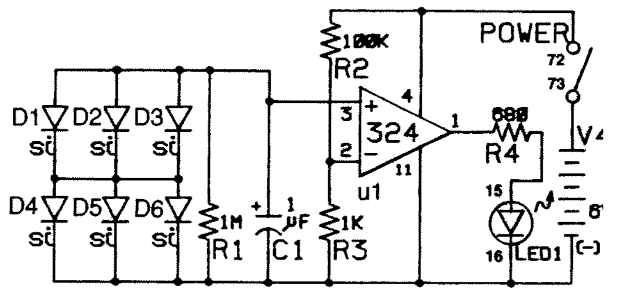
U1	324	R3	1KΩ	D1	Si	D4	Si
R1	1MΩ	R4	680Ω	D2	Si	D5	Si
R2	100KΩ	C1	1µF	D3	Si	D6	Si

We all know that the brilliant rays of the sun pouring down on the ground have energy (because we feel warm when we go out into the sun), but have you ever thought of turning this energy into electricity? Maybe it sounds like a dream to you, but this dream comes true if a solar cell is used, and that's what we're going to build in this project! The silicon diode in your kit is a semiconductor just as any solar cell, so we'll use it to generate electricity.

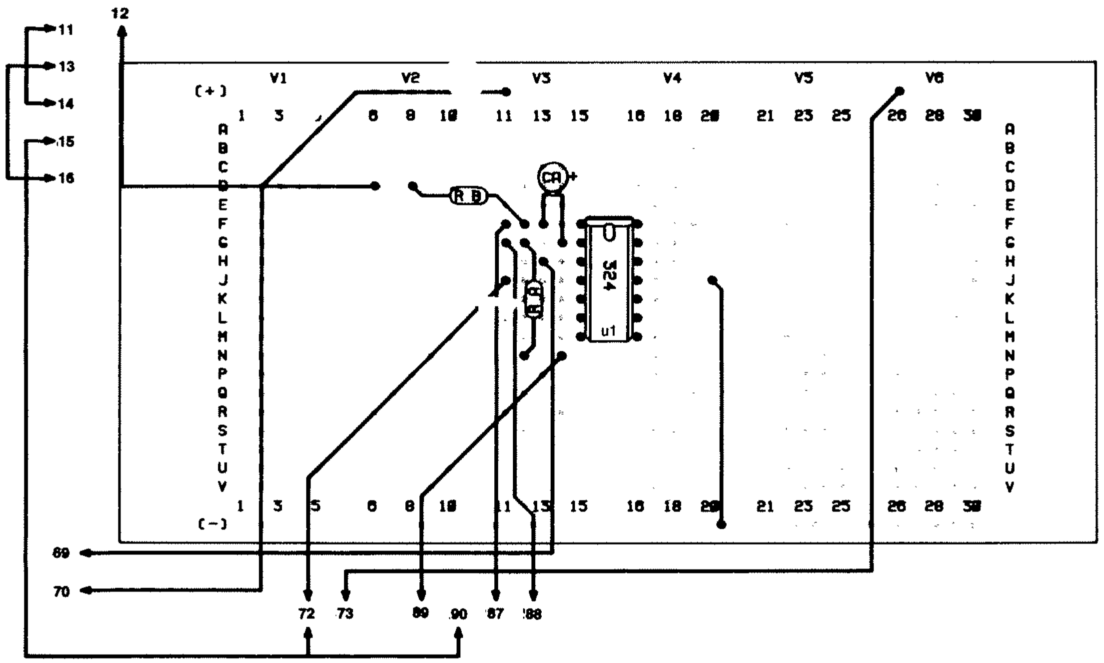
The circuit for this experiment uses "comparator," a device for voltage comparison, to check if electricity is actually created in the silicon diode. IC 324 acts as a comparator, and the reference voltage for comparison is set at 60 mV. So, when electricity with a voltage of more than 60 mV is generated in the silicon diode, it lights up **LED 1** and tells you electricity is actually produced.

Now, see what happens when you expose the silicon diode to the sun. If **LED 1** lights up, it means that you've succeeded in turning the energy of the sun into electricity.

Try light energies from other sources such as a incandescent lamp and a fluorescent lamp, and see ho the voltage changes according to the distance an brightness of each light source.

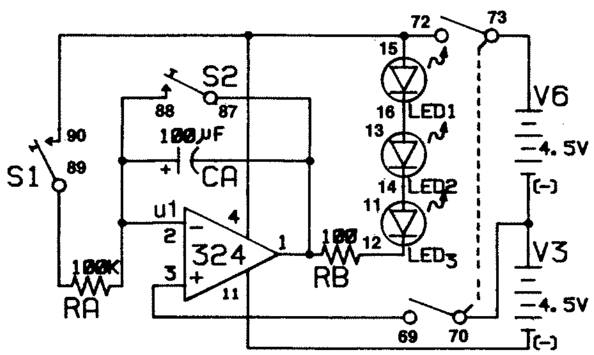


PROJECT 98. INTEGRATING CIRCUIT



U1	324	RA	100KΩ	CA	100μF
		RB	100Ω		

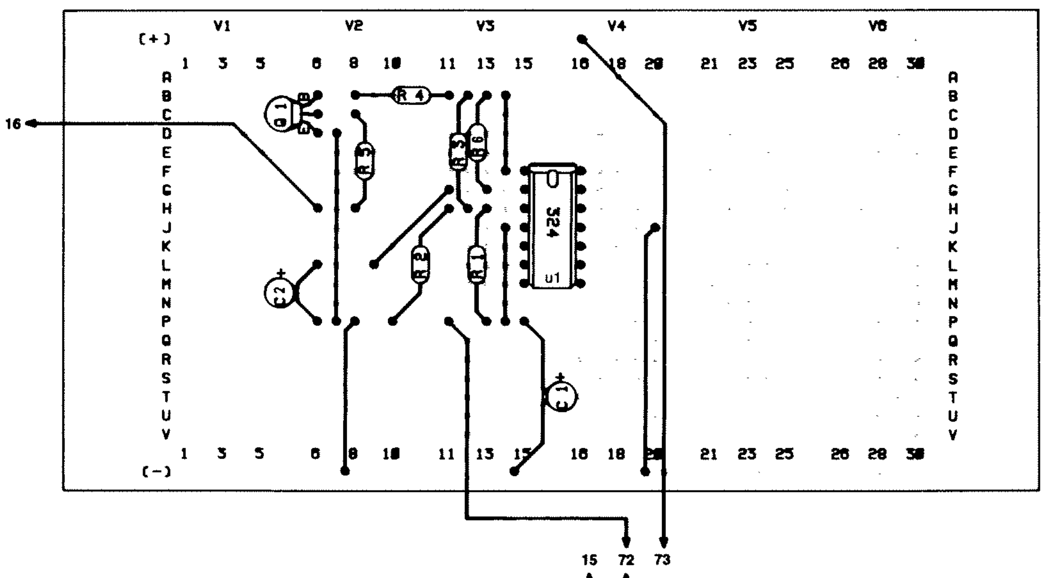
You'd think that any **LED** instantly gives a flash when you switch it ON. But by applying some knowledge of electronics, you can light it up gradu-a-lly... It's exciting to see an **LED** getting brighter slowly while holding down the key, and you're going to feel that excitement in this project!



The circuit that realizes this interesting performance is called Miller integrating circuit. Capacitor CA shows a very high nominal value because of the function of IC 324. When you press **S1** ON, the capacitor is charged slowly through resistor RA and the **LED** becomes bright slowly. **S2** is used to discharge the electric charge stored in the capacitor.

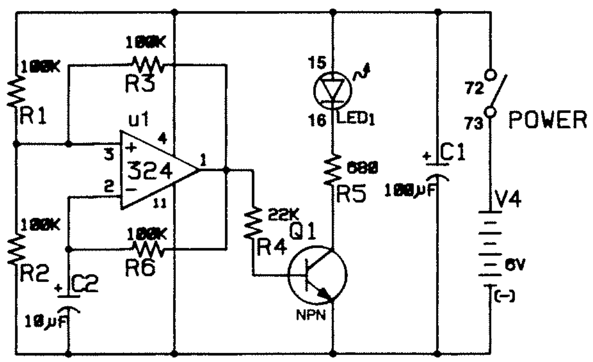
To use this project, press **S2** after switching power ON to discharge the electric charge from capacitor, then keep holding down **S1**, **LED** becomes brighter little by little. It reaches the maximum brightness in about five seconds. Press **S2** to discharge capacitor and then hold down **S1** to repeat the experiment.

PROJECT 99. ASTABLE MULTIVIBRATOR USING OP AMPLIFIER



U1	324	Q1	NPN	R3	100KΩ	R6	100KΩ		
		R1	100KΩ	R4	22KΩ	C1	100μF		
		R2	100KΩ	R5	680Ω	C2	10μF		

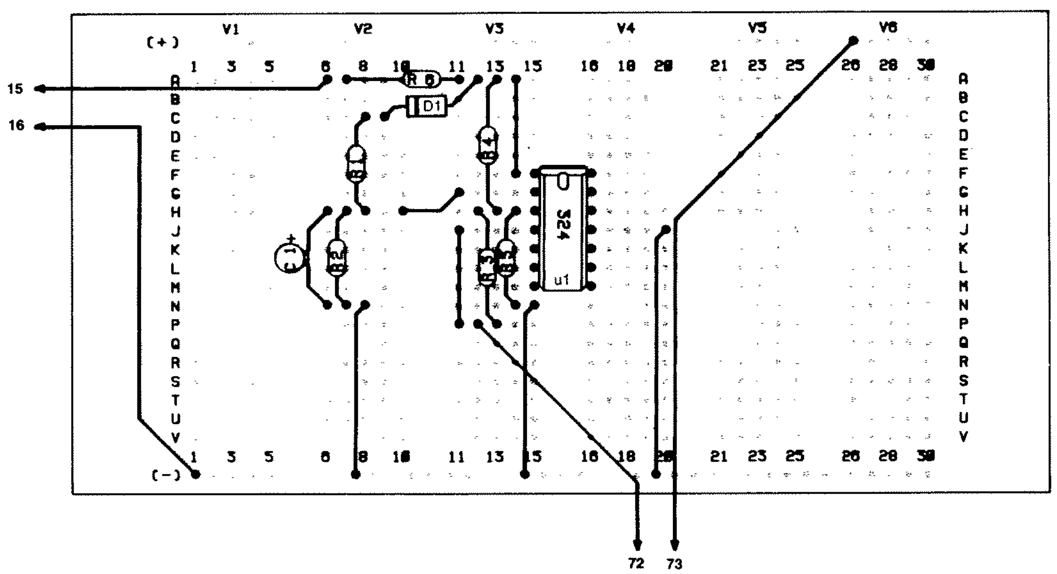
We've already learned how a multivibrator circuit works. In this project, we're going to make an **LED** flash on and off with an astable multivibrator using an operational amplifier. The astable multivibrator is used in **LED** flashers and similar electronic devices.



IC 324 acts as the astable multivibrator. Remember that the output of a multivibrator is a square wave? The circuit shown in the schematic has an oscillating frequency of about 0.5 Hz and makes **LED 1** light up and go out repeatedly. This continues over and over again at intervals of about 2 seconds. (That means the half of the intervals in a second -- 0.5 Hz, OK?) The output of the multivibrator is sent to transistor Q1. Q1 amplifies current to light up **LED 1**.

This experiment is super-easy ..... just switch power ON, and **LED 1** flashes on and off. See how the oscillating frequency changes by changing C (10μF) and R (100 k ohms) to different values.

PROJECT 100. PULSE GENERATOR



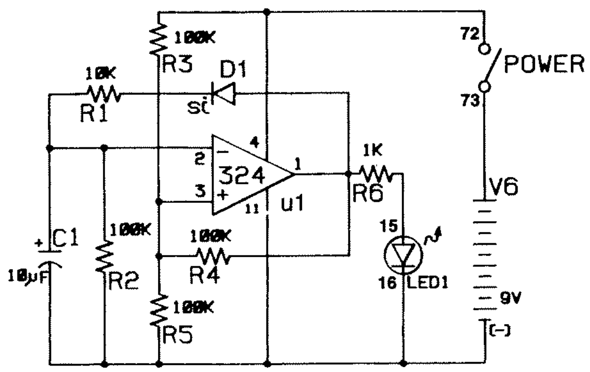
U1	324	R1	10KΩ	R4	100KΩ	C1	10μF
		R2	100KΩ	R5	100KΩ	D1	Si
		R3	100KΩ	R6	1KΩ		

This is an **LED** flasher that makes an **LED** flash on and off rapidly. Using this flasher, you make the **LED** flash for a moment and go out, flash again and go out, and repeat this cycle over and over.

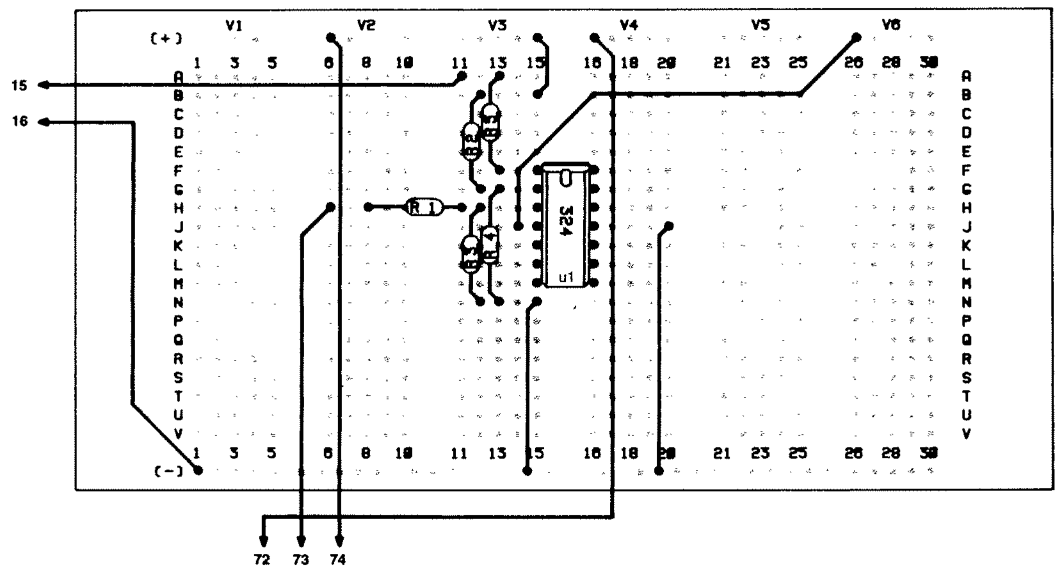
This is a pulse generator which works in the following way. When you switch power ON, an output voltage is produced and C1 is charged through R1, and **LED 1** stays on while C1 is charged. As the charging of C1 goes on, the output voltage goes down to 0. Then the current stops flowing to C1 and **LED 1** goes out. How this happens? Wait till you build the next project.

Then, the charge stored in C1 is discharged through R2. When C1 is discharged to a certain extent, the circuit restores the original power-ON state. This cycle repeats itself continuously.

Do you get **LED 1** winking all right? Then, see how its flickering changes when you use different values for R1 and R2. You'll notice that it goes on and off slowly when R2 is increased, and fast when R2 is decreased.



PROJECT 101. COMPARATOR



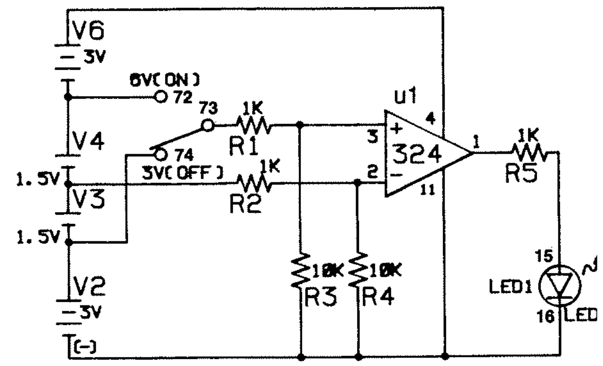
U1	324	R1	1KΩ	R4	10KΩ
		R2	1KΩ	R5	1KΩ
		R3	10KΩ		

A comparator is a device which has its own reference voltage and compares it with any external input voltage to tell you if the input voltage is higher or lower than the reference voltage.

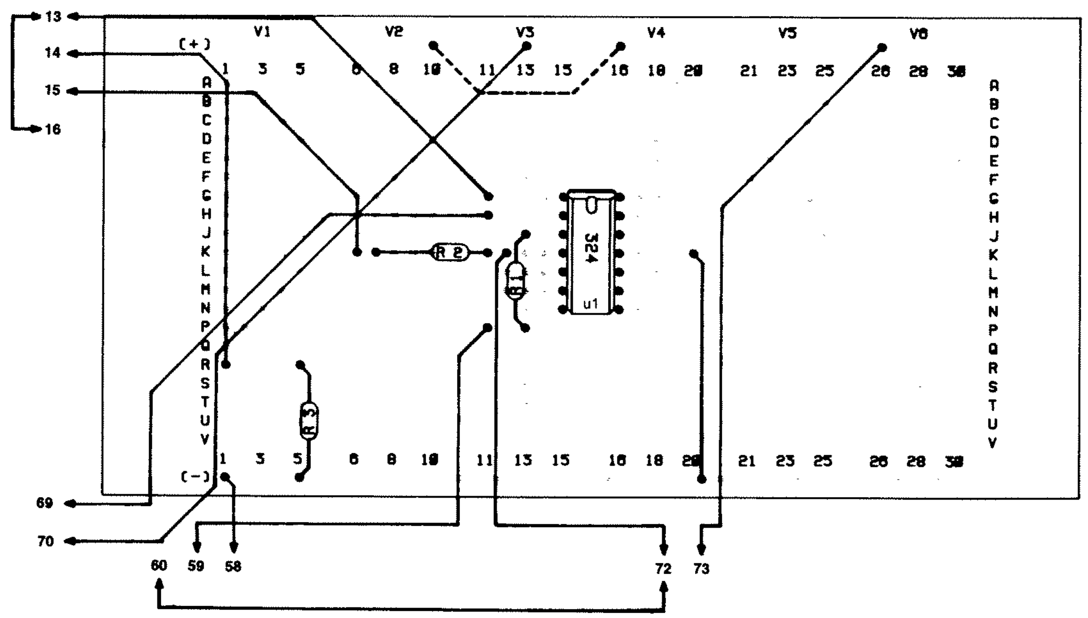
The schematic diagram shows you that this reference voltage is about 4.5 V and is applied to the pin 2 of IC 324. Input voltage is supplied to the pin 3 of IC 324, and the **LED** lights up if this voltage is higher than the reference voltage, and goes out if it is lower than the reference voltage.

Now, let's see how this comparator works. Slide the **select switch** down to supply 3V. The **LED** doesn't light up, does it? Now slide the **select switch** up. This makes the input voltage to pin 3 to 6V. The comparator outputs a voltage as the input voltage is now higher than the reference voltage, and the **LED** lights up.

Go back to previous project and see how the pulses are output.



PROJECT 102. EXPERIMENT OF COMPARATOR

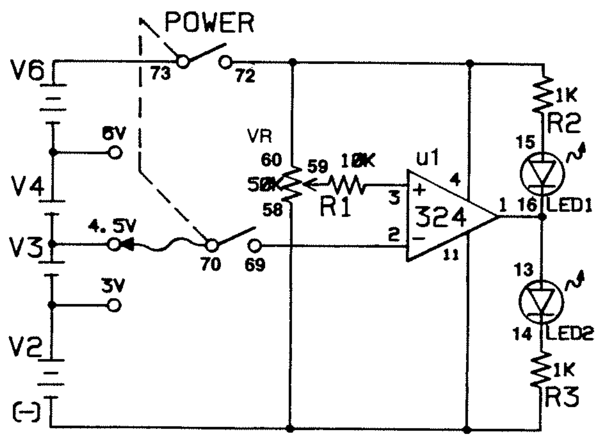


U1	324	R1	10KΩ	R2	1KΩ
R3	1KΩ				

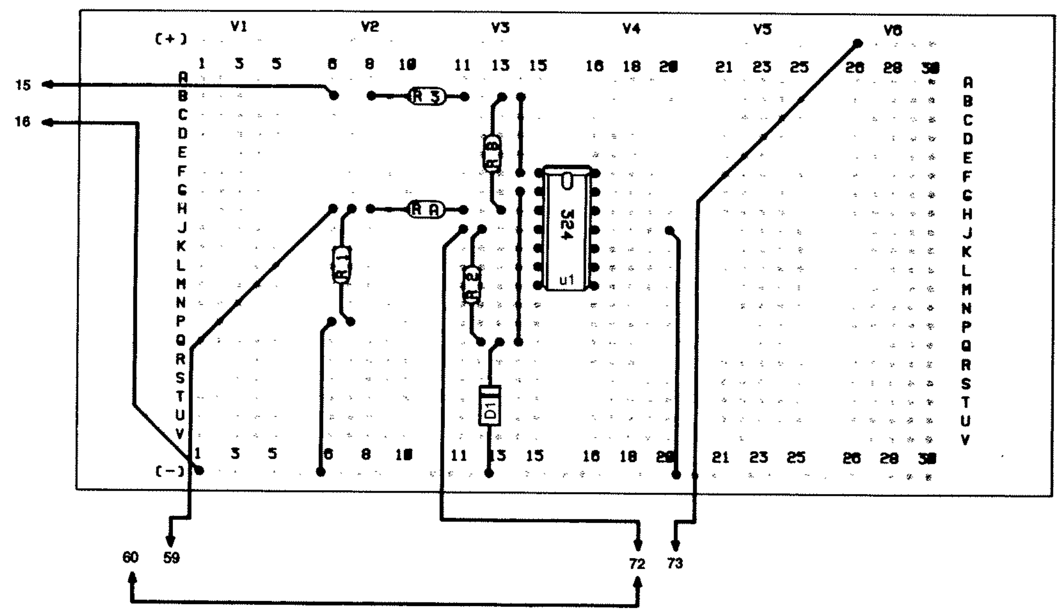
Here's another project to see how the comparator works.

You can adjust the input voltage with the **control volume**. When you turn it counterclockwise, **LED 1** lights up and **LED 2** goes out. Now turn the **control volume** clockwise gradually... **LED 2** lights up and **LED 1** goes out at some point. At this point, the input voltage is the same as the reference voltage.... 4.5V.

Now, you've got the knack of the comparator operation. So, see what happens when the reference voltage is changed to 3 V or 6 V. Did you notice the change in the control setting at which the **LEDs** take turns lighting?



PROJECT 103. COMPARATOR WITH HYSTERESIS

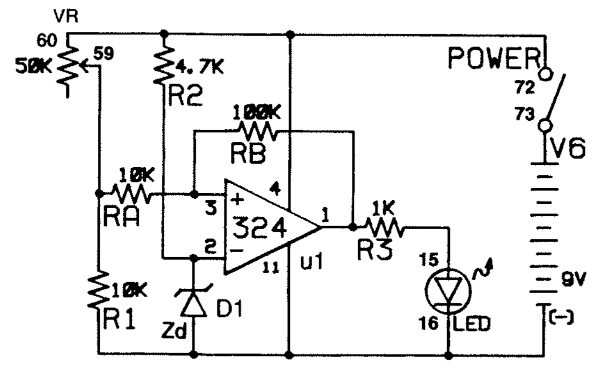


U1	324	RA	10KΩ	R1	10KΩ	R3	1KΩ
D1	Zd	RB	100KΩ	R2	4.7KΩ		

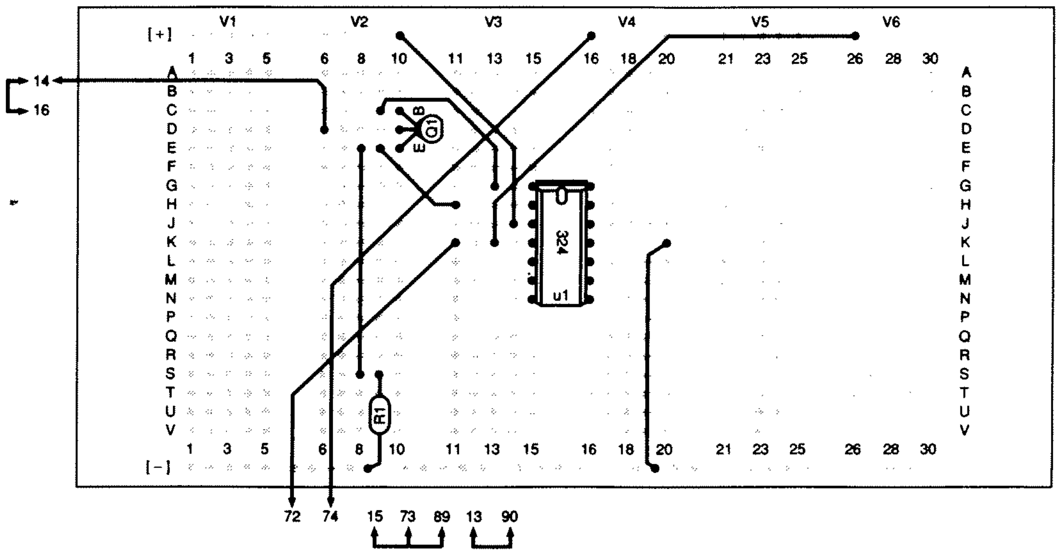
Do you always take the same route on your way to and back from school? Maybe you do so usually, but you must certainly have taken some other route for a change from time to time, haven't you? Well, current which can be made to flow back by a route different from the going route in a circuit, and we're going to find out how in this project. The difference between the going and returning routes is called the "hysteresis."

In this experiment, we're going to build a comparator with the hysteresis characteristic. First, turn the **control volume** fully counterclockwise, and turn the power ON. The **LED** goes out. Now, turn the **control volume** clockwise, and the **LED** lights up at some point. Note this point.

Next, turn the **control volume** clockwise and then slowly back counterclockwise, and check where the **LED** goes out. Did you notice the difference between the point where the **LED** lights up and the point where it goes out? That's the hysteresis, and its width becomes larger when RA is increased or RB is decreased.



PROJECT 104. CONSTANT CURRENT SOURCE BY OP AMPLIFIER



U1	324	Q1	NPN	R1	1KΩ

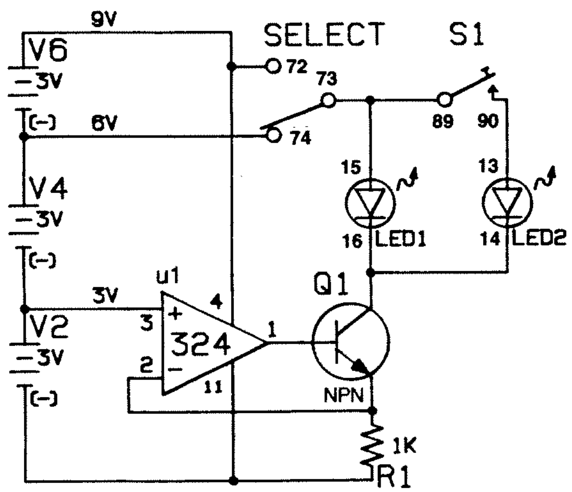
In this project, we're going to make a constant current circuit, using the Operational Amplifier and the transistor. This circuit can maintain a constant current not just when the source voltage is changed but also when the load is changed.

Look at the schematic. When the current is about to change, the voltage across R1 changes. This is fed back to the operational amplifier. The output of the operational amplifier changes in accordance with the feedback signal, which in turn controls the base voltage of the transistor Q1 to maintain the current constant.

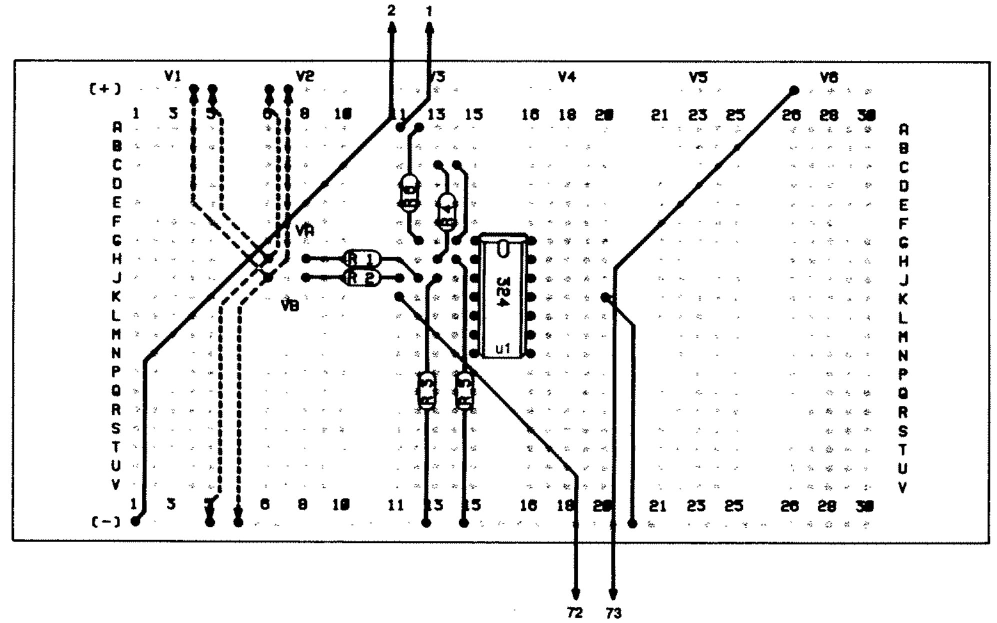
Let's get to the experiment. Set the **select switch** to 9V(up) side first, and press **S1** ON and OFF while watching the brightness of **LED 1**. **LED 1** becomes dimmer when **S1** is ON. Now, set the **select switch** to 6 V, with **S1** turned OFF.

As you've perhaps guessed already, when **S1** is turned ON and OFF, the current flows to both **LED 1** and **LED 2**, thereby increasing the load to the circuit. The transistor Q1's current tends to increase, but it is held constant as explained above. Because the load increases but current doesn't, **LED 1** becomes dimmer.

When the power is switched over between 6 V and 9 V, the current should change because of the voltage change. However, it is again kept constant, so the **LED** brightness remains the same.



PROJECT 105. NON-INVERTING ADDER



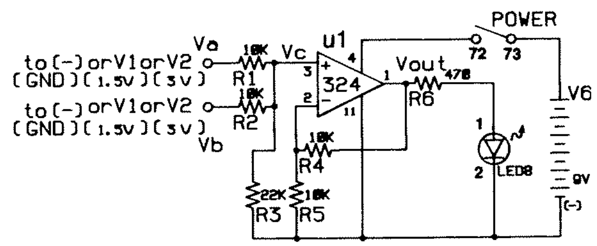
U1	324	R1	10KΩ	R4	10KΩ
		R2	10KΩ	R5	10KΩ
		R3	22KΩ	R6	470Ω

In this project, we're going to make an experimental adder using an operational amplifier. The circuit shown in the schematic is also called a "Summing Amplifier," because it sums up two or more voltages.

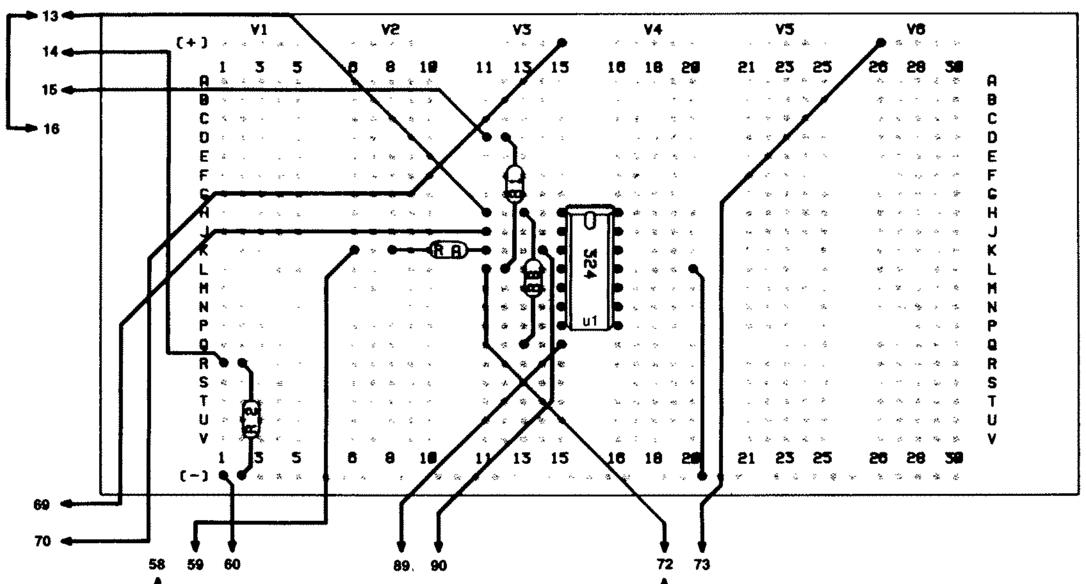
This circuit has two input terminals, Va and Vb. Input voltages applied to Va and Vb are summed up by operational amplifier IC 324 working as a non-inverting amplifier, and then displayed as an output.

Let's get to the experiment. Turn power ON, and connect the two input terminals to the (–) terminal. Since both input voltages are zero at this time, the output voltage is also zero and the **LED** doesn't light up.

See what happens when you leave Vb as it is and connect Va to **V1**. The **LED** lights up, but it's not very bright. Now connect Vb to **V1**. What happens this time? Observe how the **LED** changes its brightness when the voltage from the (–) terminal, **V1** and **V2** is applied. The way the **LED** changes its brightness lets you understand how the voltages are summed up by this circuit.



PROJECT 106. SCHMITT TRIGGER CIRCUIT



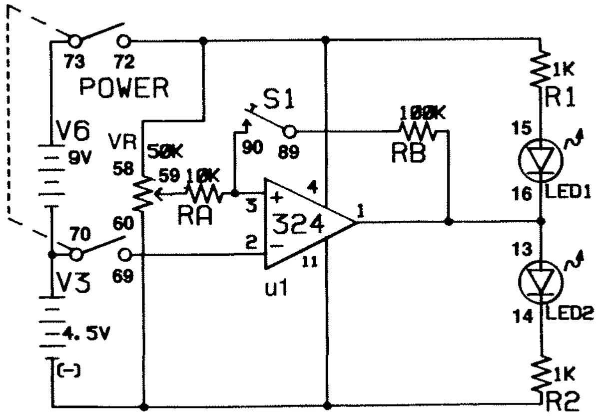
U1	324	RA	10KΩ	R1	1KΩ
		RB	100KΩ	R2	1KΩ

In this project, we're going to see how the operational amplifier works when used as a Schmitt trigger circuit and a comparator.

Figure shows that the Schmitt trigger circuit provides different hysteresis loops as you turn the **control volume** clockwise or counterclockwise.

Now, let get to the experiment. Turn power ON while leaving **S1** OFF, and the operational amplifier works as a comparator. When you rotate the **control volume**, **LEDs 1** and **2** take turns lighting at some point. Note that this point doesn't change whether you turn the **control volume** clockwise or counterclockwise. Now, press **S1** ON, and you'll have a Schmitt trigger circuit that produces hysteresis loops, as shown in Figure 1.

The width of hysteresis becomes narrower as the RB/RA ratio increases. Notice how the width changes when by using different value for RA and RB.



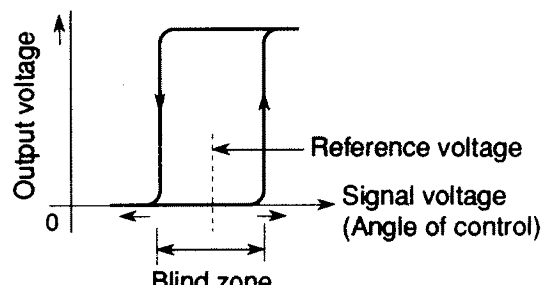
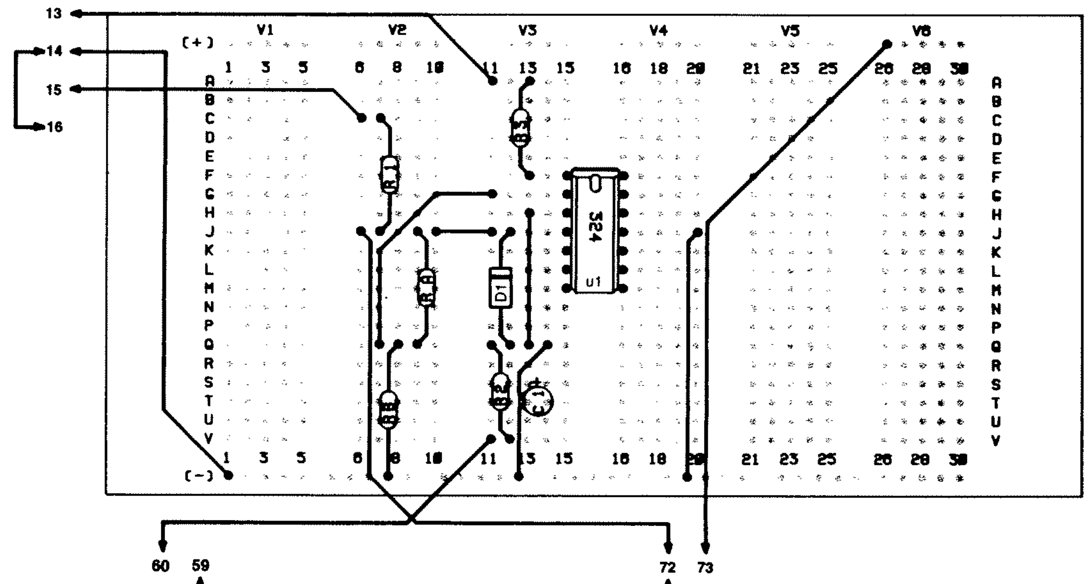


Figure 1

PROJECT 107. DELAYED TIMER



U1	324	R1	1KΩ	RA	1KΩ
D1	Si	R2	100KΩ	RB	1KΩ
		R3	1KΩ	C1	100µF

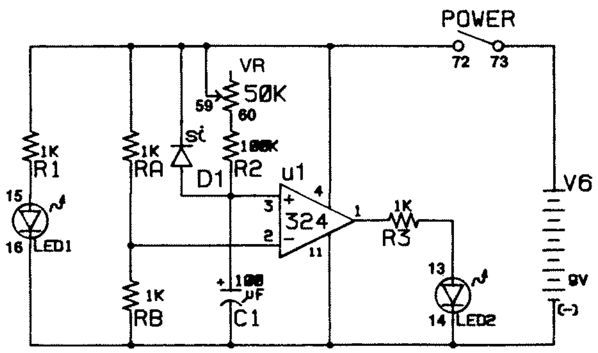
Back in project 102 we saw an operational amplifier working as a comparator. In this project, we're going to make an experiment on a delayed timer, using this operational amplifier and the CR time constant. The CR stands for, as you might suspect, capacitor and resistor.

The – terminal of the operational amplifier has a voltage of about 4.5 V, obtained by RA and RB. This is the comparator's reference voltage.

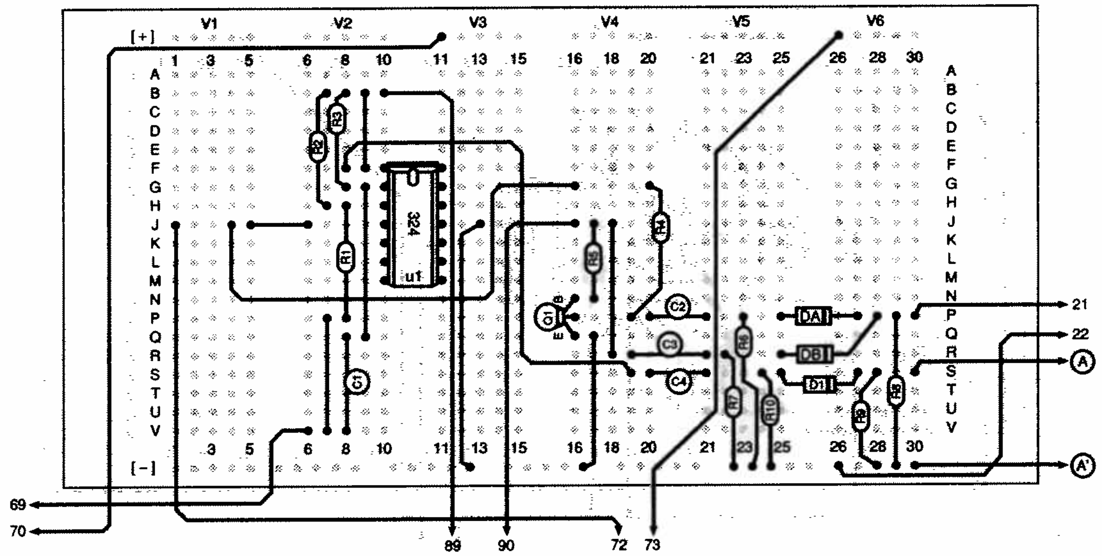
The + terminal of the comparator is connected to C1. C1 is charged by the series resistance of R2 and the **control volume**. The charging speed goes down if the resistance is large, and gets faster if the resistance is small. The delay time is set by this charging speed.

To use this project, set the **control volume** fully clockwise. Turn power ON: **LED 1** lights up first and **LED 2** also lights up about 7 - 8 seconds later. This 7- to 8-second time difference is the delay time set by the CR time constant.

Now, turn power OFF, set the **control volume** fully counterclockwise (minimum resistance position), and see what happens when you turn power ON again. **LED 2** again lights up later than **LED 1**, but how many seconds later? Now you see what this project's demonstrating.



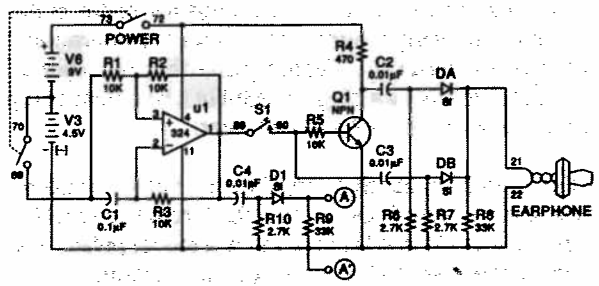
PROJECT 108. PULSE FREQUENCY DOUBLER



U1	324	R3	10KΩ	R7	2.7KΩ	C1	0.1µF	DA	Si
Q1	NPN	R4	470Ω	R8	33KΩ	C2	0.01µF	DB	Si
R1	10KΩ	R5	10KΩ	R9	33KΩ	C3	0.01µF	DC	Si
R2	10KΩ	R6	2.7KΩ	R10	2.7KΩ	C4	0.01µF		

This is a pulse frequency multiplier that works with just one transistor. It is called a "pulse frequency doubler" because it has a multiplication factor of 2.

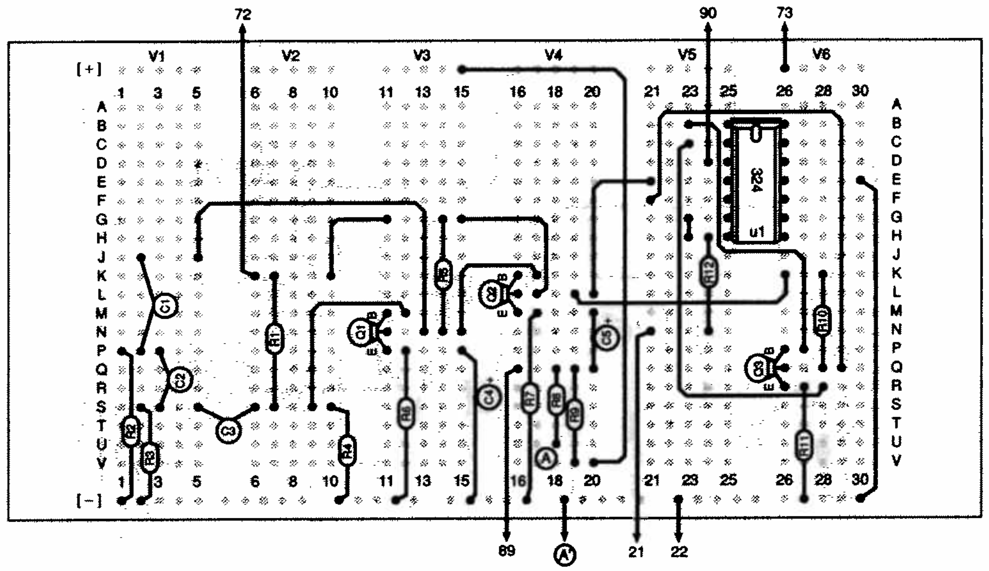
The operational amplifier IC 324 is a square-wave oscillator that generates the AC signals this time. Q1 is a frequency doubler, and its collector output signal is opposite in phase with respect to the base input signal. The two outputs from the base and collector are combined by Da and Db to double the frequency.



When you finish wiring the project, connect the earphone to A A', and turn power on. You'll hear a oscillating sound through the earphone. Remember the tone of that sound.

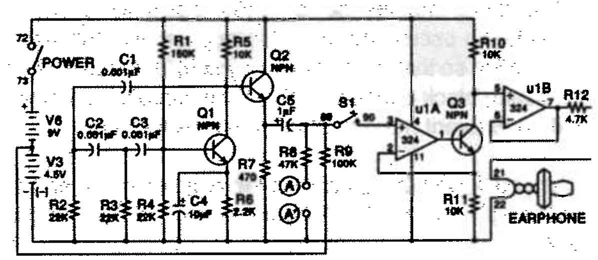
Now, connect the earphone to terminals **21** and **22**, and press **S1**. You will hear a sound higher by an octave, showing you that the frequency is doubled.

PROJECT 109. PITCH DOUBLING CIRCUIT



U1	324	R1	150KΩ	R5	10KΩ	R9	100KΩ	C1	0.001µF
Q1	NPN	R2	22KΩ	R6	2.2KΩ	R10	10KΩ	C2	0.001µF
Q2	NPN	R3	22KΩ	R7	470Ω	R11	10KΩ	C3	0.001µF
Q3	NPN	R4	22KΩ	R8	47KΩ	R12	4.7KΩ	C4	10µF
								C5	1µF

In this project, you're going to make a circuit that doubles a sound pitch by manipulating an OP amplifier. The result can be heard from the earphone. Precisely speaking, the circuit applies full-wave rectification to the sine wave input so that the output frequency is doubled, as shown in Figure 1. It is not that the frequency of the sine wave itself is doubled.



After you finish wiring, connect the earphone to A and A' and turn power ON. You'll hear a beep. Remember this tone.

Next, change the connections of the earphone to the terminals 21and 22, and then press **S1**. Again, you'll hear a beep, but the tone is different from the one you heard before. In fact, the beep now is one octave higher. In other words, the frequency, or pitch, has doubled.

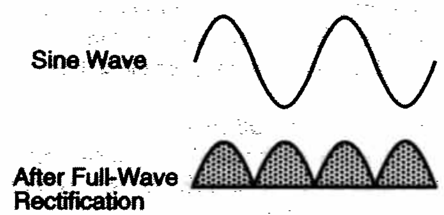
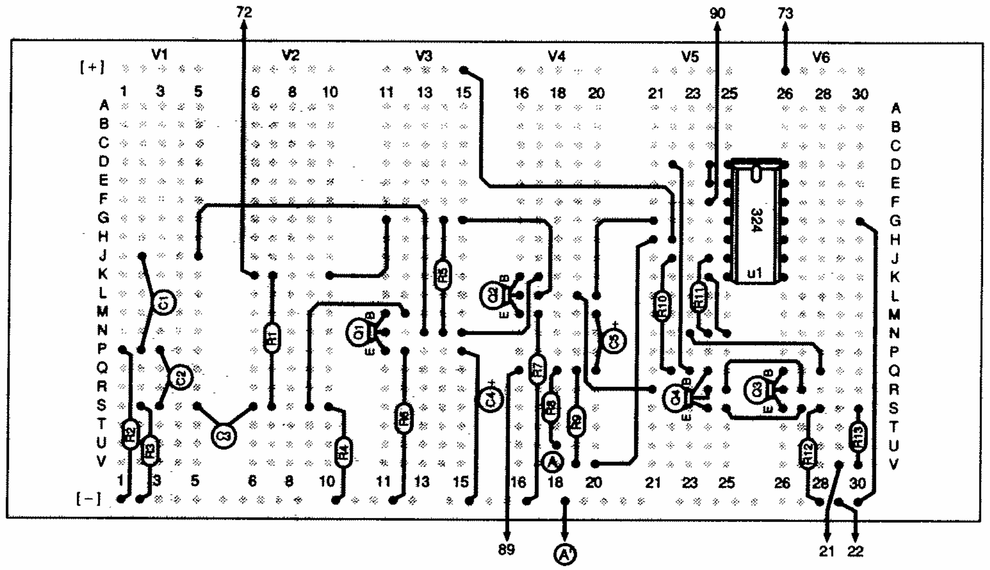


Figure 1

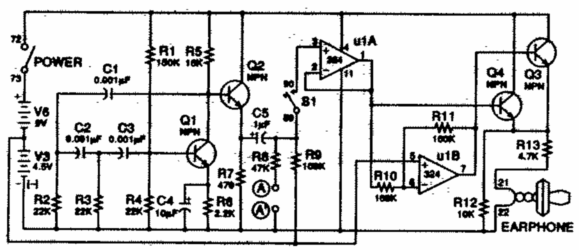
In this circuit, Q1 and Q2 make up an oscillator. The sine wave output from the oscillator passes through point A to the OP amplifier U1A. The circuit block consisting of U1A, Q3, R10, and R11 provides full-wave rectification. The voltage follower U1B has high input impedance (AC resistance) to prevent the Q3 output from being affected by the next stage.

PROJECT 110. PITCH DOUBLING CIRCUIT II



U1	324	R1	150KΩ	R6	2.2KΩ	R10	100KΩ	C1	0.001μF
Q1	NPN	R2	22KΩ	R7	470Ω	R11	100KΩ	C2	0.001μF
Q2	NPN	R3	22KΩ	R8	47KΩ	R12	10KΩ	C3	0.001μF
Q3	NPN	R4	22KΩ	R9	100KΩ	R13	4.7KΩ	C4	10μF
Q4	NPN	R5	10KΩ					C5	1μF

This project produces the same result as in the previous project. An oscillating output is applied with full-wave rectification so that the frequency (pitch) is doubled. The oscillating block consists of the same circuit, but it uses quite a different method of full-wave rectification.



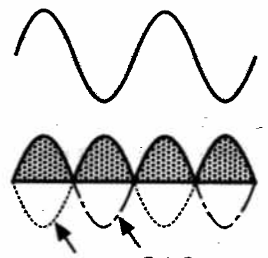
After you finish wiring, connect the earphone to A and A' and turn power ON. You'll hear a beep. Remember this tone.

Next, change the connections of the earphone to the terminals 21and 22, and then press **S1**. Again, you'll hear a beep, one octave higher than the one you heard before. That is, the frequency, or pitch, has doubled.

In this circuit, Q1 and Q2 make up an oscillator. The sine wave output from the oscillator passes through point A, voltage follower U1A, to U1B. Do you know how full-wave rectification occurs at U1B, Q3, and Q4. A hint for you--the input signal and output signal of the OP amplifier U1B are opposite in phase, and they are added together (ORed) as they pass through the emitter followers Q4 and Q3. You see, when the output of Q3 is high, Q4 is OFF, and when the output of Q4 is high, Q3 is OFF. Therefore, only the alternations of high level are kept output as shown in Figure 1.

Sine Wave

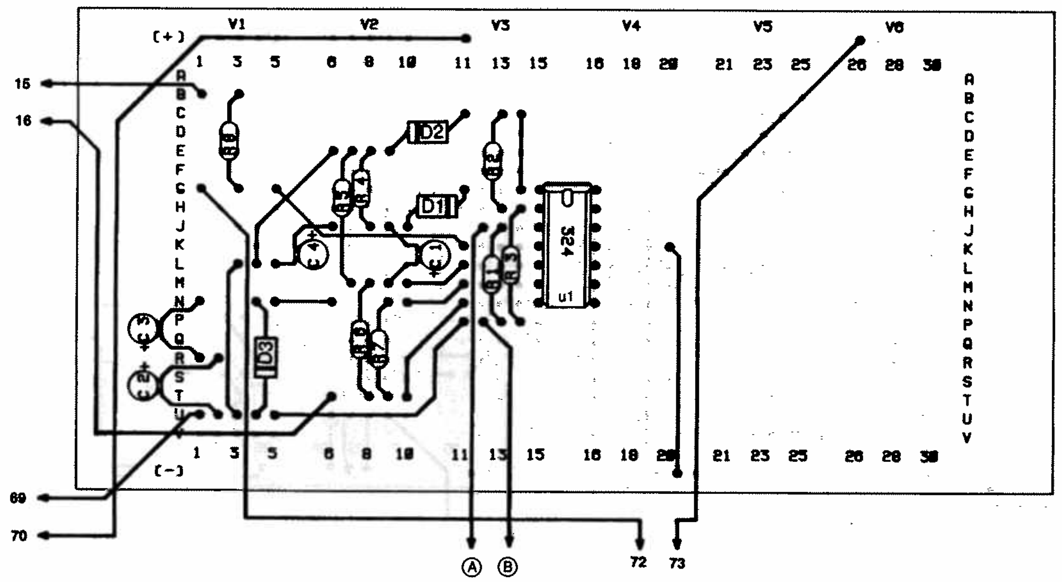
After Full-Wave Rectification



Q3 Output Q4 Output

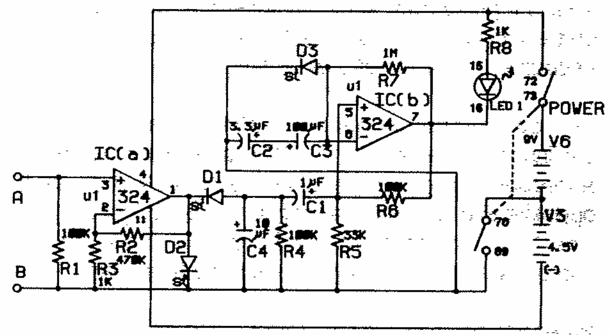
Figure 1

PROJECT 111. TOUCH SWITCH USING OP AMPLIFIER



U1	324	R1	100KΩ	R5	33KΩ	C1	1μF
D1	Si	R2	470KΩ	R6	100KΩ	C2	3.3μF
D2	Si	R3	1KΩ	R7	1MΩ	C3	100μF
D3	Si	R4	100KΩ	R8	1KΩ	C4	10μF

Here's a touch switch: you can change the status of the circuit by touching the terminal. Here the operational amplifier acts as an amplifier and a one-shot multivibrator. You can turn ON by a single touch but it is automatically turned OFF about 1 - 2 seconds later.



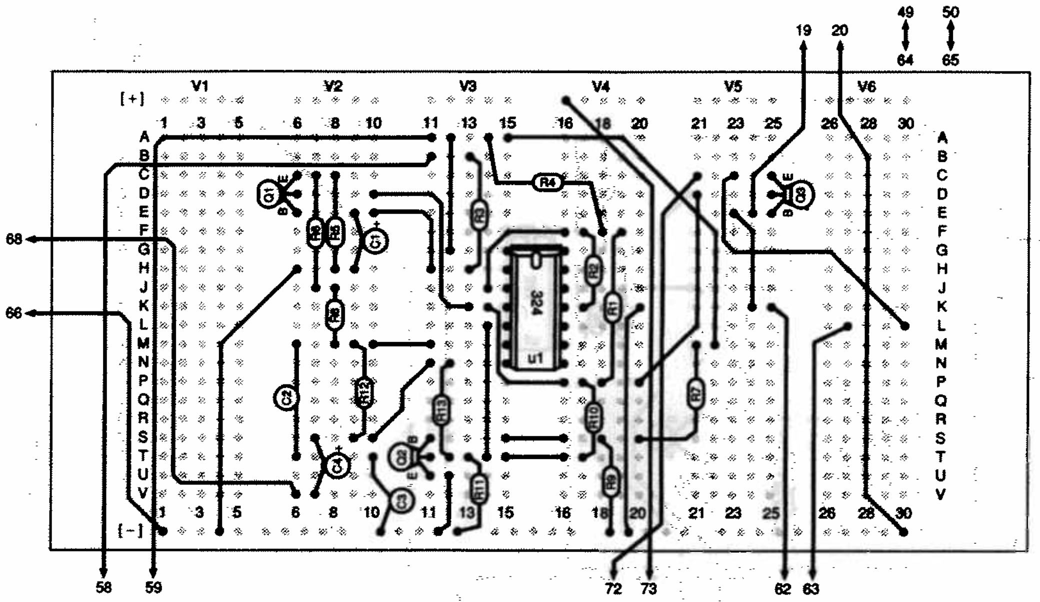
IC(a) is an amplifier that amplifies the voltage induced by a touch of your finger. The output from this amplifier is rectified by D1 and D2 to produce a trigger pulse. The pulse is then sent to the one-shot multivibrator.

IC(b), as the one-shot multivibrator, works as an electronic switch. When it receives an input, the output stays at low level for 1 - 2 seconds, then lights up the **LED** again.

Turn power ON and touch the wire "A".

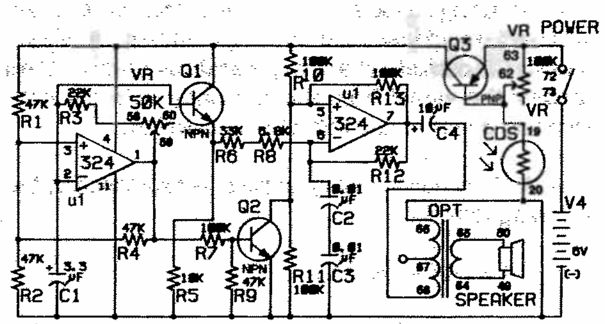
If this touch switch doesn't work as explained above, your finger tip might be too dry. See if it works well when you touch it after wetting your finger tip.

PROJECT 112. EARLY BIRD



U1	324	R2	47KΩ	R7	100KΩ	R12	22KΩ	C1	3.3µF
Q1	NPN	R3	22KΩ	R8	6.8KΩ	R13	100KΩ	C2	0.01µF
Q2	NPN	R4	47KΩ	R9	47KΩ			C3	0.01µF
Q3	PNP	R5	10KΩ	R10	100KΩ			C4	10µF
R1	47KΩ	R6	33KΩ	R11	100KΩ				

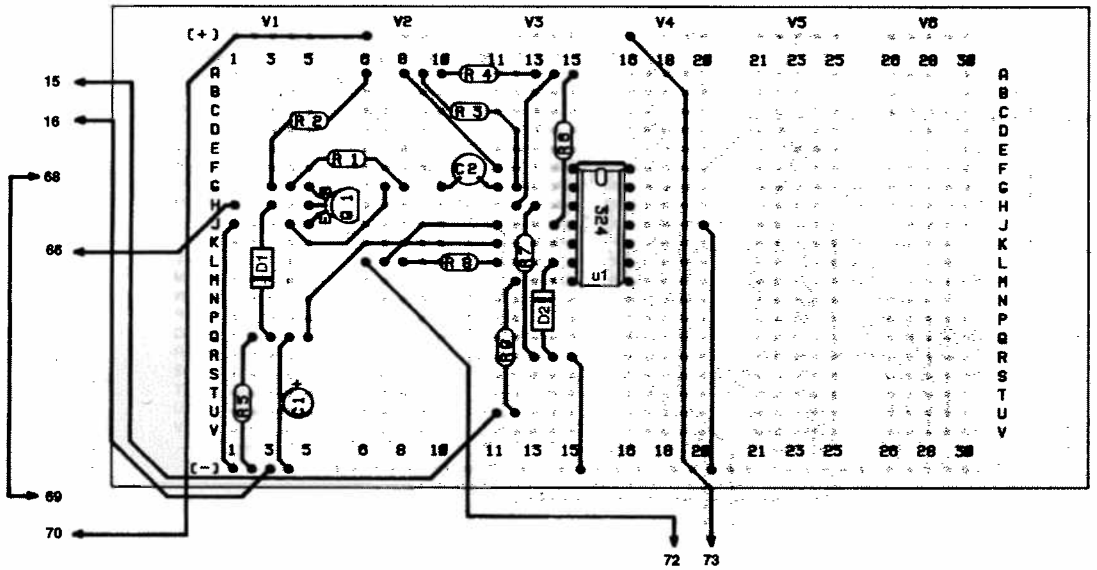
Let's make an early bird that begins chirping when it grows light in the morning. No doubt you suspect we use **CdS cell** :yes you're right. In this project the op amplifier IC is used as the square-wave oscillator and the sweep oscillator. When the resistance of the **CdS cell** goes down, the switching transistor Q3 turns ON, and power is supplied to the circuit.



When you finish wiring up the circuit, turn the 100K **control volume** tuner fully clockwise. Set the 50K **control volume** in the 12 o'clock position, and turn power ON. You'll hear the early bird chirping from the **speaker**. Rotate the 50K **control volume**, and you can change the tone of the chirp. Set the 50K **control volume** in the position where you can hear the twitter you like the best.

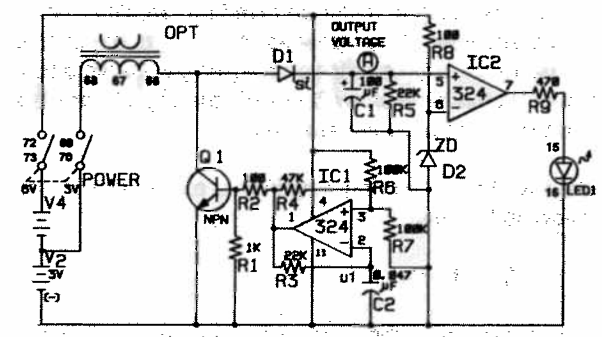
The 100K **control volume** adjusts how much brightness is required to wake up the early bird. Adjust to the best point so you can wake at the time not too late to school!

PROJECT 113. DC-DC CONVERTER BY OP AMPLIFIER



U1	324	Q1	NPN	R3	22KΩ	R6	100KΩ	R9	470Ω
D1	Si	R1	1KΩ	R4	47KΩ	R7	100KΩ	C1	100µF
D2	Zd	R2	100Ω	R5	22KΩ	R8	100Ω	C2	0.047µF

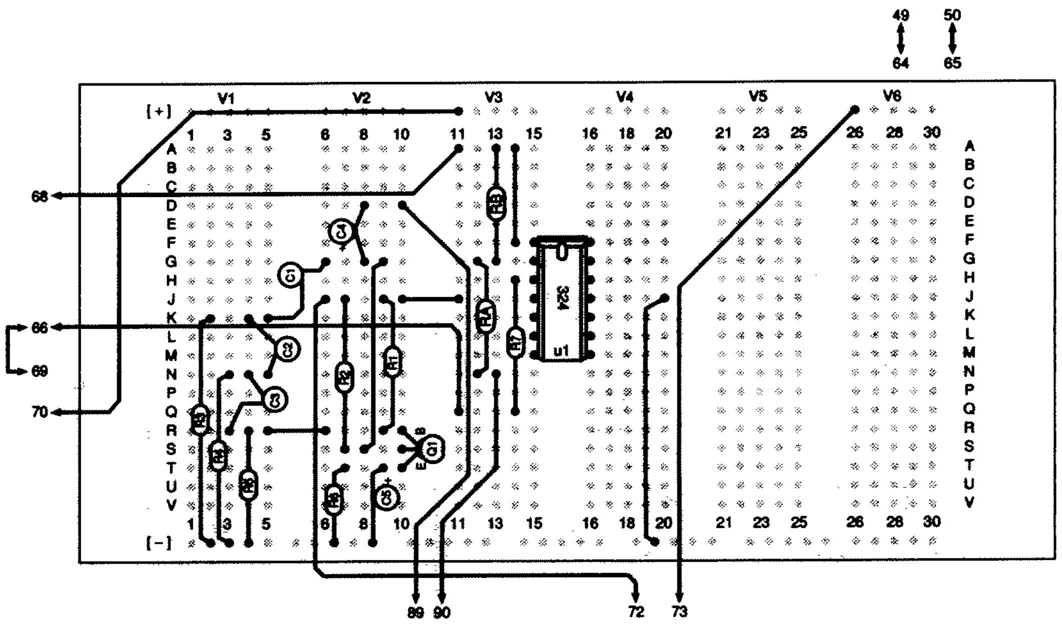
Here's another DC-DC converter, but this one is intended specifically to obtain 5 V DC from 3 V DC.



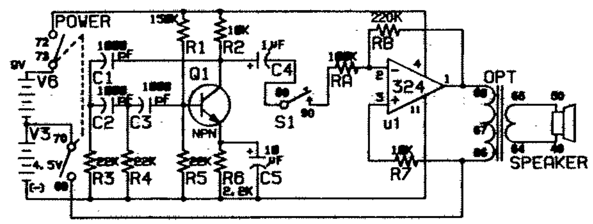
Look at the schematic for this project. IC1 is an oscillator. The output is used to turn Q1 ON. A high voltage is generated instantly by the self-induction of the coil. This voltage is rectified by D1 to obtain a high DC voltage. IC2 is a comparator for examining the voltage increase. When its input voltage goes up to more than 5 V, the **LED** lights up.

When you wire the project, turn power ON and see how this circuit works. Note that the voltage applied to the coil is 3 V. But when the **LED** lights up, you can see that you've obtained a voltage of more than 5V.

PROJECT 114. INVERTING AMPLIFIER



U1	324	R3	22KΩ	R7	10KΩ	C1	1000pF	C4	1µF
Q1	NPN	R4	22KΩ	RA	100KΩ	C2	1000pF	C5	10µF
R1	150KΩ	R5	22KΩ	RB	220KΩ	C3	1000pF		
R2	10KΩ	R6	2.2KΩ						



We're now going to make an inverting amplifier using the op amplifier. The signal source is a phase-shifting CR oscillator using a transistor, which generates a sine wave. In this project, we'll amplify the signal from this oscillator to activate the **speaker**.

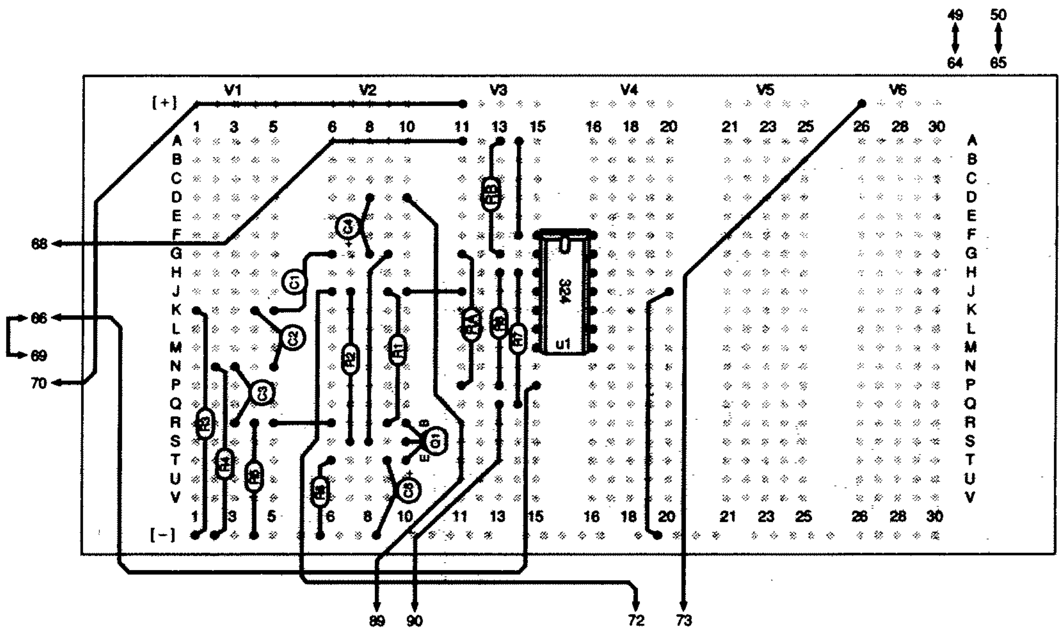
Gain A of the inverting amplifier is expressed by the following formula:

A = RB/RA

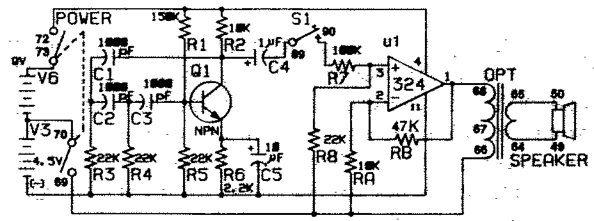
This means that the gain becomes greater if RB is increased or RA is decreased.

When you finish wiring up the equipment, turn power ON and see what happens when you press **S1**. You can hear a clear, high-pitched sound from the **speaker**. You can practice sending Morse code by pressing **S1** intermittently.

PROJECT 115. NON-INVERTING AMPLIFIER



U1	324	R3	22KΩ	R7	100KΩ	C1	1000pF	C5	10µF
Q1	NPN	R4	22KΩ	R8	22KΩ	C2	1000pF		
R1	150KΩ	R5	22KΩ	RA	10KΩ	C3	1000pF		
R2	10KΩ	R6	2.2KΩ	RB	47KΩ	C4	1µF		



In this project, we're going to make a non-inverting amplifier using the op amplifier. The signal source is the same as we've just used in our last project, a phase-shifting CR oscillator that can generate sine wave. We'll amplify the signal from this oscillator with the non-inverting amplifier and activate the **speaker**.

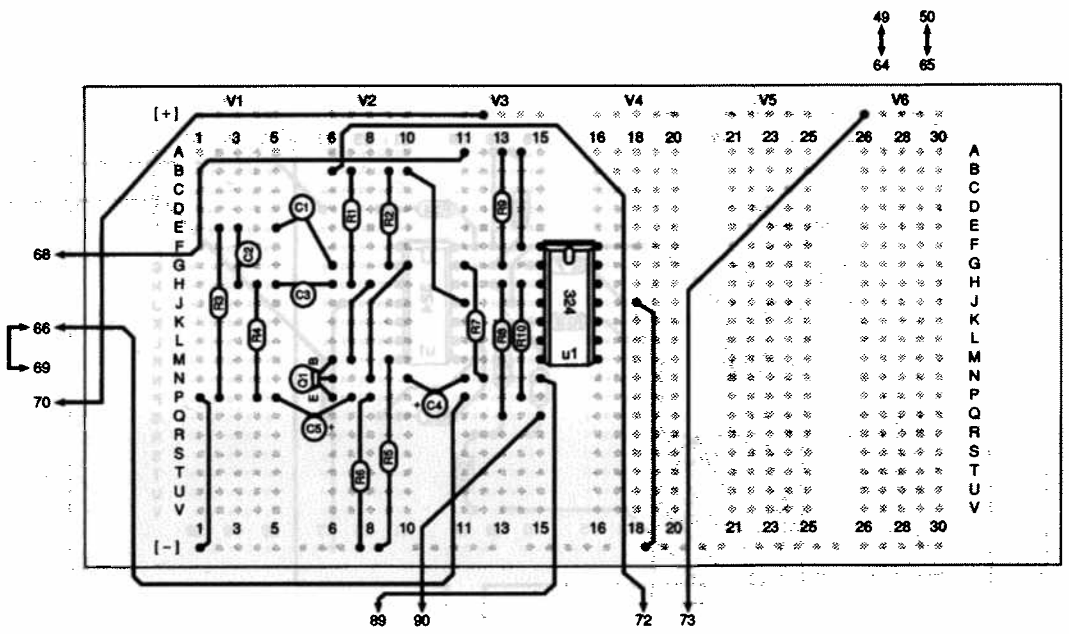
Gain A of the non-inverting amplifier is expressed by the following formula:

A = (RA + RB) /RA

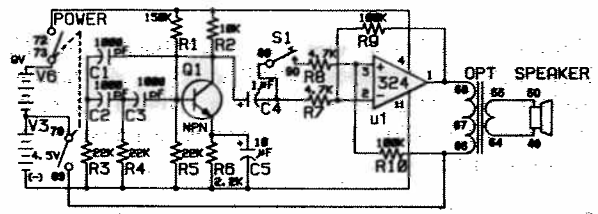
This means that the gain becomes greater if RB is increased or RA is decreased.

When you've assembled the equipment, turn power ON and press **S1**, and you'll hear a clear, high-pitched sound coming continuously from the **speaker**. Again you can send Morse code by pressing **S1** intermittently.

PROJECT 116. DIFFERENTIAL AMPLIFIER



U1	324	R3	22KΩ	R7	4.7KΩ	C1	1000pF	C5	10μF
Q1	NPN	R4	22KΩ	R8	4.7KΩ	C2	1000pF		
R1	150KΩ	R5	22KΩ	R9	100KΩ	C3	1000pF		
R2	10KΩ	R6	2.2KΩ	R10	100KΩ	C4	1μF		

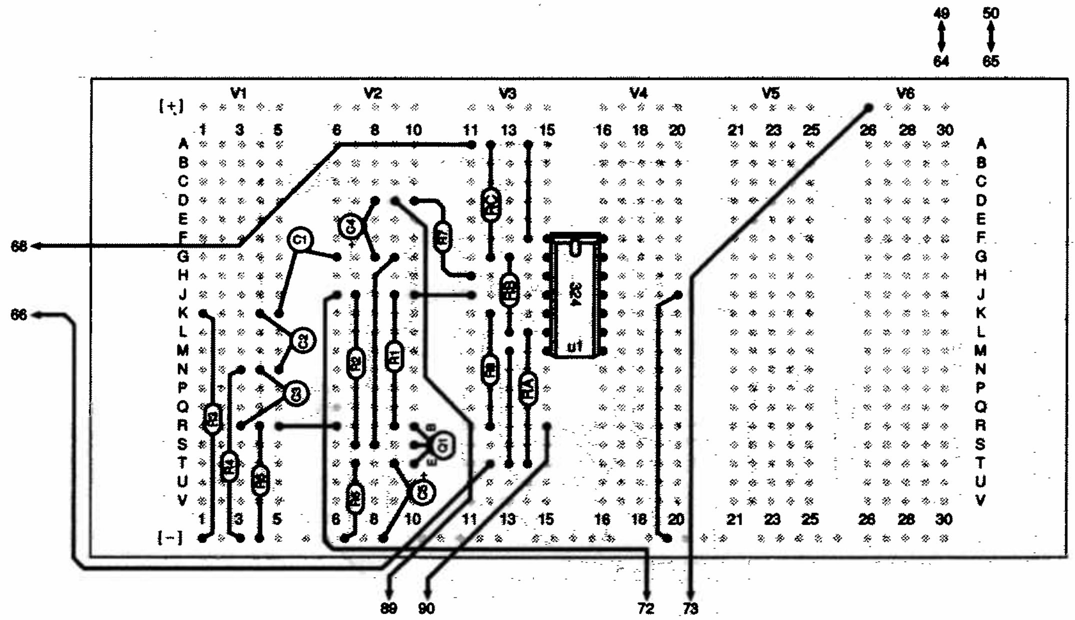


Now, we're going to construct a differential amplifier using the op amplifier. In this experiment, we'll provide input to both the + and – terminals of the op amplifier. The differential amplifier amplifies the difference between these two inputs. If the two inputs are the same, it produces no output at all.

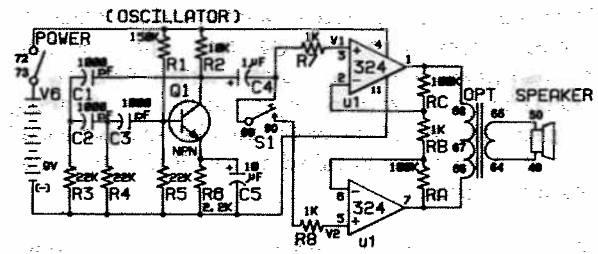
Let's get to the experiment. Use the signal output from the phase-shifting CR oscillator. Turn power ON, and you'll hear a high-pitched sound produced by the **speaker**. Can you guess why? Yes, it's because **S1** is still OFF at this time and the op amplifier is working as an inverting amplifier.

Now, see what happens by pressing **S1**. You'll notice the sound from the **speaker** is toned down or becomes almost inaudible. This is the function of the differential amplifier. Using this function, we can suppress noise and amplify only the signal we want to listen.

PROJECT 117. DIFFERENTIAL OUTPUT AMPLIFIER



U1	324	R3	22KΩ	R7	1KΩ	RC	100KΩ	C4	1μF
Q1	NPN	R4	22KΩ	R8	1KΩ	C1	1000pF	C5	10μF
R1	150KΩ	R5	22KΩ	RA	100KΩ	C2	1000pF		
R2	10KΩ	R6	2.2KΩ	RB	1KΩ	C3	1000pF		

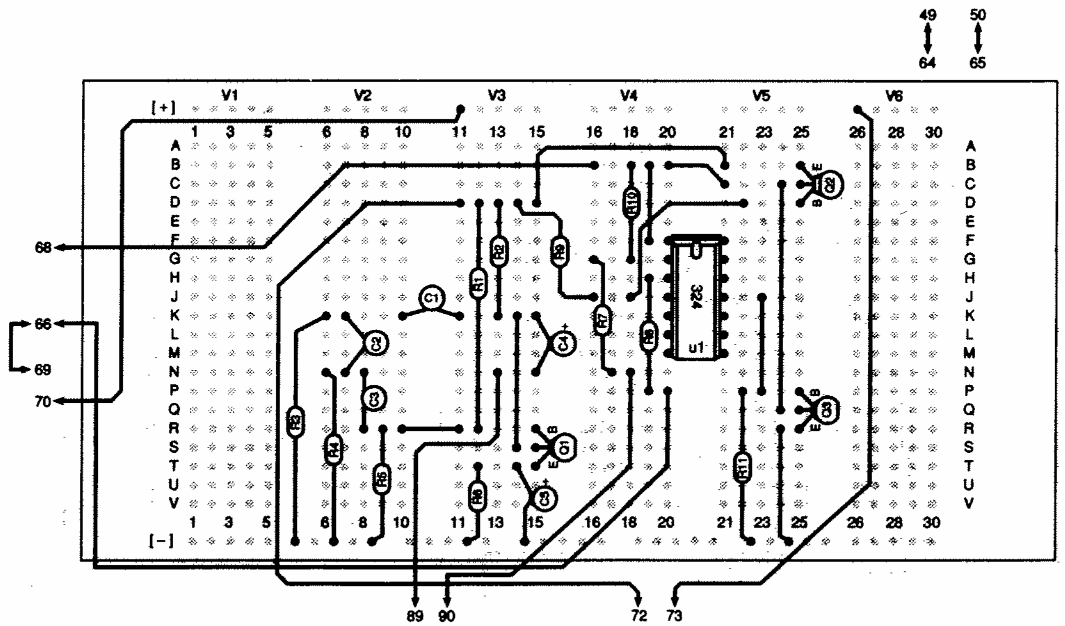


This project lets you see the function of a differential output amplifier using two op amplifiers. The differential output amplifier produces an output when there's difference for the output side. The audio signal is generated from a phase-shifting CR oscillator (again).

When you finish wiring, turn power ON. You'll hear a sound from the **speaker**. This is because the op amplifier is not working as a differential output amplifier yet.

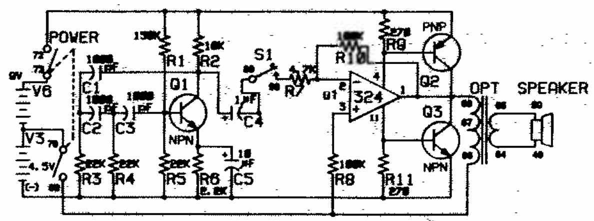
Now, press **S1**. The sound from the **speaker** disappears. This is the function of the differential output amplifier. The gain of this amplifier can be controlled by varying the value of RB. It becomes greater if RB is reduced.

PROJECT 118. POWER AMPLIFIER USING OP AMPLIFIER



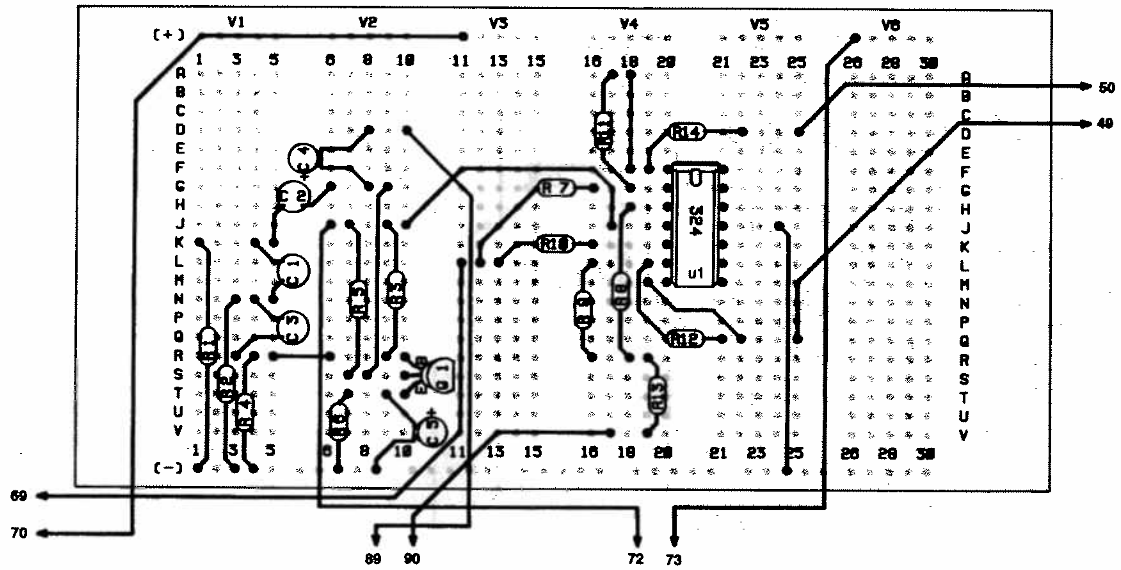
U1	324	R1	150KΩ	R5	22KΩ	R9	270Ω	C2	1000pF
Q1	NPN	R2	10KΩ	R6	2.2KΩ	R10	100KΩ	C3	1000pF
Q2	PNP	R3	22KΩ	R7	4.7KΩ	R11	270Ω	C4	1μF
Q3	NPN	R4	22KΩ	R8	100KΩ	C1	1000pF	C5	10μF

We're now going to produce a loud sound by combining the op amplifier and two transistors. The schematic shows you that the signal source is a phase-shifting CR oscillator. The op amplifier is used as an inverting amplifier, Q2 and Q3 cause the **speaker** to make a loud sound. This circuit is called an SEPP (Single Ended Push-Pull) circuit.



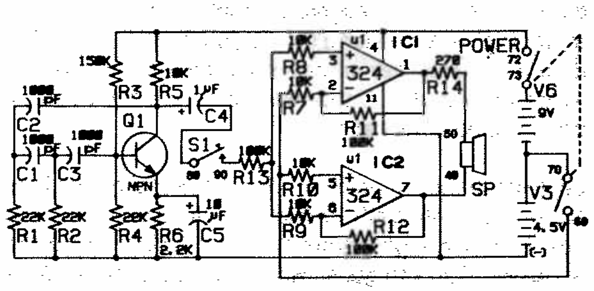
Turn power ON and press **S1**. You'll hear loud sound from the **speaker**.

PROJECT 119. BALANCED TRANSFORMERLESS AMPLIFIER



U1	324	R3	150KΩ	R7	10KΩ	R11	100KΩ	C1	1000pF
Q1	NPN	R4	22KΩ	R8	10KΩ	R12	100KΩ	C2	1000pF
R1	22KΩ	R5	10KΩ	R9	10KΩ	R13	100KΩ	C3	1000pF
R2	22KΩ	R6	2.2KΩ	R10	10KΩ	R14	270Ω	C4	1μF
								C5	10μF

We are now going to make an experiment of a BTL (balanced transformerless amplifier, not bacon, tomato and lettuce), using two op amplifiers. The BTL is made up of two SEPPs linked with a pushpull. It increases the speaker output by four times.

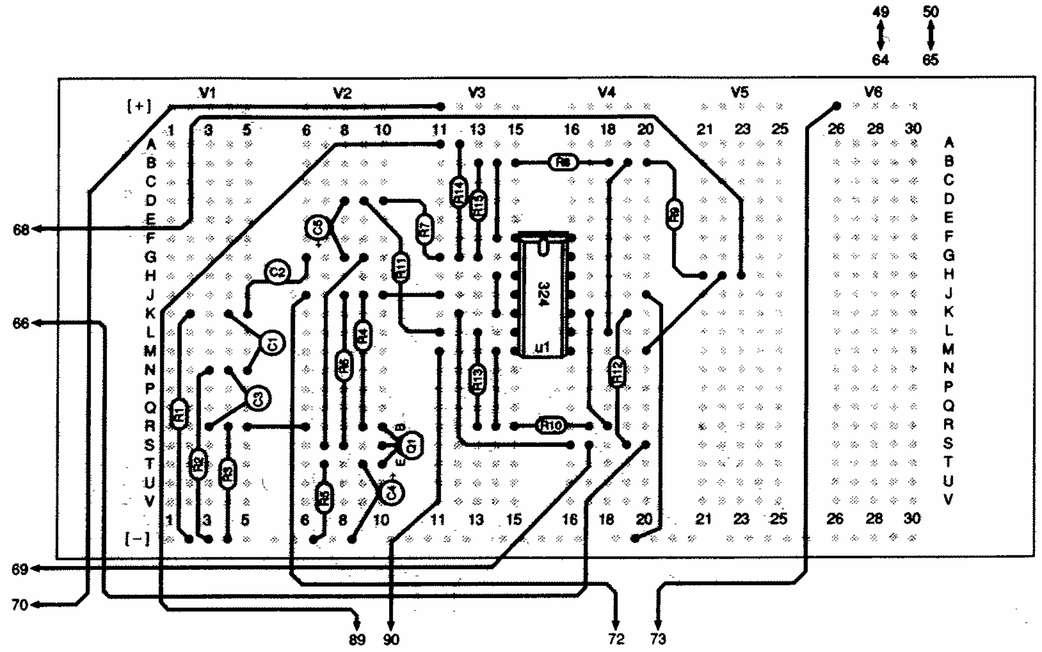


The signal source is a phase-shifting CR oscillator. The BTL uses two ICs, IC1 serving as a non-inverting amplifier and IC2 as an inverting amplifier.

When you finish wiring the project, turn power ON, press **S1** and see (listen?) what happens. You'll hear a high-pitched beep from the **speaker**.

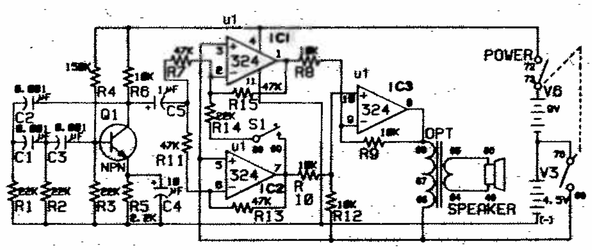
You must have noticed you can't make a loud sound in this experiment, and that's because the power of the two ICs is small. BTLs actually used in various electronic devices have a transistor operating as a "driver" to increase the output of the two ICs.

PROJECT 120. THREE-STAGE DIFFERENTIAL AMPLIFIER



U1	324	R4	150KΩ	R9	10KΩ	R14	22KΩ	C4	10μF
Q1	NPN	R5	2.2KΩ	R10	10KΩ	R15	47KΩ	C5	1μF
R1	22KΩ	R6	10KΩ	R11	47KΩ	C1	0.001μF		
R2	22KΩ	R7	47KΩ	R12	10KΩ	C2	0.001μF		
R3	22KΩ	R8	10KΩ	R13	47KΩ	C3	0.001μF		

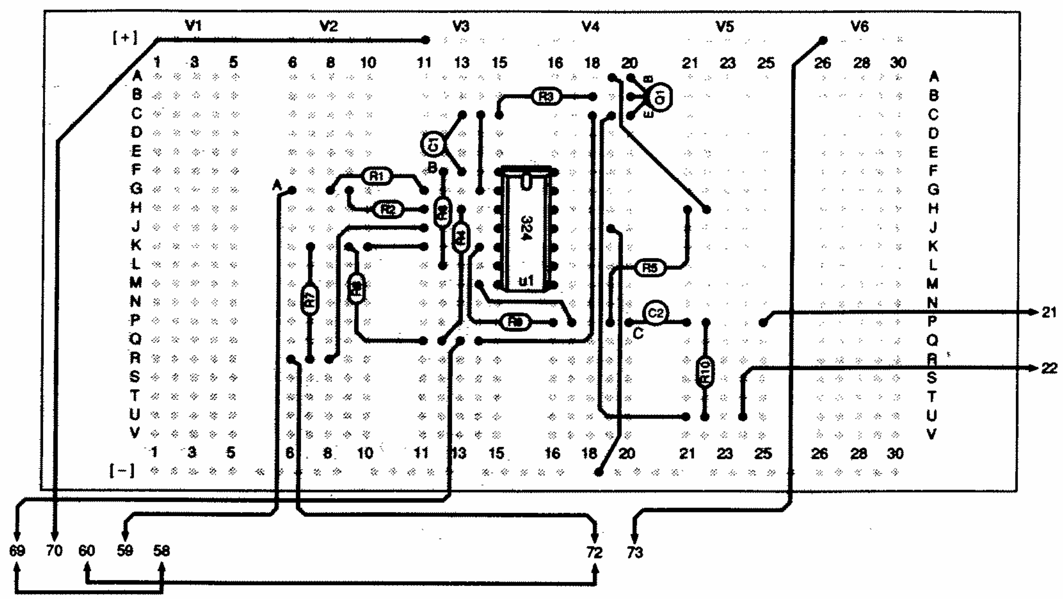
Back in project 116 we built a differential amplifier. We're again going to build a differential amplifier, but this time the amplifier uses three op amplifiers to achieve the same.



The schematic shows you that Q1 is an oscillator that is already familiar to us. It gives out a highpitched "beep." IC1 and IC2 generate in-phase signal when **S1** is off, opposite phase signals when **S1** is on. IC3 works as a differential amplifier.

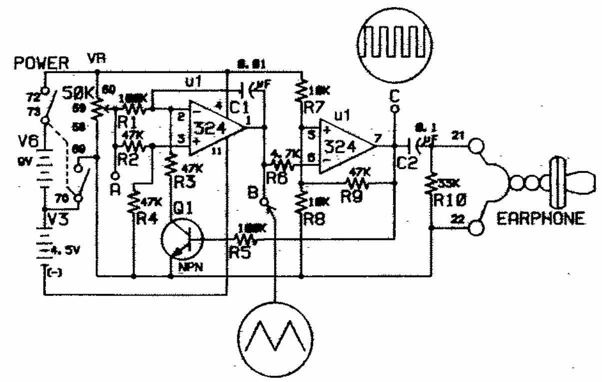
When you finish wiring up the circuit, switch power ON, and try to listen to the sound with **S1** released. Do you hear a faint beep and a noise? That's because in-phase input signals offset each other. Now press **S1** and see what happens. You can hear a louder beep. That's because the opposite phase signals are applied to the differential amplifier, activating the **speaker**.

PROJECT 121. VCO USING OP AMPLIFIER



U1	324	R3	47KΩ	R7	10KΩ	C1	0.01μF
Q1	NPN	R4	47KΩ	R8	10KΩ	C2	0.1μF
R1	100KΩ	R5	100KΩ	R9	47KΩ		
R2	47KΩ	R6	4.7KΩ	R10	33KΩ		

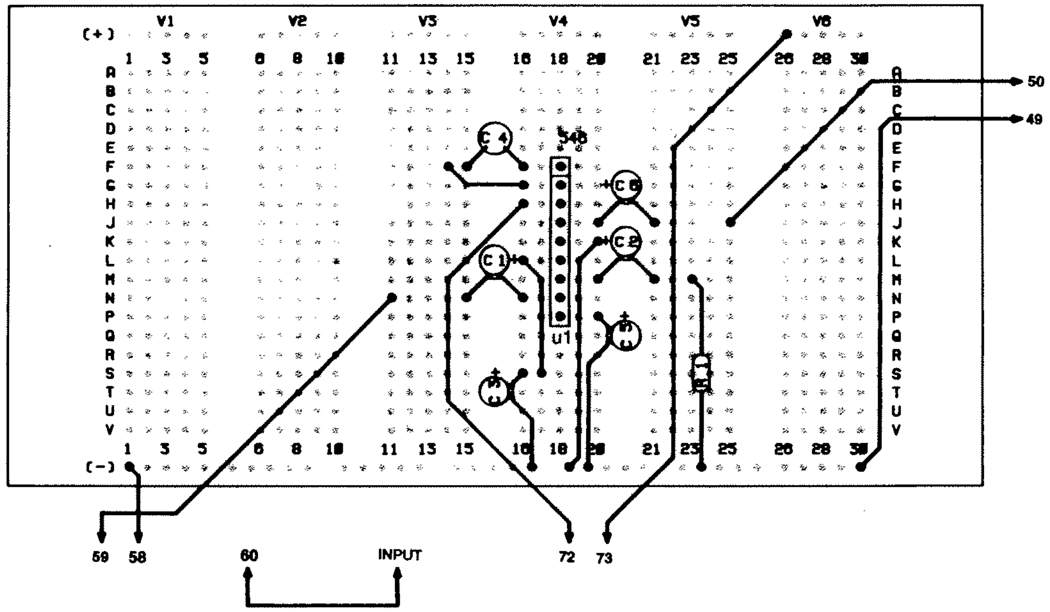
In this project, we're going to make a VCO using the operational amplifier. You know what the VCO is. If not, go back to project 96. This circuit produces two output signals, triangular and square waves.



The project's circuit lets you see that when the voltage of terminal A is changed, it causes a change in the charging discharging time and frequency of the CR integrating circuit. A triangular wave output is obtained from terminal B, and sent to the comparator. Terminal C produces a squarewave output.

When you finish wiring up the project, switch power ON, and turn the **control volume** slowly while listening to the sound from the earphone. When you turn the **control volume** clockwise, the sound becomes higher in pitch. That is the function of the VCO. The circuit shown in the schematic is used in music synthesizers to create sounds of different pitches.

PROJECT 122. IC POWER AMPLIFIER

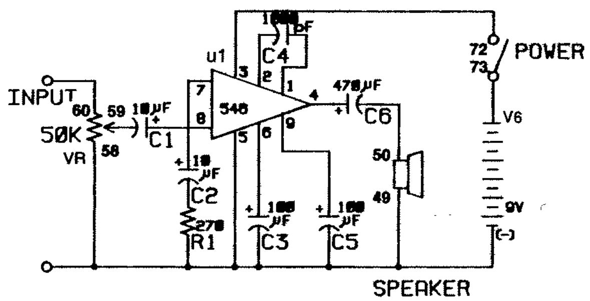


U1	546	C1	10µF	C4	1000pF
R1	270Ω	C2	10µF	C5	100µF
		C3	100µF	C6	470µF

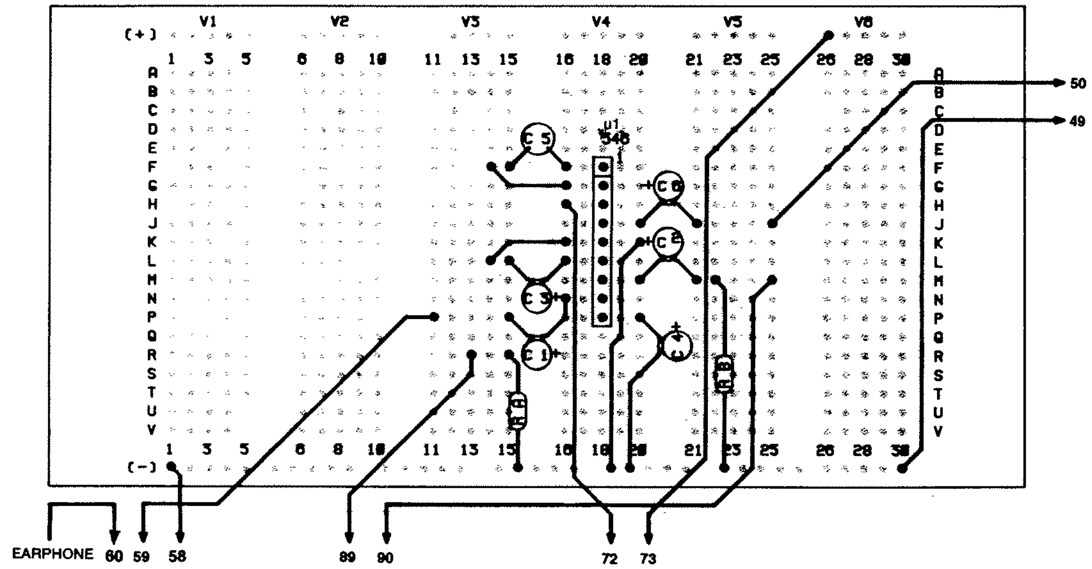
Let's make an audio power amplifier that can activate the **speaker**, using an audio power amplifier IC.

The IC used in the project is one that has been developed for use in portable radios, radio-cassette tape recorders, intercoms, etc. It can produce an output of about 700 mW. Using this IC, you can change the amplification rate according to the value of R1. The amplification rate Av is about 2400/R (value of the resistor). So, it is about 9 when R1 = 270 ohms.

To use this project, connect a germanium radio, earphone-type radio, radio-cassette tape recorder, etc. to the input terminal, then turn the **control volume** to adjust the sound volume from the **speaker**.



PROJECT 123. IC POWER AMPLIFIER II

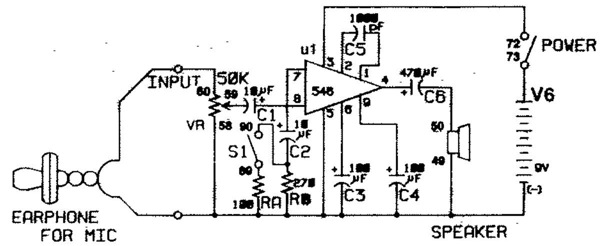


U1	546	C1	10µF	C4	100µF
RA	100Ω	C2	10µF	C5	1000pF
RB	270Ω	C3	100µF	C6	470µF

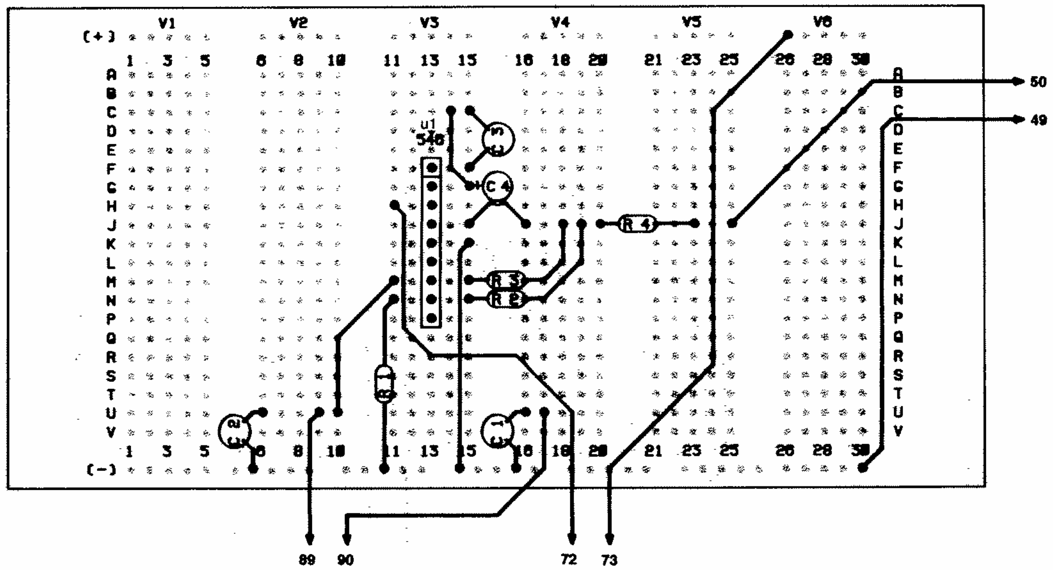
In this project, we're going to make an audio amplifier using the power amplifier IC. As said in the last project, you can change the gain of this IC by turning **S1** ON and OFF: it is about 9 when **S1** is turned OFF, and increases to about 33 when **S1** is turned ON.

Now, let's get to the experiment using the earphone as a microphone. Turn power ON, and speak into the earphone while turning the **control volume**. When you turn the **control volume** completely clockwise, you'll notice your voice becomes louder, but not very loud.

See what happens when you turn **S1** ON. Your voice becomes much louder because of the increased gain. Adjust the **control volume** to a suitable level and practice speaking over the microphone.



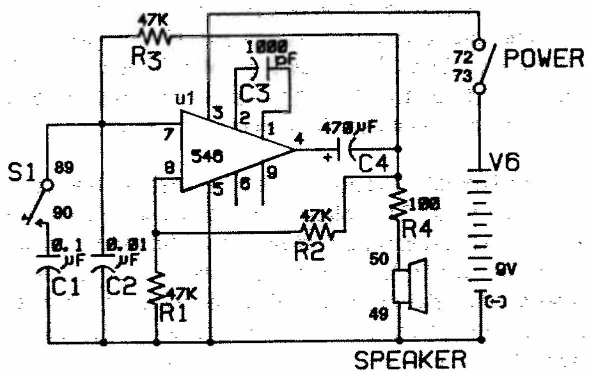
PROJECT 124. OSCILLATOR USING POWER AMPLIFIER IC



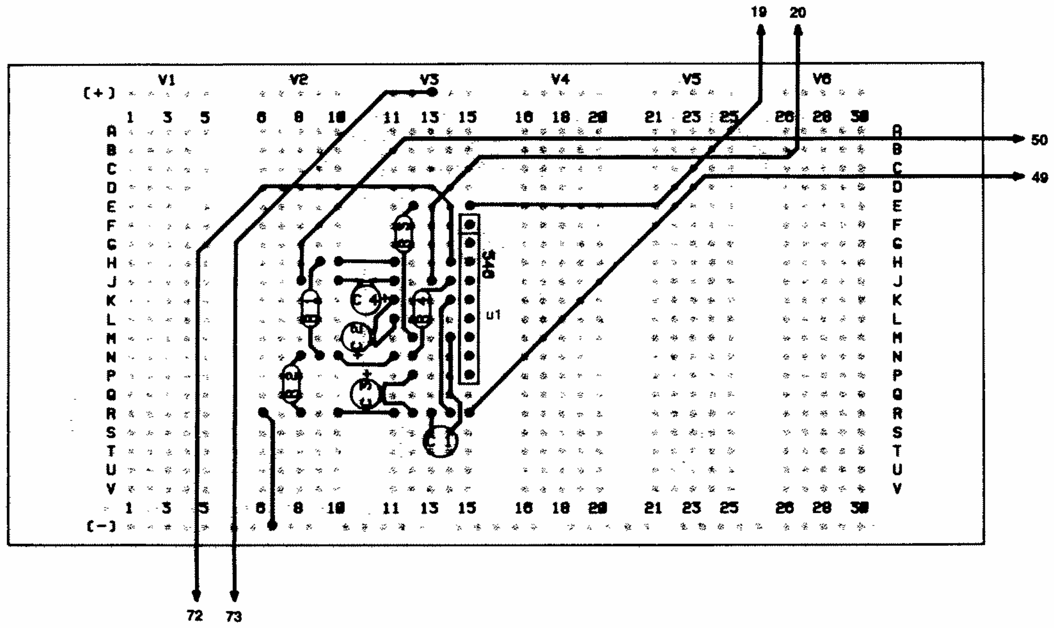
U1	546	R1	47KΩ	R3	47KΩ	C1	0.1µF	C3	1000pF
		R2	47KΩ	R4	100Ω	C2	0.01µF	C4	470µF

Here's another audio oscillator using the audio power amplifier IC, but it's a bit different because it can produce two kinds of sound, high- and low-pitched, by changing the position of the key. Since we use the power amplifier IC to build this project, this oscillator can produce a very loud sound. The frequency is determined by R1 and R2 when **S1** is OFF, and by R1, C1 and C2 when **S1** is ON.

When you finish wiring up the project, turn power ON and see how it makes two kinds of sound. You'll see that it makes a high-pitched sound when **S1** is OFF. The frequency at this time is about 5 kHz. Turn **S1** ON and the pitch goes down. That's because the frequency goes as low as about 500 Hz when **S1** is turned ON.



PROJECT 125. CdS CONTROLLED IC OSCILLATOR

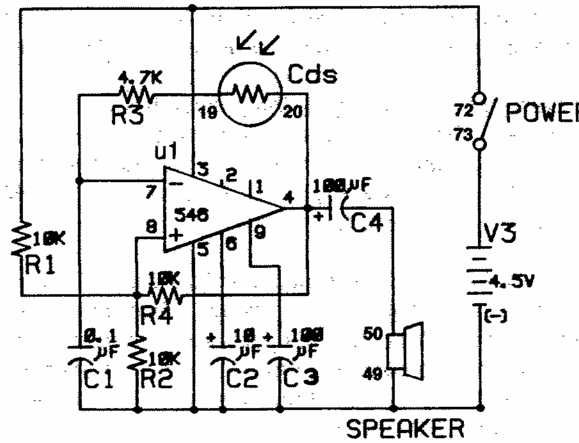


U1	546	R1	10KΩ	R3	4.7KΩ	C1	0.1µF	C3	100µF
		R2	10KΩ	R4	10KΩ	C2	10µF	C4	100µF

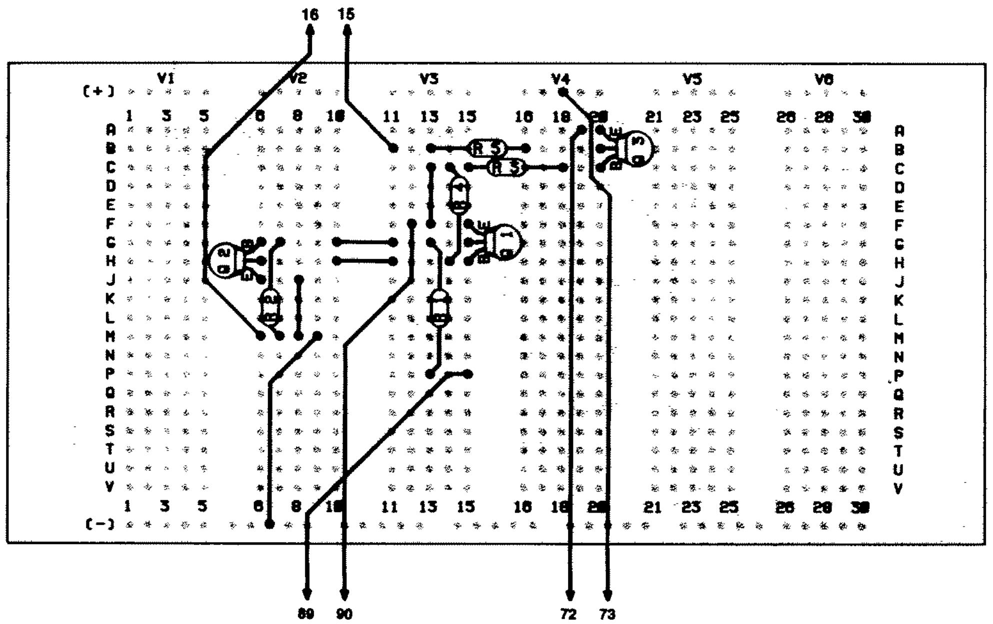
This project is a circuit with the IC as an oscillator. The tone of this oscillator varies with the amount of light hitting the **CdS cell**. Try changing the tone by moving your hand over the **CdS cell**.

The **CdS cell** acts as an open circuit in total darkness, but when some light strikes the **CdS cell** its resistance decreases enough to allow feedback of output signal to get through and sustain oscillations.

You might want to try using different values of resistors to see if there are any changes in oscillation. You can try every capacitor on the parts container without fear of damage. Have fun!



PROJECT 126. SWITCHING CIRCUIT



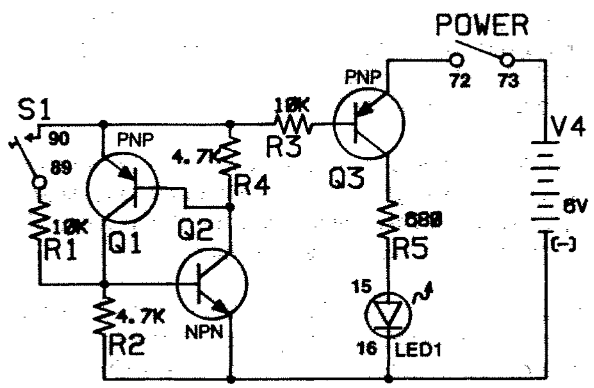
Q1	PNP	R1	10KΩ	R4	4.7KΩ
Q2	NPN	R2	4.7KΩ	R5	680Ω
Q3	PNP	R3	10KΩ		

Here's an electronic switch that allows the current to keep flowing once it is turned on, even if it is turned OFF later.

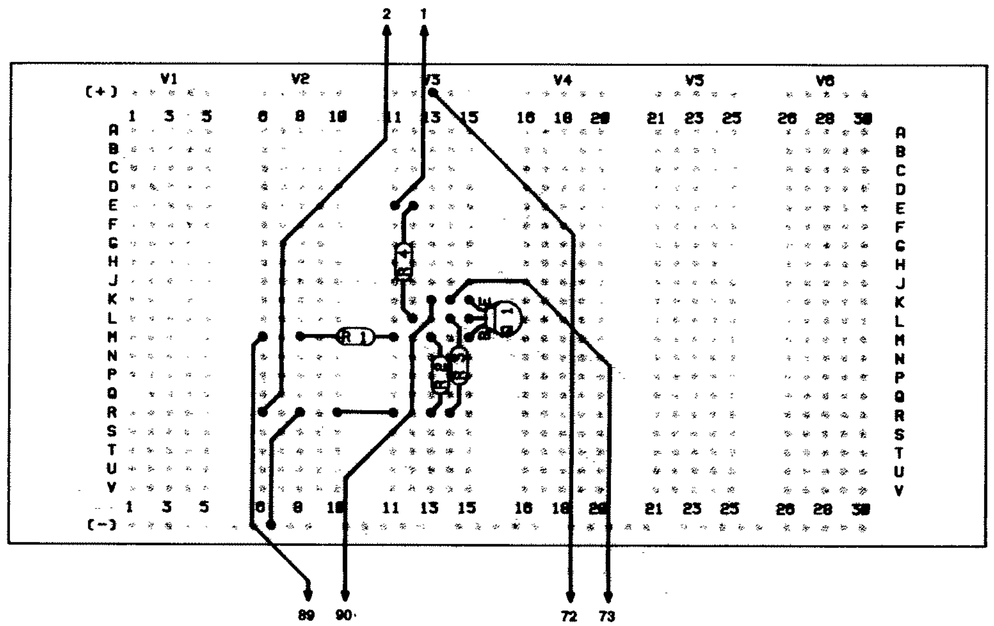
When you finish wiring up the project, turn power ON and see if the **LED** lights up. Don't be surprised that it doesn't light up: it shouldn't light at this moment. Now press **S1**, and the **LED** lights up. Release **S1**. The **LED** is still on. Can you explain why?

When **S1** is closed, Q3 conducts to light the **LED**. At the same time, Q1 and Q2 also conduct. Check the current flow on the schematic, and you'll see that once these two transistors conduct, they keep on even after you release **S1**. As long as these conduct, Q3 is also kept on and the **LED** stays lit.

Turn power OFF for retrial



PROJECT 127. RTL INVERTER



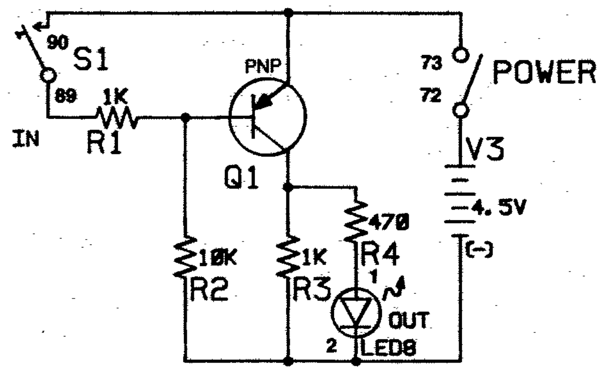
Q1	PNP	R3	1KΩ
R1	1KΩ	R4	470Ω
R2	10KΩ		

So far we've played with logic circuit made up of switches to show how various circuit such as NAND, NOR, AND, etc., work. Of course, digital circuits in the real world aren't made of switches they use transistors, diodes, resistors, etc., just like other electronic circuit.

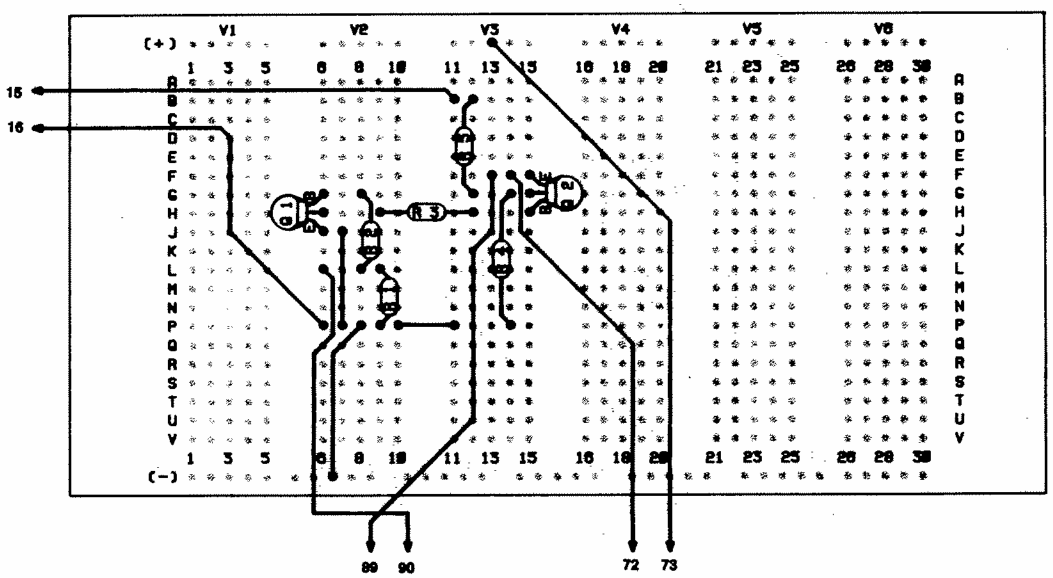
This project lets us take a look at a "real" all-electronic digital circuits. This circuit as another version of the inverter we first saw back at project 24 "Inverter Circuit." It's called an RTL inverter because it makes use of resistor-transistor logic. It's called that because the circuit is made of resistors and transistors, logically. (Sorry about that ... but we couldn't pass up that pun!)

When you turn power ON, **LED 8** lights (and that means the output is 1). Press **S1** and you make the input 1. **LED 8** immediately goes out, making the output 0. That's what an inverter does - reverses an input.

This circuit uses a transistor's ability to function as a switch. Of course, we don't have to always press the key to use a transistor as a switch - we could use the output from another circuit, couldn't we? (Of course we could!)

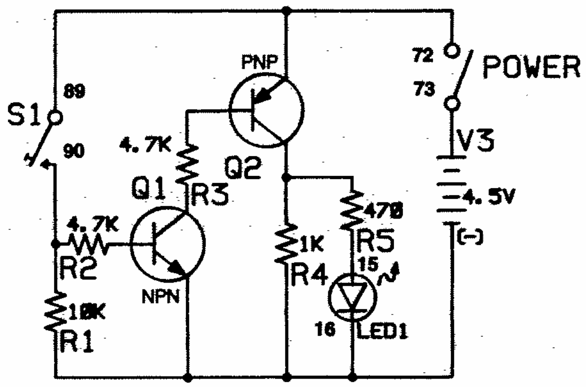


PROJECT 128. RTL BUFFER



Q1	NPN	R1	10KΩ	R4	1KΩ
Q2	PNP	R2	4.7KΩ	R5	470Ω
		R3	4.7KΩ		

If you look carefully at the schematic for this project, you'll notice that it looks an amplifier circuit. It turns out that's what an RTL buffer is.

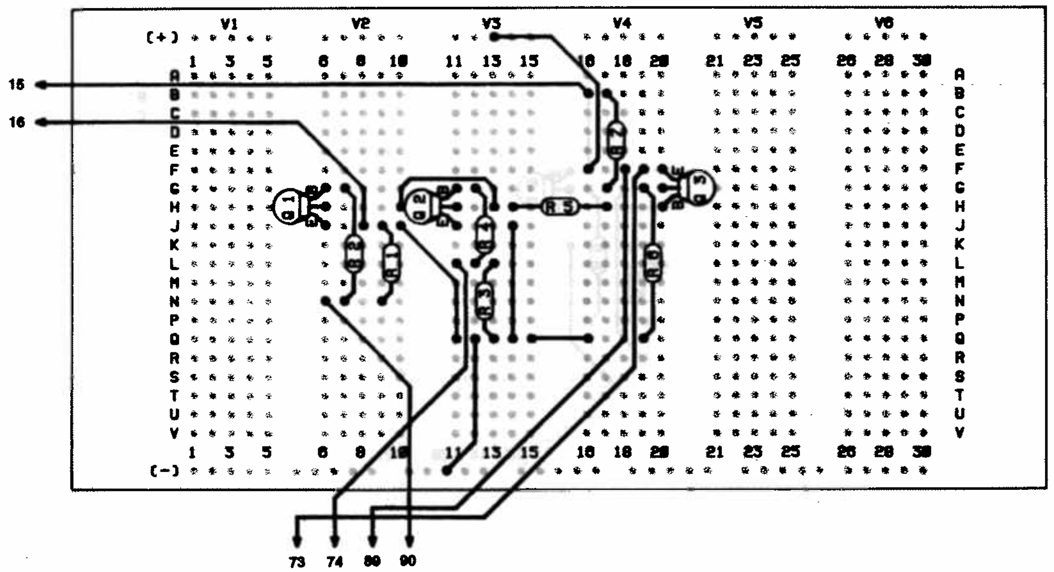


After building this circuit, turn power ON. Now press **S1**. You'll see **LED 1** light. The input is 1; the output is also 1.

So why do we need circuits like an RTL (Resistor Transistor Logic) buffer? Many times we want to control the operation of some device, such as a lamp, that requires more current than a digital circuit can deliver. A buffer circuit between the digital circuit and the external device lets us amplify the digital output enough to "drive" (that's engineering talk meaning "to operate" or "control") the external device.

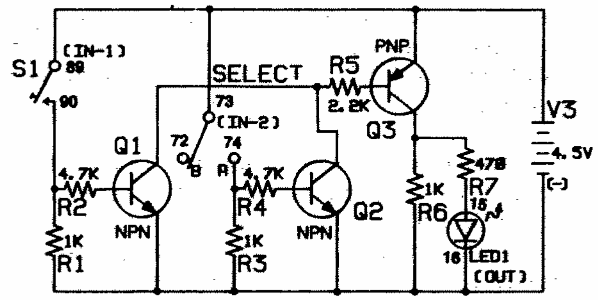
Strictly speaking, this buffer circuit isn't a true digital circuit. But buffers are important in digital electronics to help digital circuits get along with the "outside world."

PROJECT 129. RTL OR GATE



Q1	NPN	R1	1KΩ	R4	4.7KΩ	R7	470Ω
Q2	NPN	R2	4.7KΩ	R5	2.2KΩ		
Q3	PNP	R3	1KΩ	R6	1KΩ		

It's easy to make an OR gate using resistor-transistor logic. This circuit is an all-electronic version of project 25.

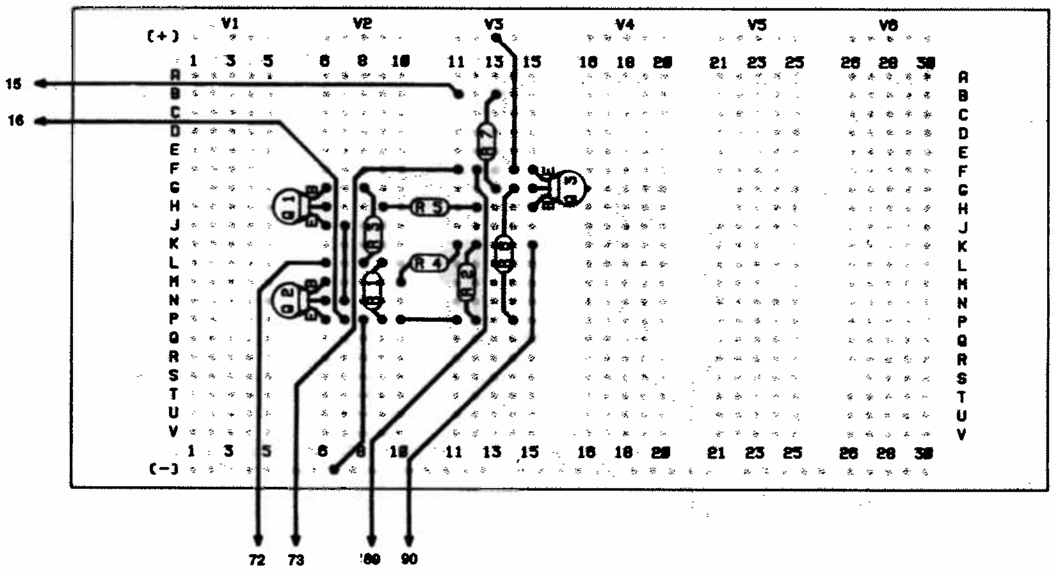


As you build this circuit, set the **select switch** up. Press **S1** and watch **LED**. What happens? Now release the key and set the **select switch** down. Does anything happen now? While the **select switch** is at the down position, press **S1** again. Is there any change?

You saw that this circuit behaves just like the OR gate in project 25. You can see why it does so by looking at the schematic. When you press **S1** or set the **select switch** down, you let current flow to the base of one of the two NPN transistors. This lets the NPN transistor operate, and in turn this causes the PNP transistor to operate and light the **LED**. And the PNP transistor operates if both NPN transistors are operating.

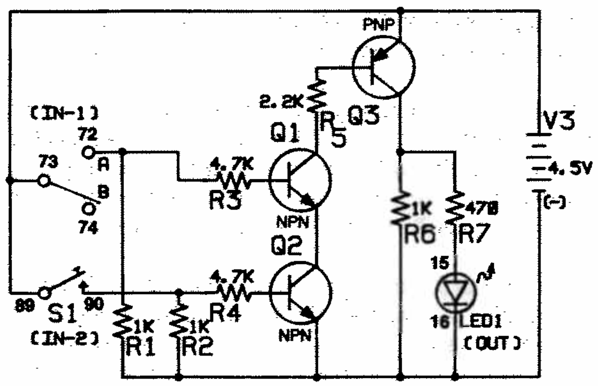
As you probably suspect, we seldom use actual switches (like the key in this project) with OR gates. Another circuit like a multivibrator can supply the input signal to turn on OR gate on or off (or make it 0 or 1, or make it high or low... well, you know what we mean!).

PROJECT 130. RTL AND GATE



Q1	NPN	R1	1KΩ	R4	4.7KΩ	R7	470Ω
Q2	NPN	R2	1KΩ	R5	2.2KΩ		
Q3	PNP	R3	4.7KΩ	R6	1KΩ		

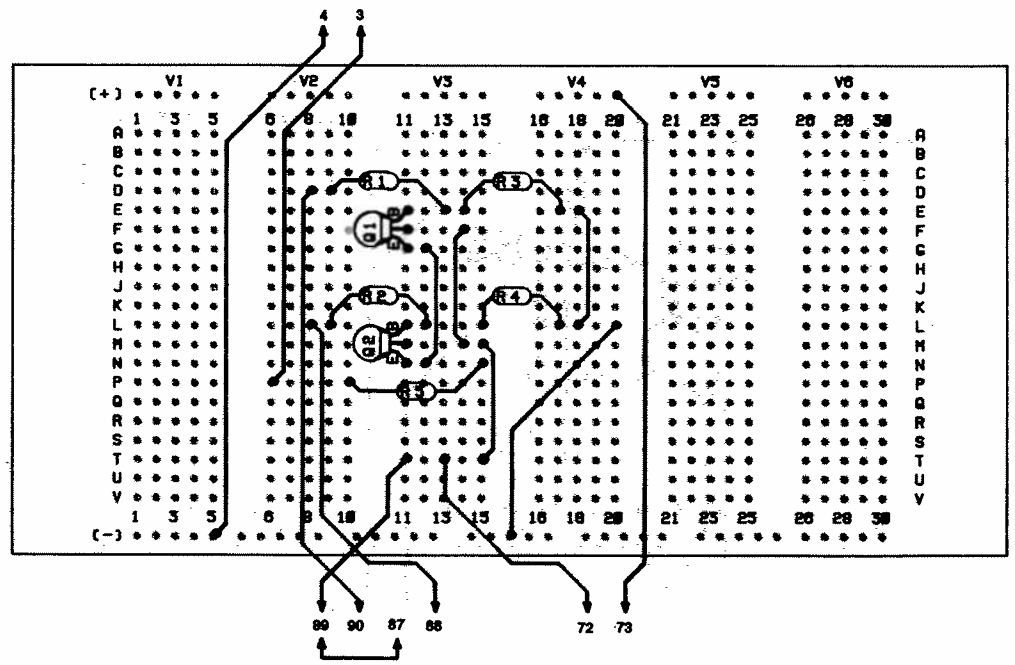
Since we just finished making an RTL OR gate, you might be wondering if there's any reason why couldn't make an AND gate using RTL. There isn't ... and this project proves it.



You can see how this circuit operates by looking at the schematic. You can see that both NPN transistors are connected in series, with the current from the collector of one going to the emitter of the other. Only when both transistors are operating can the PNP transistor operate. The PNP transistor causes the **LED** to light.

RTL digital circuits were among the first to be developed, but they are not used very often today. One big problem is that slight changes in voltage can cause RTL circuits to operate improperly. Another problem is that only a small number of devices can be connected to an RTL circuit without affecting its operation. Other digital circuits have been developed to overcome these problems. One is called DTL and we're going to look at it soon. Can you guess what DTL stands for? Make a mental guess... because we're soon going to find out.

PROJECT 131. TRANSISTOR OR GATE

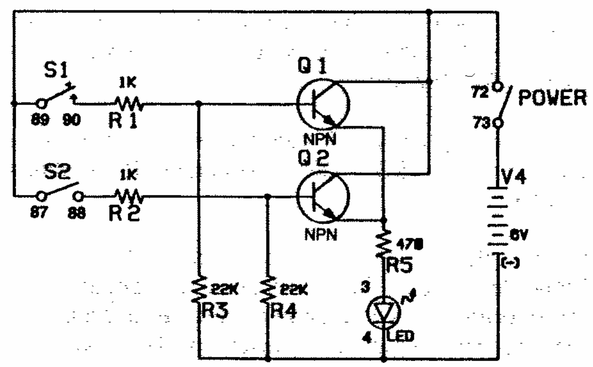


Q1	NPN	R1	1KΩ	R3	22KΩ	R5	470Ω
Q2	NPN	R2	1KΩ	R4	22KΩ		

Of course you know an OR gate is a logic circuit that has two or more input terminals and produces a high output when one or more input signals are high.

In this project, we're going to make a transistor OR circuit and see how it works. Note that the input signals are 1 when keys are pressed and at 0 if released. The output signal is 1 when the **LED** lights up, and 0 when the **LED** goes out.

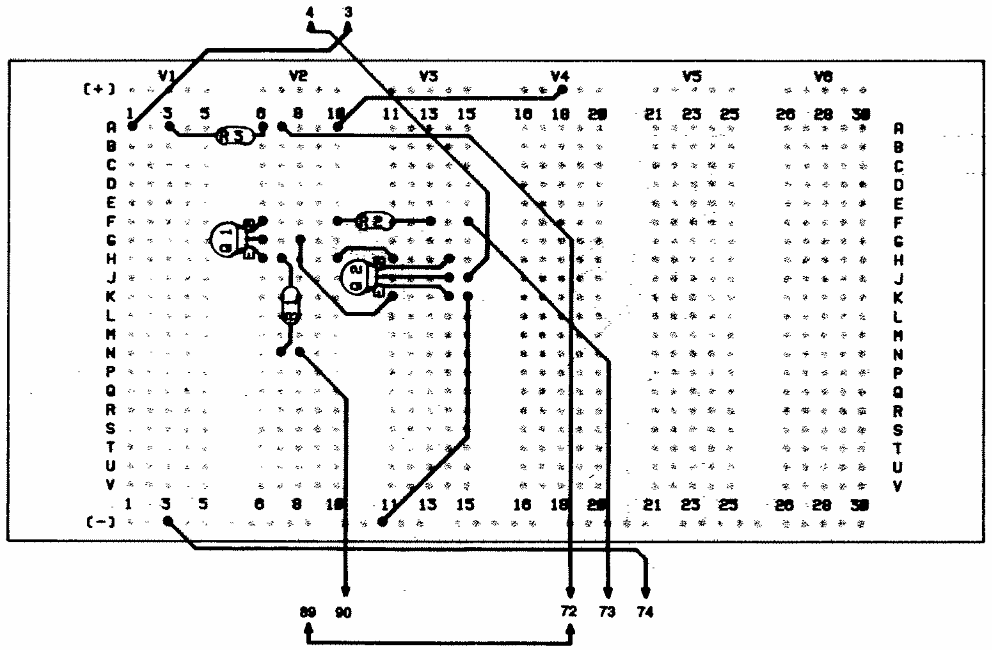
Wire the project, turn power ON, and get to the experiment. We believe no more explanation is necessary for you by now!



S1	S2	LED7
0	0	0
1	0	1
0	1	1
1	1	1

LED ON : 1 or H
LED OFF : 0 or L
Switch ON : 1 or H
Switch OFF : 0 or L

PROJECT 132. TRANSISTOR AND GATE

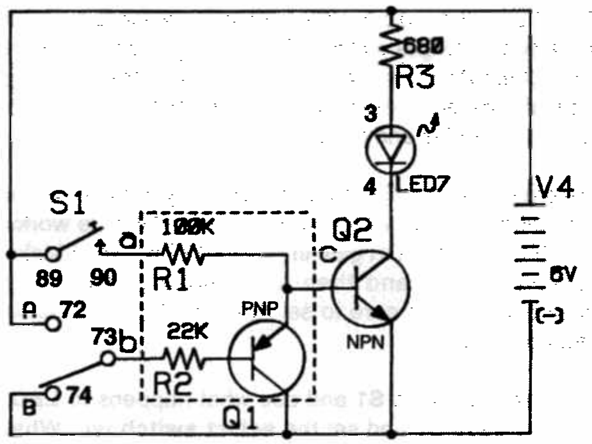


Q1	PNP	R1	100KΩ
Q2	NPN	R2	22KΩ
		R3	680Ω

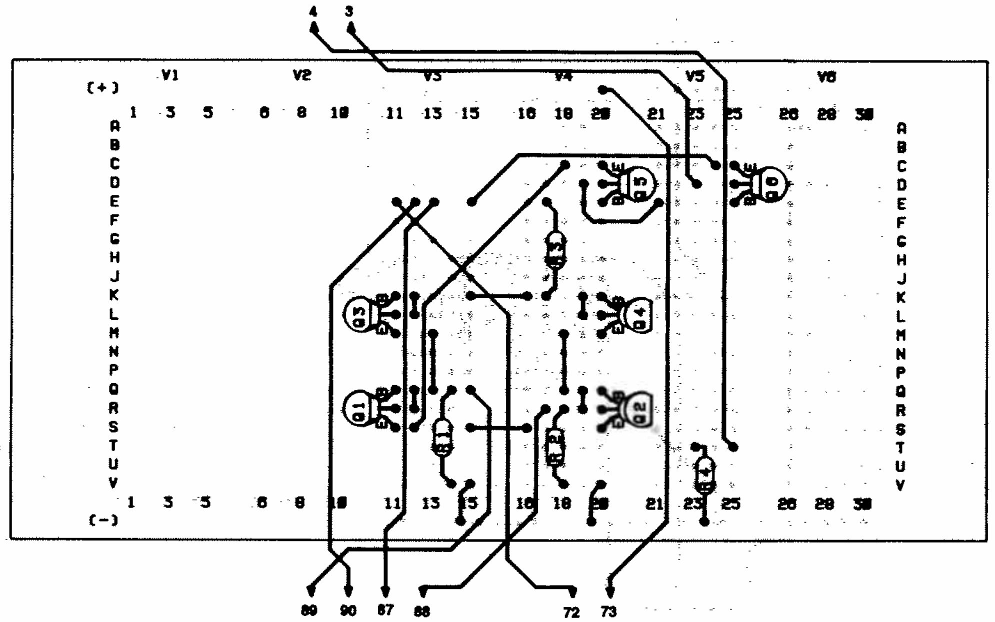
Reminder... an AND gate is a logic circuit which has two or more input terminals and generates an 1 output only when all the inputs are 1: any other combinations outputs 0.

In this project, we're going to make a transistor AND circuit and find out how it works by referring to the schematic. Remember that input signals are 1 when key is pressed or the **select switch** is up, and 0 when key is released or the **select switch** is down. The output signal is 1 when the **LED** lights up, and 0 when the **LED** goes out.

Wire the project, switch power ON, and see what happens to the LED when you changes the input condition. Again, we believe no other explanation is necessary.



PROJECT 133. TRANSISTOR XOR GATE



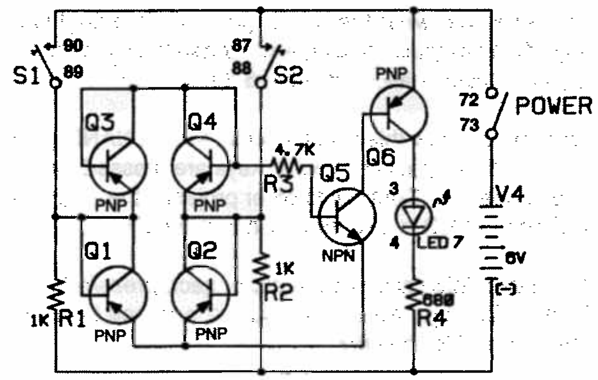
Q1	PNP	Q4	PNP	R1	1KΩ	R4	680Ω
Q2	PNP	Q5	NPN	R2	1KΩ		
Q3	PNP	Q6	PNP	R3	4.7KΩ		

An XOR Gate??? Don't be surprised if you don't know what that means... because we haven't mentioned XOR (exclusive OR) gates so far. But here's the project that lets us find out about this fascinating circuit....

An XOR gate is a logic circuit that produces an output 0 only when 1 and 0 signals are applied at the same time to two or more input terminals, and 1 when the input signals are same (both 1 or both 0).

We're going to make a transistor XOR gate and find out how it works. Note that the input signals are 1 when keys are pressed, and 0 when released. The output signal is 1 when the **LED** lights up, and 0 when the **LED** goes out.

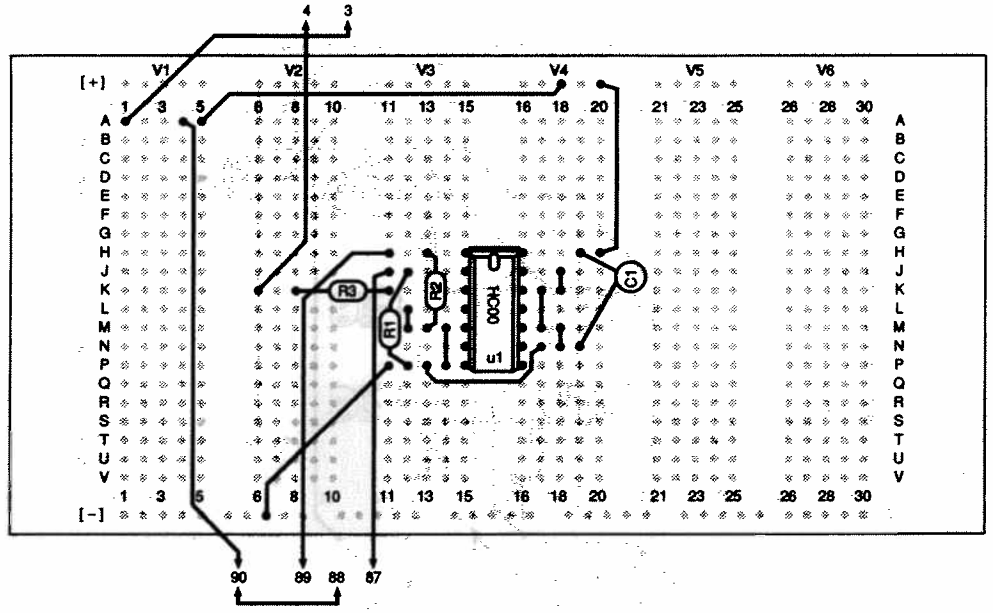
Wire the project, turn power ON, and see what happens to the **LED** when you press and release **S1** and **S2**. Figure 1 shows how this circuit works. Try following the current path on schematic.



S1	S2	LED7
0	0	0
1	0	1
0	1	1
1	1	0

LED ON : 1 or H
LED OFF : 0 or L
Switch ON : 1 or H
Switch OFF : 0 or L

PROJECT 134. SPECIAL NAND GATE

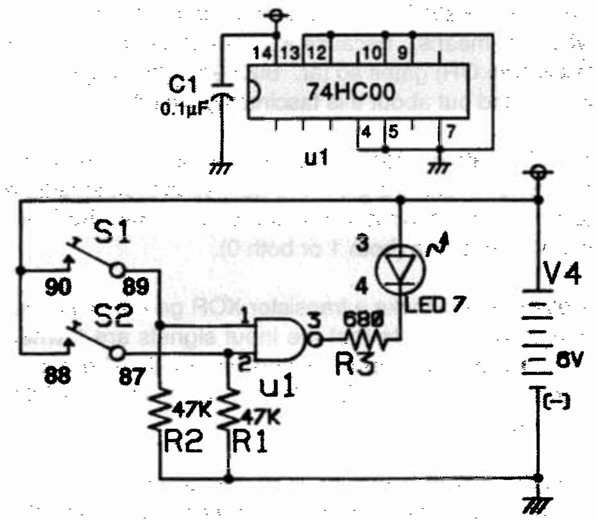


U1 74HC00 R1 47KΩ R2 47KΩ R3 680Ω C1 0.1µF

Still remember what the NAND gate is? Its output is 0 only when all inputs are at 1, and 1 when the inputs have any other combination--reverse of the AND gate.

Let's find out how this circuit works, using a NAND gate IC. Note that inputs are 1 when keys are pressed, and 0 when released. Contrary to other projects, the output is 0 when the **LED** is ON, and 1 when it is OFF.

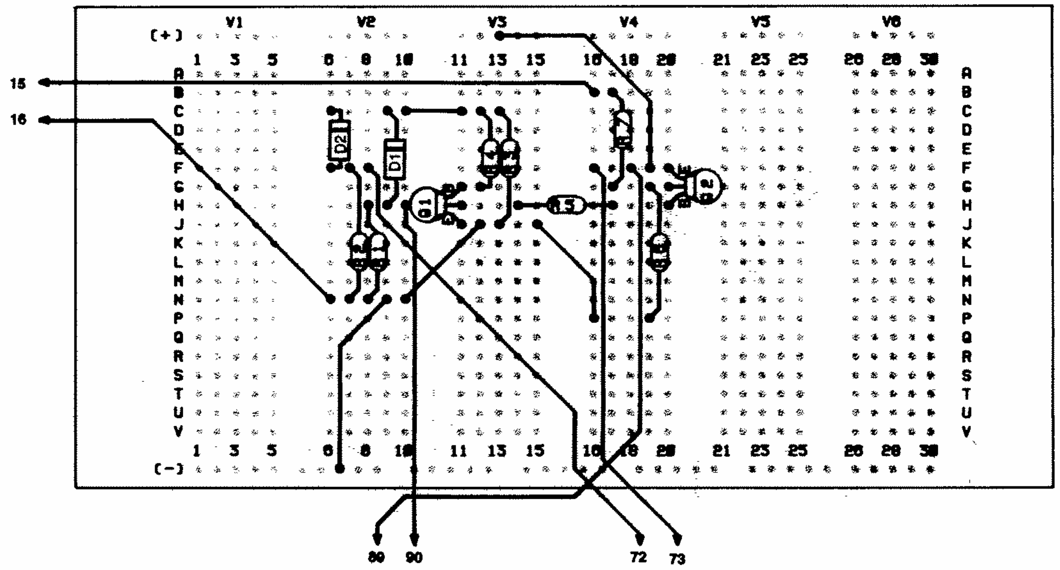
When you finish assembling the project, press/release **S1** and **S2** while watching Figure 1 and see how the **LED** blinks ON and OFF.



S1	S2	LED7
0	0	1
1	0	1
0	1	1
1	1	0

LED ON : 1 or H
LED OFF : 0 or L
Switch ON : 1 or H
Switch OFF : 0 or L

PROJECT 135. DTL OR GATE



Q1 NPN R2 1KΩ R5 2.2KΩ
Q2 PNP R3 22KΩ R6 1KΩ
R1 1KΩ R4 4.7KΩ R7 470Ω

"DTL" stands for diode-transistor logic, which is a digital circuit using diodes and transistors. As you can see from the schematic for this project, other components (such as resistors) are used but the circuit's operation depends upon the diodes and transistors.

Before you start building this project, do you think there's any difference between how this DTL OR gate works and how the RTL OR gate in project 129 worked? Make a mental note and then start making the wiring connections. Be sure to set the **select switch** down during the wiring.

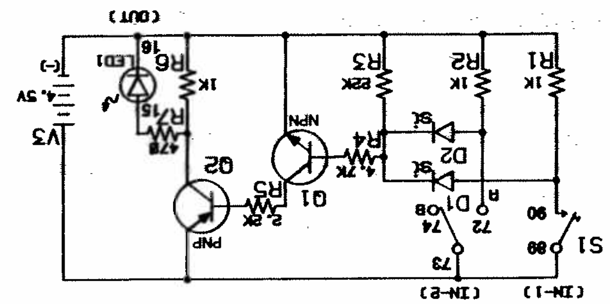
After wiring, press **S1** and see what happens to **LED**. Now release **S1** and set the **select switch** up. What happens now? Leave the **select switch** at the up position and press **S1**. Is there any change?

You discovered this circuit behaves like the other OR gates you've built. You can see how this circuits works by looking at the schematic. Pressing **S1** or setting the **select switch** up allows current to flow to the base of the NPN transistor, which then enables the PNP transistor to operate.

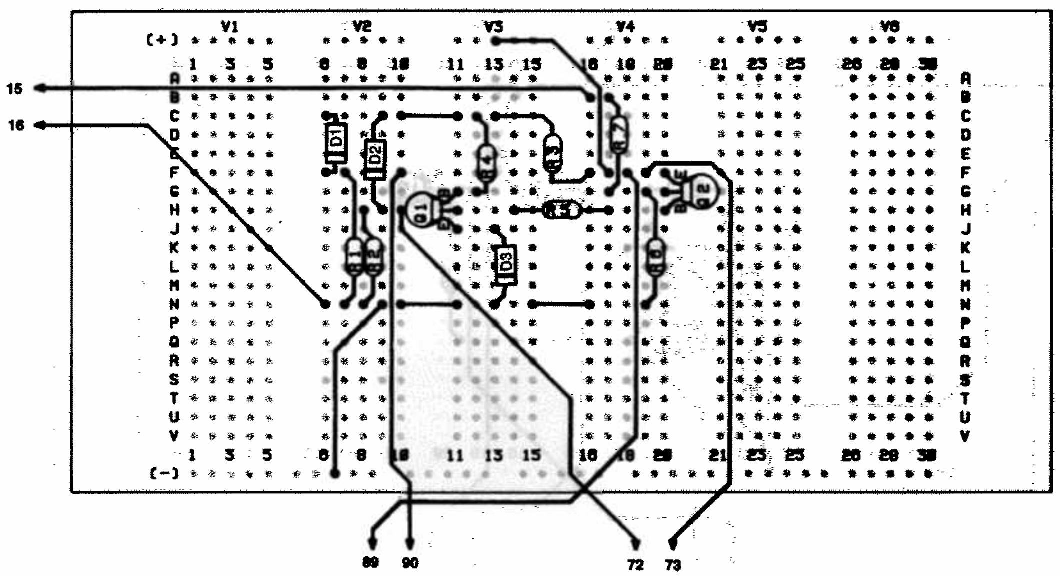
Compare your circuit with the one below (we've printed it upside-down... don't peek until you've drawn your own circuit.)

It's also possible to make an OR gate out of just two diodes and a resistor. Can you figure out how?

Be sure to draw schematics for any circuits you come up with.



PROJECT 136. DTL AND GATE



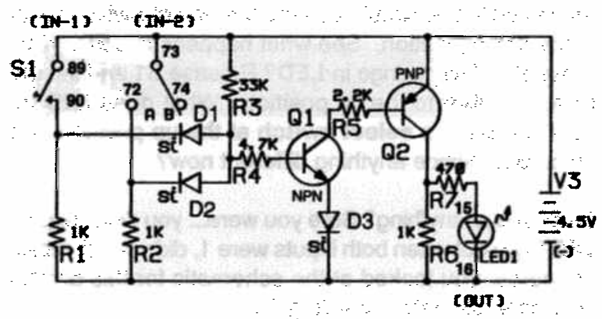
Q1	NPN	R2	1KΩ	R5	2.2KΩ	D1	Si
Q2	PNP	R3	33KΩ	R6	1KΩ	D2	Si
R1	1KΩ	R4	4.7KΩ	R7	470Ω	D3	Si

DTL can also be used in AND gates, as we'll see in this project. You'll find it works just like other AND circuits we've played with in the past.

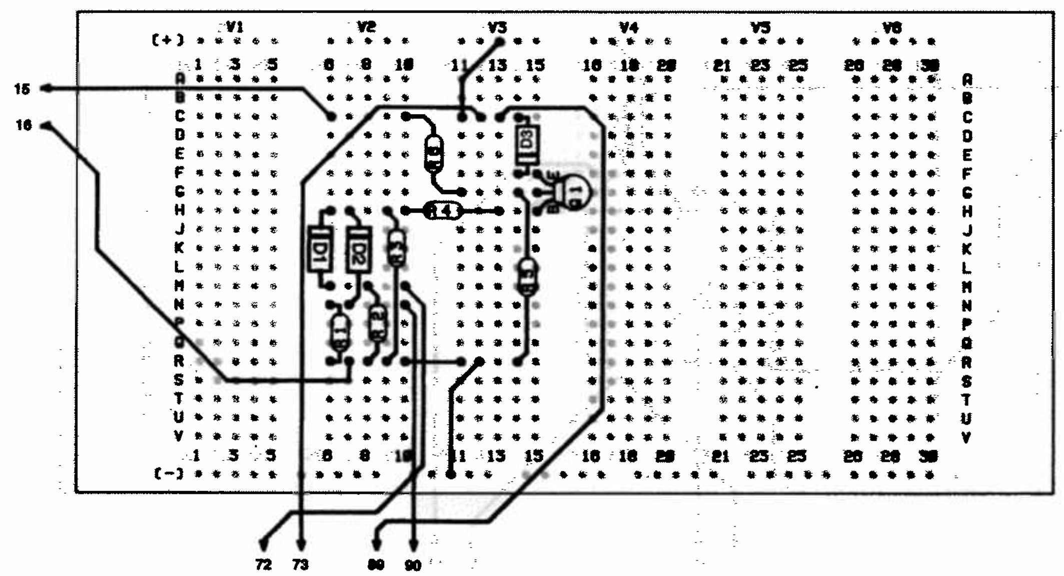
When you finish the wiring connections, you'll find that **LED** does not light until both the key is pressed and the **select switch** is up.

Can you tell how this project works by looking at the schematic? At first glance it might seem like the circuit for this project and our last one are virtually identical... can you spot the differences? (Okay, here's a clue - which way is the diode arrow pointing?)

It's also possible to make up an AND gate using just two diodes and a resistor. Can you figure out how? Be sure to make a note of what you come up with and compare it with our schematic.



PROJECT 137. DTL NOR GATE

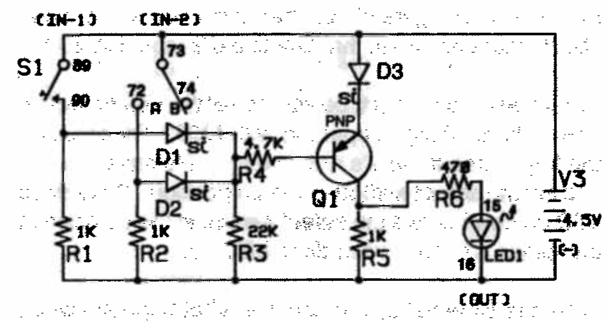


Q1	PNP	R1	1KΩ	R4	4.7KΩ	D1	Si
		R2	1KΩ	R5	1KΩ	D2	Si
		R3	22KΩ	R6	470Ω	D3	Si

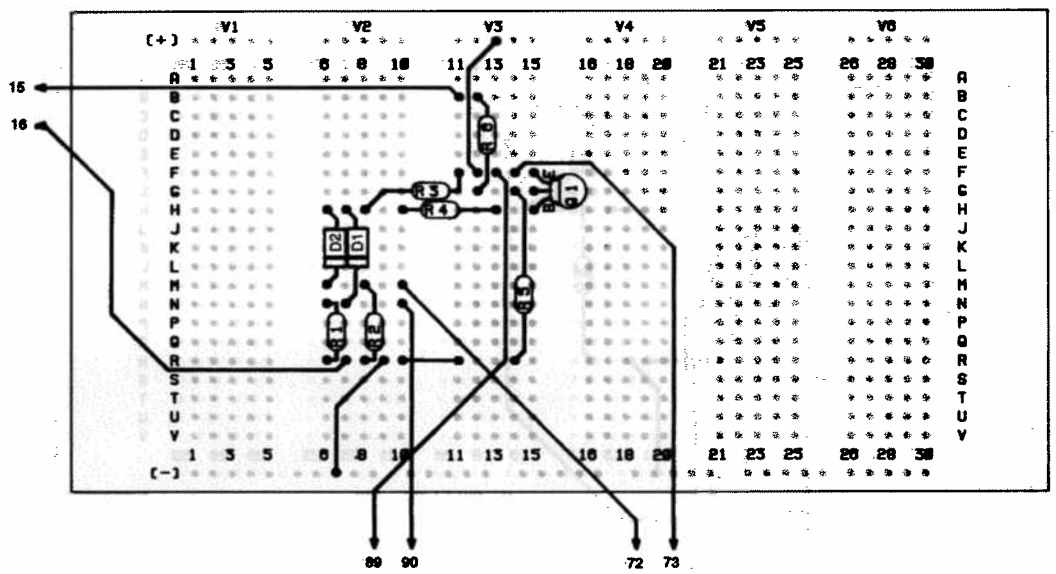
OR and AND aren't the only digital circuits we can whip up using DTL - and here's proof. If your memories gotten rusty about how a NOR gate works, take a look at back project 27 before building this circuit.

While building this project, be sure to set the **select switch** down. Is anything happening to **LED**? Press **S1** and see if there's any change in **LED**. Release **S1** and set the **select switch** up. What happens to **LED**? While the **select switch** is still set to the up position, press **S1** and see if there's any change in **LED**.

You can see how this circuit works by referring to the schematic. When **S1** is not pressed and the **select switch** is down, current can flow to the base of the transistor and **LED** lights. But if the key is pressed or if the **select switch** is at the up position, the base is connected to the positive side of the batteries. When this happens, the transistor can't operate (since there's no current flow to the base) and the **LED** goes out.



PROJECT 138. DTL NAND GATE



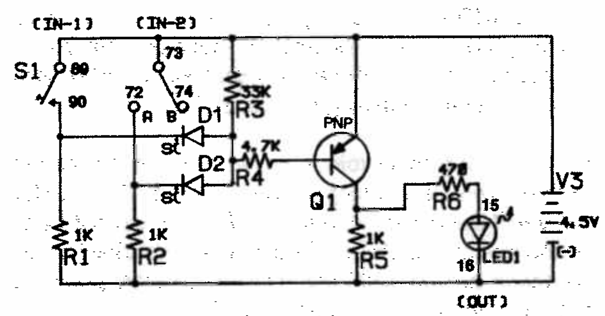
Q1	PNP	R1	1KΩ	R4	4.7KΩ
D1	Si	R2	1KΩ	R5	1KΩ
D2	Si	R3	33KΩ	R6	470Ω

Here's a NAND gate circuit that gives you some clues about what goes on in that tiny Quad 2-input NAND IC.

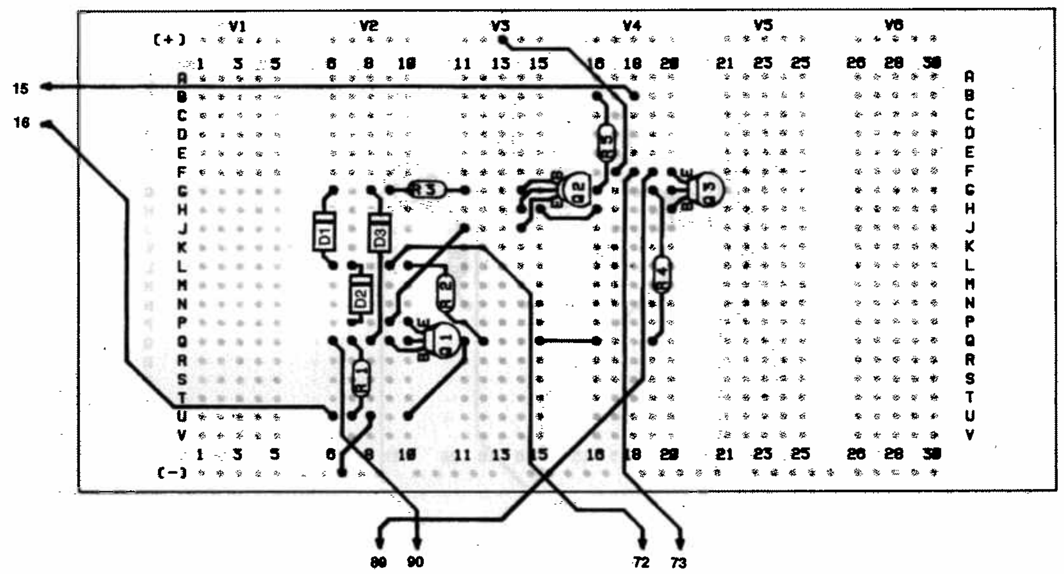
As you build this project, be sure to set the **select switch** down. Is anything happening with **LED**? Try pressing the key to see if there's any change. Release the key and set the **select switch** to the up position. What does **LED** do now? Leave the **select switch** at the up position and press **S1**. Now what does **LED** do.

From the schematic, you can see what happens when you press the key and have the **select switch** set to the up position. The base of the PNP transistor has its current flow cut off and the transistor can no longer operate. And this results in **LED** going out.

While the NAND gates in the Quad 2-input NAND IC are all-electronic, they are not DTL. They're something called "TTL" - can you figure out what that means? Take a guess, because we'll find out for sure very soon!



PROJECT 139. DTL EXCLUSIVE OR GATE



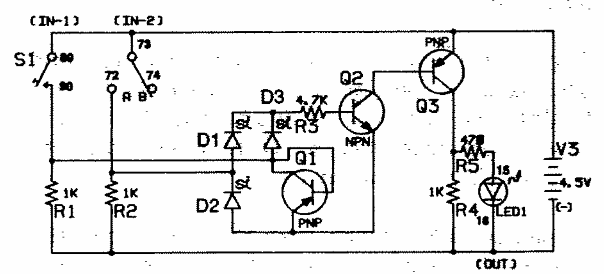
Q1	PNP	R1	1KΩ	R4	1KΩ	D1	Si
Q2	NPN	R2	1KΩ	R5	470Ω	D2	Si
Q3	PNP	R3	4.7KΩ			D3	Si

As you build this circuit, be sure the **select switch** is set to the down position. See what happens to **LED**. Now press **S1**. Any change in **LED**? Release **S1** and set the **select switch** to the up position. What does **LED** do now? Lease the **select switch** at the up position and press **S1**. Notice anything different now?

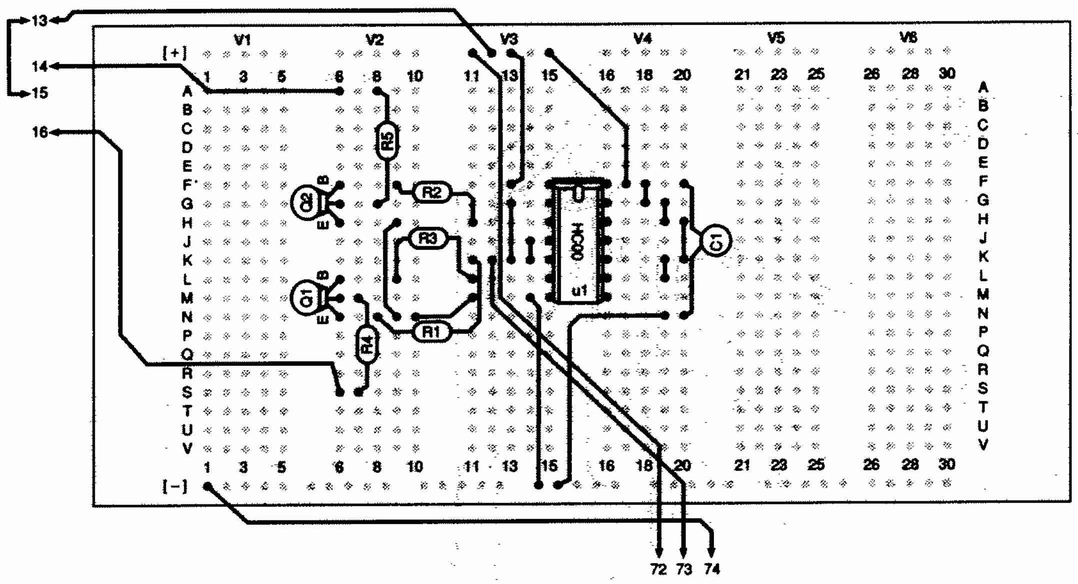
Surprised at anything? Sure you were... you didn't expect **LED** to go out when both inputs were 1, did you? (Unless, of course, you looked at the schematic for this project very carefully before building it!)

So you can see that an XOR gate has an output of 0 if both inputs are the same. If the inputs are different, then the output is 1. You can see that this is a handy circuit to let us know if we have two inputs that are the same. (Or to let us know if we have two different inputs.)

There's also an XNOR gate (exclusive NOR). We won't build one here, but maybe you can figure out how, since it's the same thing as a NOR gate followed by an inverter! (That ought to tell you how an XNOR gate should work.) Be sure to keep track of what you do in your notebook if you decide to design an XNOR gate.



PROJECT 140. C-MOS INVERTER



U1	74HC00	R1	10KΩ	R4	470Ω	C1	0.1μF
Q1	NPN	R2	10KΩ	R5	470Ω		
Q2	NPN	R3	10KΩ				

C-MOS devices have a wider voltage range and draw much less power than other logic ICs.

C-MOS devices contain P-Channel MOS FETs (metal oxide semiconductor field effect transistors) and N-Channel MOS FETS, which are connected together internally. By the way, the "C" in C-MOS stands for "complementary", because the P-and N-Channel MOS FETs complement each other. (One is on while the other is off, and vice versa.)

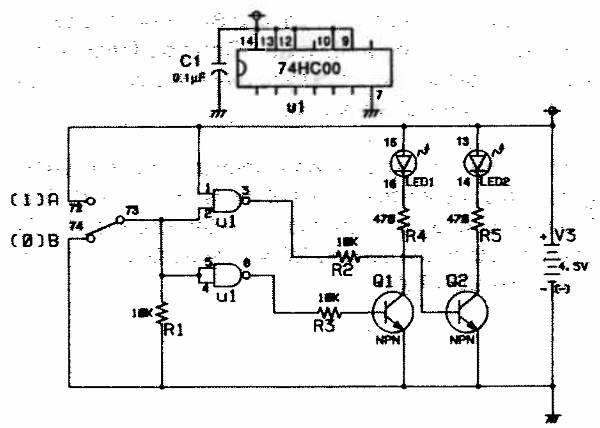
Digital C-MOS ICs contain various numbers of FETS, depending on the function of the IC.

You can see from the schematic that this project uses two of the four NAND gates in the IC.

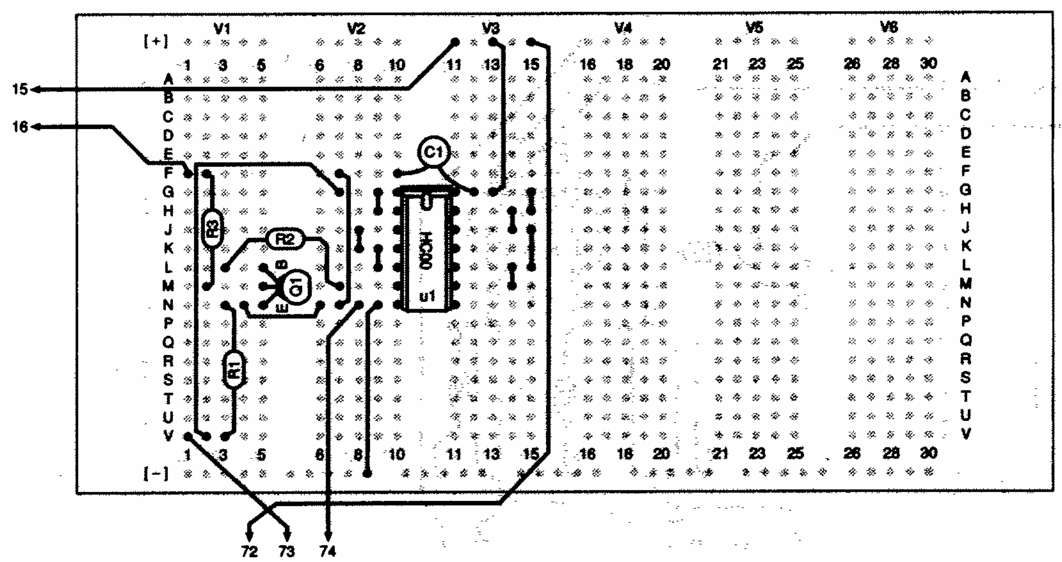
Be aware, in circuits like this where some of the gates of the IC are not used, we have to connect the unused input pins to the positive or negative power supply. Otherwise, the IC can be damaged.

As you build this project, set the **select switch** Up. You'll notice that both **LED 1** and **LED 2** are off. Since the output is 0, the input must be ... 1, of course! Now set the **select switch** down and both **LEDs** come on, indicating you're inputting 0.

You can see why this happens by looking at the schematic. With the **select switch** at the up position, both inputs to the two NANDs are 1. That means the output of both go to 0 (and turns off both transistors). When the **select switch** is set to the down, we no longer have all inputs at 1... the outputs of NANDs are 1... transistors turn on... and the **LEDs** come on.



PROJECT 141. C-MOS BUFFER



U1	74HC00	Q1	NPN	R2	10KΩ	C1	0.1μF
		R1	10KΩ	R3	470Ω		

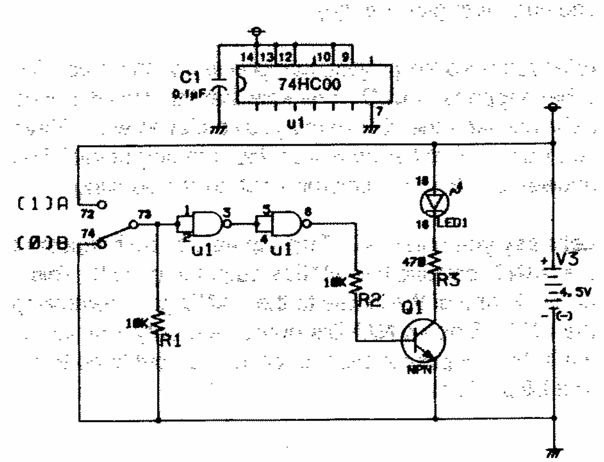
Ever wonder what happens when you start adding digital circuits together, using the output of one as the input of another? Build this project and find out.

As you can see from the schematic, we take the output from one NAND gate and use it as the input of another: we "split" the output of the first NAND and use it for the inputs to the second (so that the two inputs for the second NAND are the same). From what you know about NANDs, what do you think happens if the input to the first NAND is 1? If the first input is 0? Try to figure it out before building this project.

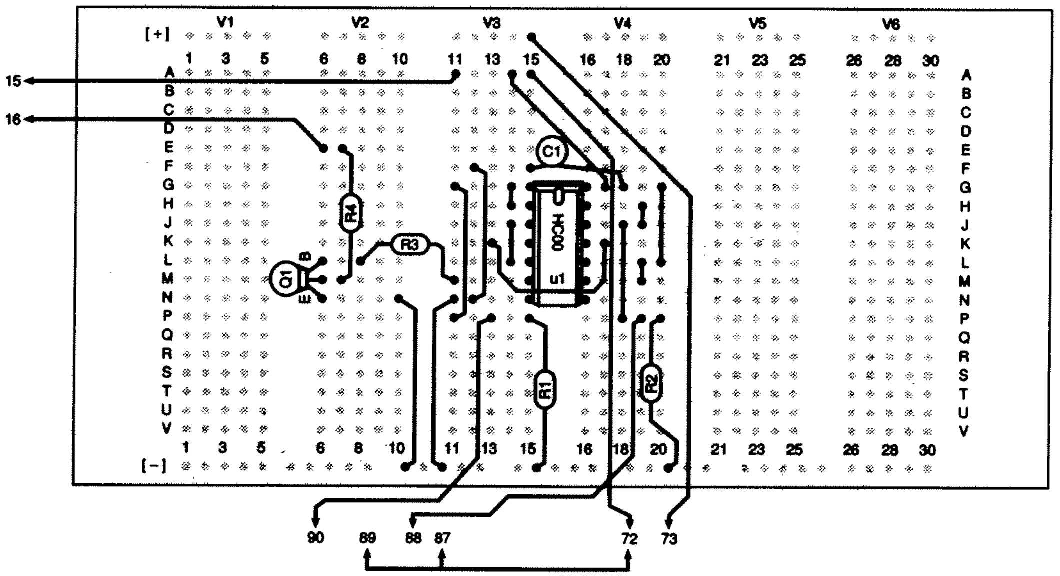
After you finished the wiring connection, set the **slide switch** down. What does **LED 1** do? Now set the **select switch** up. **LED 1** should come on.

As you've probably figured out, the input is 1 when the **select switch** is up, 0 when the **select switch** is down. What happens is, when the input to the first NAND is 1, its output is 0. The 0 input to the second causes its output to become 1, lighting the **LED**. Sometimes we use buffer circuits to simply keep two portions of a device isolated from each other. This type circuit does that job well.

One of the amazing things to think about is how large the RTL and DTL circuits were that we played with in earlier projects. Believe it or not, four circuits like that have been shrunk down to fit inside that tiny IC! There's even a special type of IC, which are actually computers shrunk to miniature size. They're called microprocessors. The process which lets us several circuits inside just one IC is called large-scale integration, or LSI. You'll see this term often used describe ICs.



PROJECT 142. C-MOS OR GATE



U1	74HC00	R1	10KΩ	R3	10KΩ	C1	0.1µF
Q1	NPN	R2	10KΩ	R4	470Ω		

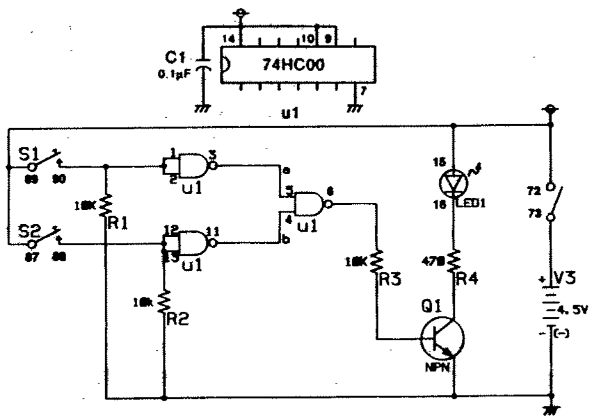
One of the nice things about the Quad 2-Input NAND IC is the way the four NAND gates can be combined to make up other logic circuits. It's possible to make up an OR gate from NANDs - as this project proves.

Our last project has given some clues about how we can use NANDs to make up other logic circuits. Take a look at the schematic for this project - can you trace what happens from each input to the eventual output? (Sure you can - just give it a try!)

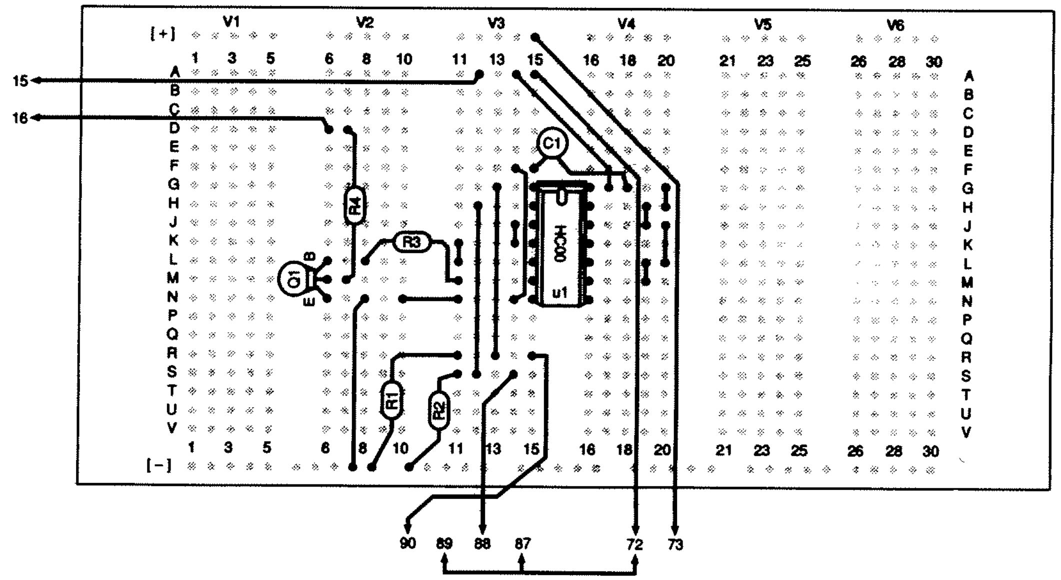
After you build the project, turn power on, and press **S1**. What happens to **LED**? You saw that this circuit indeed behaves like other OR gates you've played with. Have you traced what happens from input to output yet? The answer is in the next column - but no fair peeking.

Let's say you press **S1**. This inputs 1 to both inputs of the NAND, causing the NAND's output to go to 0. This 0 output is one of the inputs to the NAND gate controlling the **LED**. Since a NAND's output is 0 only if all inputs are 1, the 0 input causes the NAND's output to go to 1... and **LED 1** lights!

We can make up AND, NOR and XOR gates using the Quad 2-Input IC. Can you figure out how we would connect the NANDs to do this? Take your best guess and make notes ... because we're soon going to find out.



PROJECT 143. C-MOS AND GATE



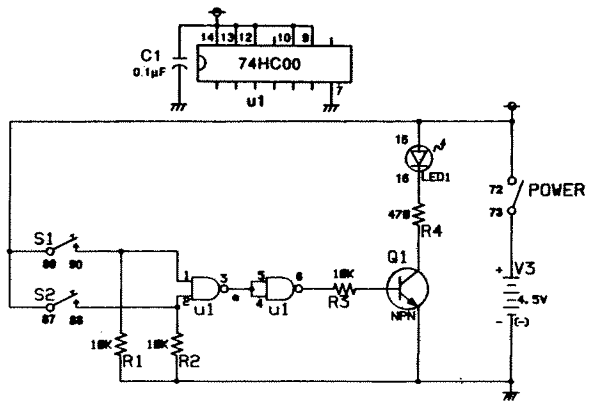
U1	74HC00	R1	10KΩ	R3	10KΩ	C1	0.1µF
Q1	NPN	R2	10KΩ	R4	470Ω		

Did you figure out how to make an AND gate using the NAND gates? Hope so - and now we'll see how right you were.

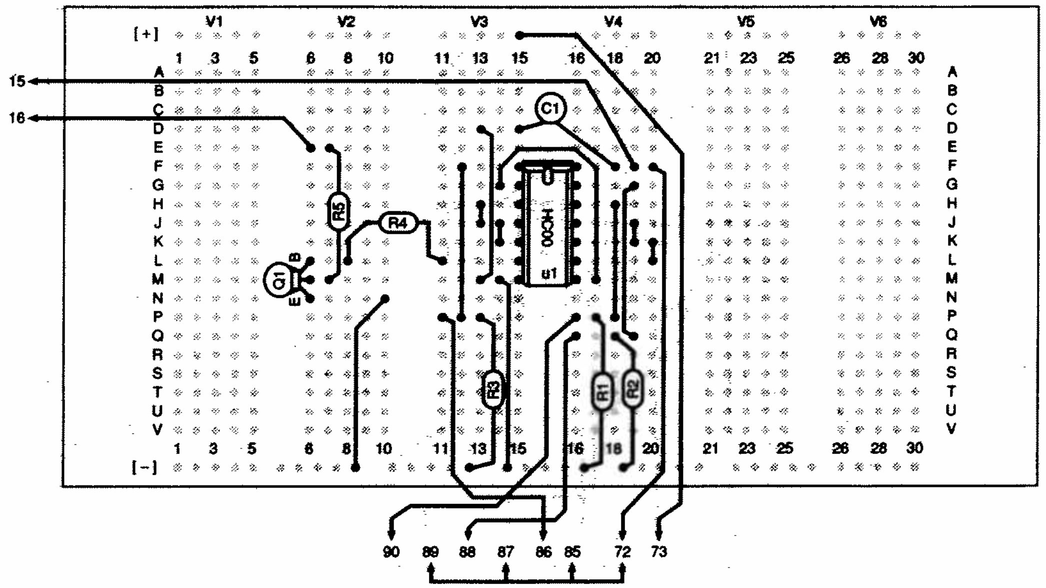
When you've finished the wiring connection, turn power ON and press **S1**. What does **LED 1** do? Now press **S2** while keep pressing **S1**. Is there any change in **LED 1**?

As you saw, pressing both **S1** and **S2** inputs 1, causing a 1 output from the AND gate. Can you follow the 1 input through the circuit until you reach a 1 output? Try it - and don't peek at the answer.

It works like this - each 1 input goes into the first NAND gate. This causes the output of the NAND to be 0. This 0 output is used for both inputs to the second NAND. The 0 inputs to the second NAND cause its output to be 1, and the **LED** lights.



PROJECT 144. C-MOS 3-INPUT AND GATE



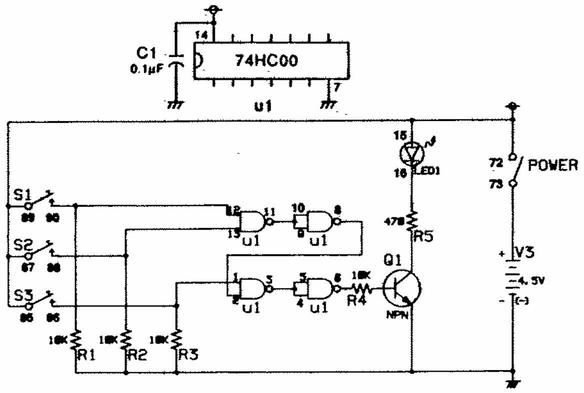
U1	74HC00	Q1	NPN	R2	10KΩ	R4	10KΩ	C1	0.1µF
		R1	10KΩ	R3	10KΩ	R5	470Ω		

Even though we've been playing with digital circuits that have just two inputs, that doesn't mean we can't have more than two inputs. Here's a C-MOS AND gate that has three inputs. Try and figure out how 3 inputs produce a 1 output from the schematic.

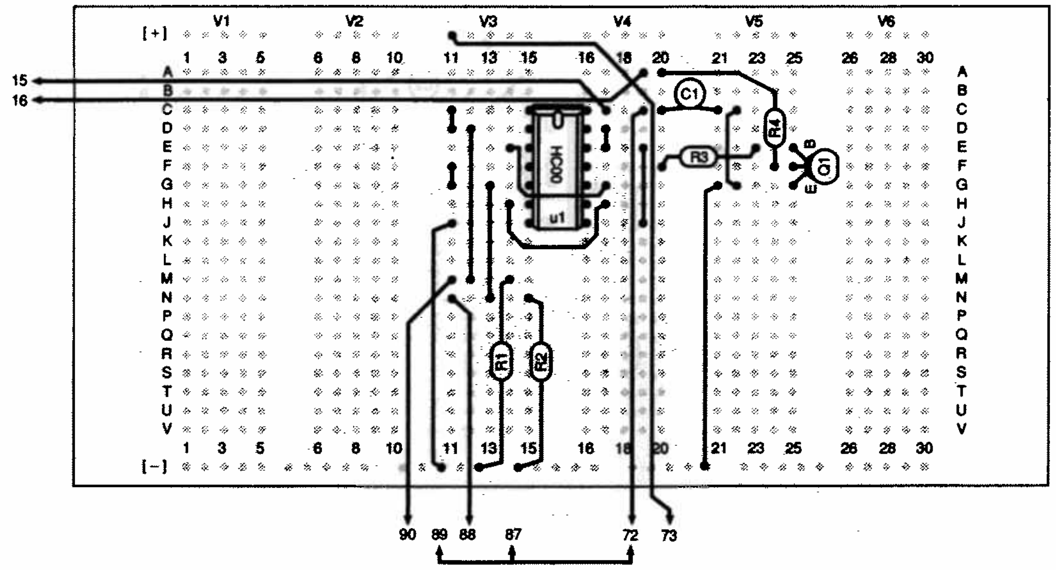
After you've finished the wiring, turn power ON. You know by now how AND gate works, so we won't go into detail here. But can you look at the schematic and figure out the setting for the three keys so that you get a 1 output? Try it ... and then see if you were right.

Here's how this circuit works: **S1** and **S2** are connected to one NAND. When they are both 1, the NAND outputs a 0. This 0 then makes up the input of another NAND, causing its output to become 1. This 1 output then goes to another NAND gate (see it on the schematic?). There it makes up one input, along with the input from S3 making up the other. When these are both 1, the NAND's output goes to 0. This output is used for both inputs of the second NAND, causing it to become 1 ... and the **LED** lights.

Seems simple, doesn't it? Believe it or not, even complex computers operate by using the same basic principles we're using with the digital circuits in this kit.



PROJECT 145. C-MOS NOR GATE



U1	74HC00	R1	10KΩ	R3	10KΩ	C1	0.1µF		
Q1	NPN	R2	10KΩ	R4	470Ω				

Try the same thing you've been doing with the past few projects... trace the "logic flow" of this circuit! Start with a 0 or 1 input and see how this circuit arrives at a 0 or 1 output. Give it a good try... and don't peek at the answer.

The kit contains the Quad 2-input NOR Gate ICs and you can organize the NOR Gate by using NAND Gate.

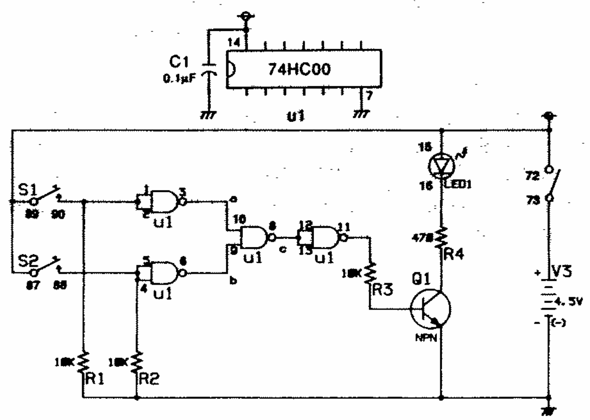
When you finish the wiring connections, turn power ON and press **S1**. Is there any change in **LED 1**? Release **S1** and press **S2**. What happens to **LED 1** now?

While keep pressing **S2**, press **S1** ... does anything different happen?

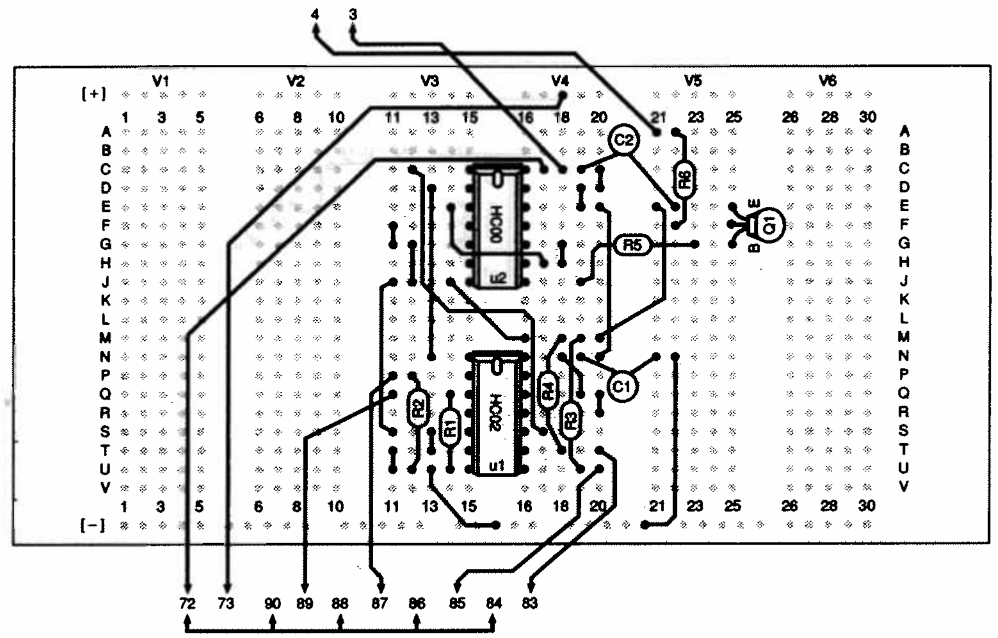
As you just saw, this project behaves just like other NOR gates we've built. And it does so because...

... pressing **S1** or **S2** inputs a signal of 1. This is used for both inputs of the NANDs marked a or b. a and b have an output of 0 with an input of 1, and their outputs are used for the inputs to the NAND marked c. As long as one or both inputs are 0, the NAND marked c has an output of 1. This 1 output is used for the inputs of the next NAND, causing an output of 0 .... and goes out the **LED**.

Don't believe us? Try putting a "0" or "1" on the schematic - see how it changes.



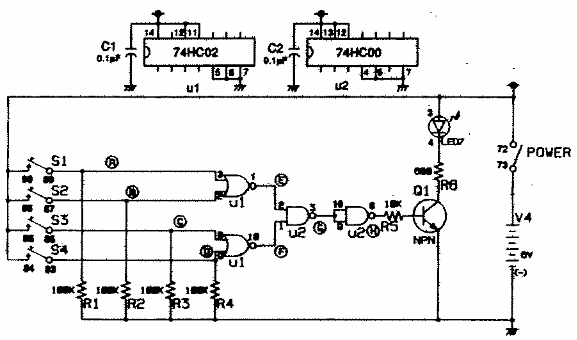
PROJECT 146. C-MOS 4-INPUT NOR GATE



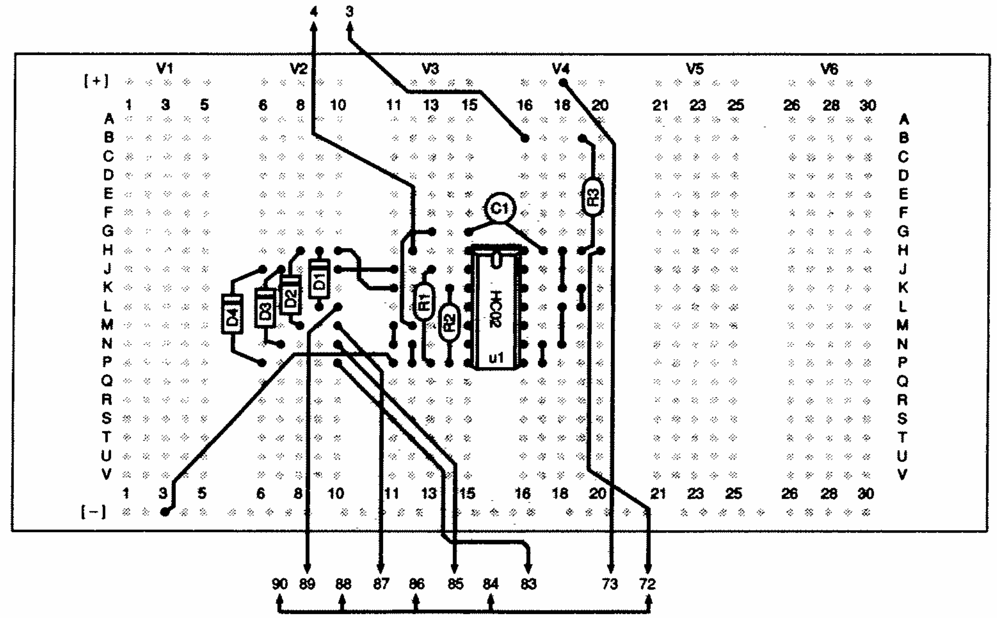
U1	74HC02	R1	100KΩ	R3	100KΩ	R5	10KΩ	C1	0.1μF
U2	74HC00	R2	100KΩ	R4	100KΩ	R6	680Ω	C2	0.1μF
Q1	NPN								

We're going to make a 4-input NOR gate using two 2-input NOR gates and two 2-input NAND gates. The output of this NOR gate is 1 only when all four inputs are at 0, and it goes to 0 if any of the four inputs is at 1.

After wiring the project, switch power ON and see what happens to the **LED**. It lights up (output is 1). This is because **S1 - S4** are still OFF at this time and the inputs are at 0. Press any of **S1 - S4**, and the **LED** goes out. See how the **LED** blinks ON and OFF as you press/release the four keys.



PROJECT 147. C-MOS 4-INPUT NOR GATE II



U1	74HC02	R3	680Ω	D2	Si
R1	220KΩ	C1	0.1μF	D3	Si
R2	220KΩ	D1	Si	D4	Si

We can make a multi-input NOR gate by combining diodes with a 2-input NOR gate. In this project, we're going to use four diodes to build this NOR gate.

S1	S2	S3	S4	F	LED7
0	0	0	0	1	0
1	0	0	0	0	1
0	1	0	0	0	1
1	1	0	0	0	1
0	0	1	0	0	1
1	0	1	0	0	1
0	1	1	0	0	1
1	1	1	0	0	1
0	0	0	1	0	1
1	0	0	1	0	1
0	1	0	1	0	1
0	0	1	1	0	1
1	0	1	1	0	1
0	1	1	1	0	1
1	1	1	1	0	1

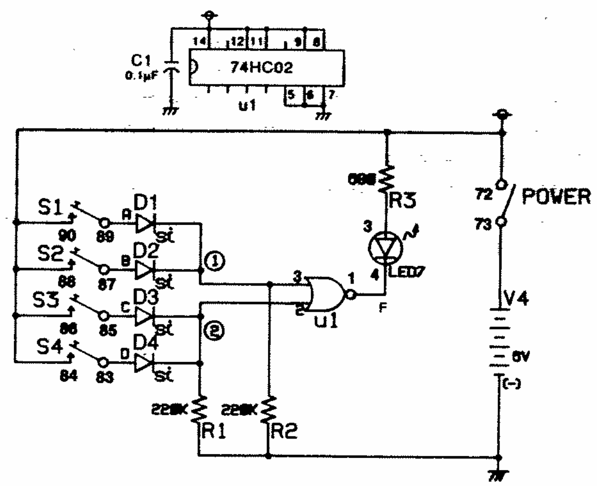
Switch ON : 1 LED7 ON : 1
Switch OFF: 0 LED7 OFF: 0

Figure 1

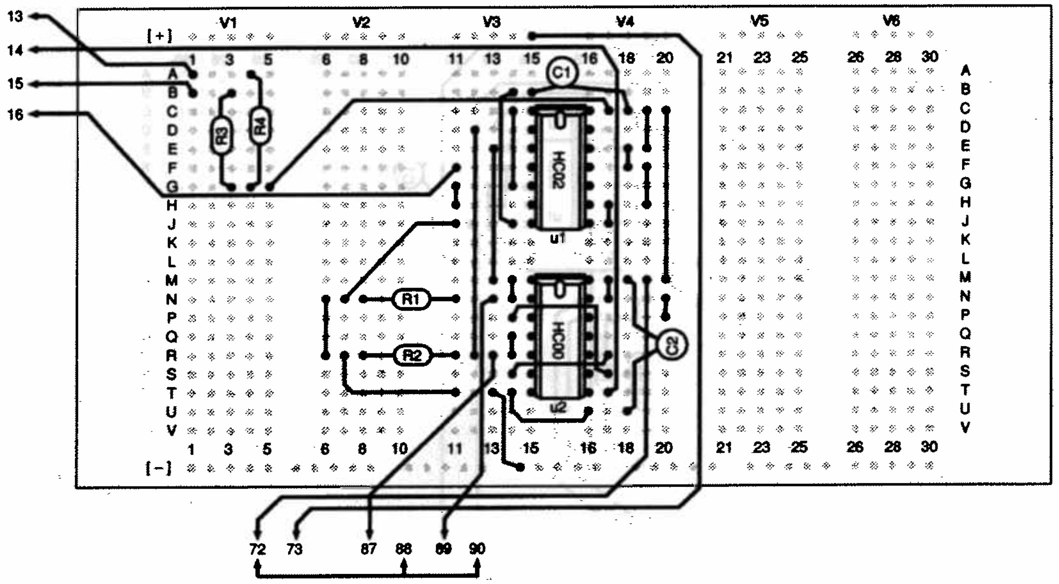
As you'll see in Figure 1, the output of this NOR gate is 1 only when all four inputs are 0.

In this project, the output level is 1 when **S1 - S4** are ON and 0 when they are OFF, and it is 0 when the **LED** is ON and 1 when it is OFF.

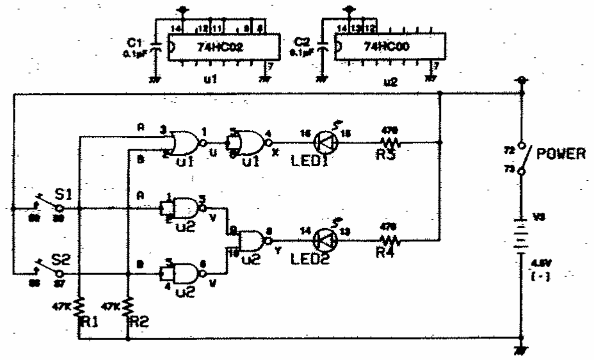
Wire the project and switch power ON, and see what happens to the **LED**: it stays OFF. This means that the output is 1 because **S1 - S4** are Off and all inputs are at 0. Now press any of **S1 - S4**, and you'll see the **LED** lights up to indicate the output is 0. See how the **LED** blinks ON and OFF while you look at Figure 1.



PROJECT 148. DE MORGAN'S THEOREM



U1	74HC02	R1	47KΩ	R3	470Ω	C1	0.1μF
U2	74HC00	R2	47KΩ	R4	470Ω	C2	0.1μF



Have you ever heard of De Morgan's theorem? It's one of the basic theorems we encounter in logical mathematics. In this project, we're going to prove this theorem which is expressed as follows.

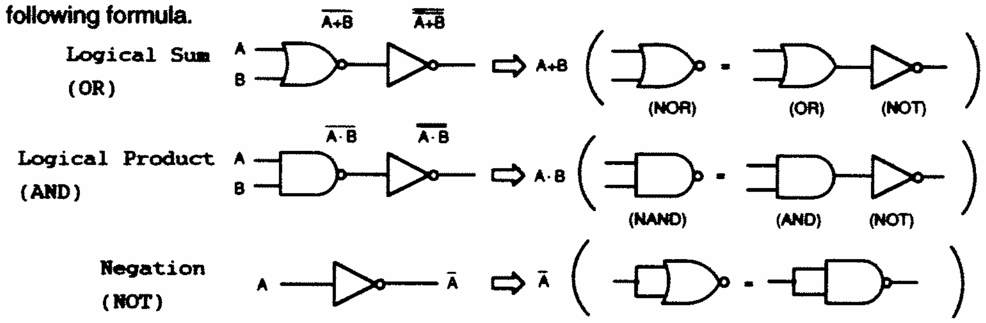
$$\overline{A + B} = \overline{A} \bullet \overline{B}, \quad \text{or} \quad A + B = \overline{\overline{A} \bullet \overline{B}}$$

Not sure what this means? Well, in ordinary language, this can be expressed as

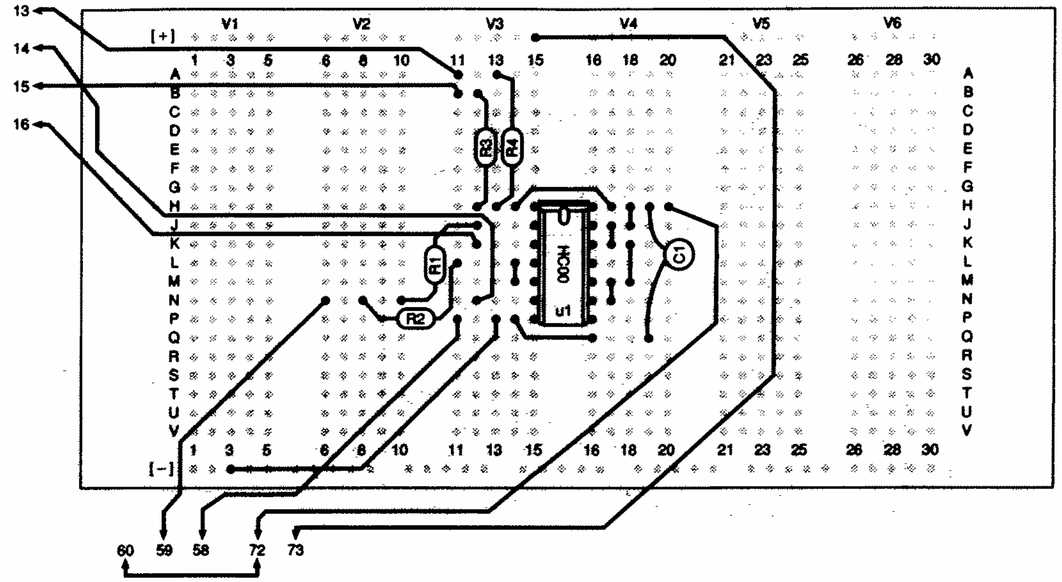
The OR gate can be constructed by an AND gate and an inverter.

De Morgan's theorem says that the gate circuit (U1) for **LED 1** and the gate circuit (U2) for **LED 2** are logically equal. Let's check this by turning power ON after wiring the project. Use **S1** and **S2** to see that the lighting statues of **LED 1** and **LED 2** match with each other. De Morgan's theorem also tells us the following formula.

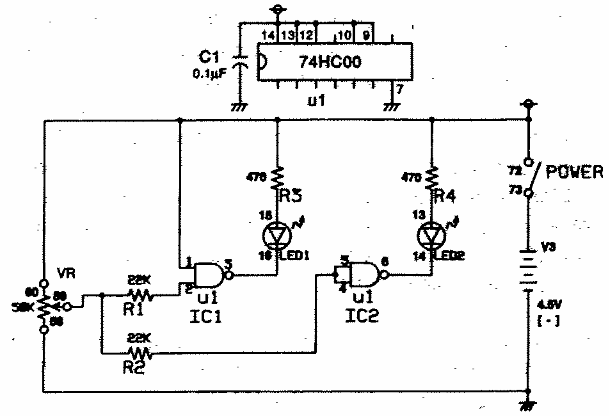
$$\overline{A} + \overline{B} = \overline{A \bullet B}$$



PROJECT 149. EXPERIMENT OF THRESHOLD VOLTAGE



U1	74HC00	R1	22KΩ	R3	470Ω	C1	0.1μF
		R2	22KΩ	R4	470Ω		



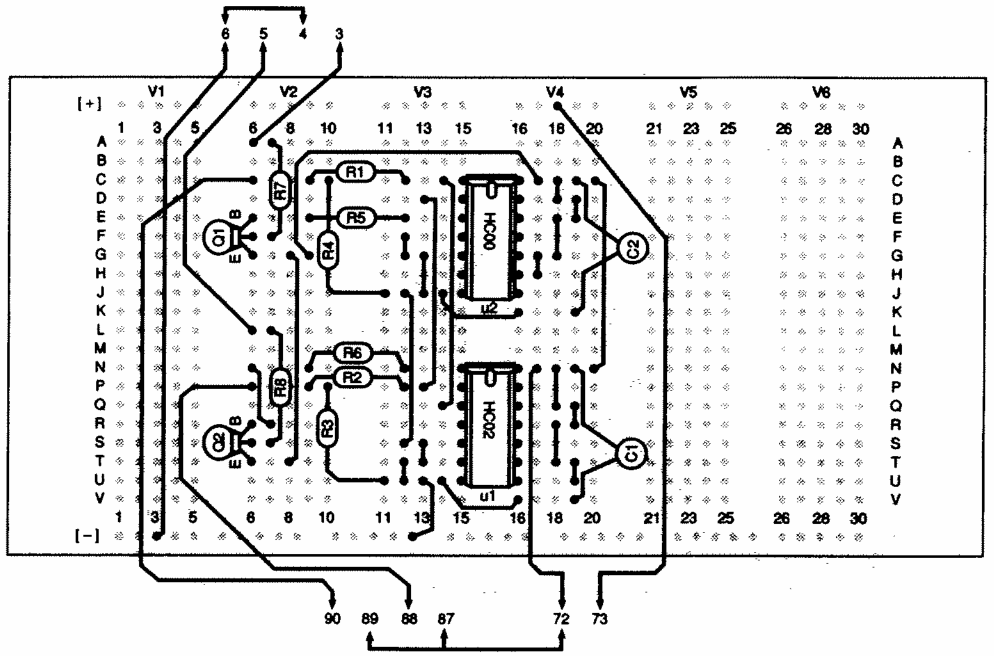
The NAND gate can be used as an inverter by setting one of its inputs to 1 or by applying one signal to both inputs. When the input is 1, the output is 0, and when the input is 0, the output is 1.

In this project, we're going to make two inverter circuits using the NAND gate to find how the level of input (threshold level) changes the output.

Turn the **control volume** fully counterclockwise and switch power ON. At this time, the output from the two inverters is at 1 level, so the **LEDs** don't light up. Now rotate the **control volume** clockwise. **LED 1** lights up, and that indicates the threshold voltage of IC1. Turn the **control volume** further. What happens now? **LED 2** lights up: this indicates the threshold voltage of IC2.

Did you notice that the inverter circuits using the same NAND circuit show different threshold voltages depending on how you use it?

PROJECT 150. NAND/NOR AND TRANSISTOR SWITCH



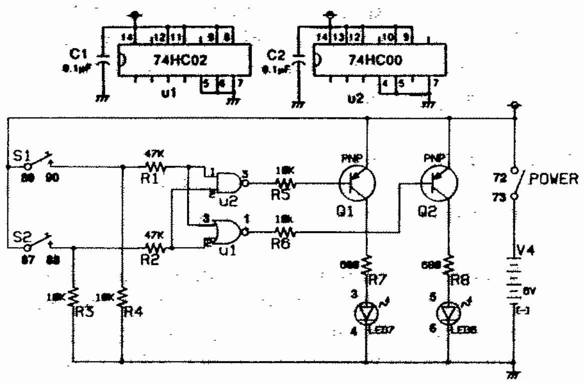
U1	74HC02	Q2	PNP	R3	10KΩ	R6	10KΩ	C1	0.1μF
U2	74HC00	R1	47KΩ	R4	10KΩ	R7	680Ω	C2	0.1μF
Q1	PNP	R2	47KΩ	R5	10KΩ	R8	680Ω		

In this experiment, we're going to find out how NAND and NOR circuits work. Transistors and **LEDs** are used to indicate the function of these logic circuits.

Table 1 shows the NAND circuit function, and Table 2 the NOR circuit function. The input "1" shown in these tables means that keys **S1**, **S2** are ON, and the input "0" means that they are OFF. As for the **LED** doesn't light up when the output is "1" but does when it is "0". Remember these points.

When you finish wiring up the circuit, switch the power ON. At this time, **S1** and **S2** are OFF (input is 0). This means the output of both the NAND circuit and NOR circuit is 1. So, neither **LED** lights up.

Now, see what happens when you press **S1**. From the tables, you'll see that the output of the NAND circuit is 0 only when the two inputs are 1, and that of the NOR circuit is 1 only when the two inputs are 0.



NAND

INPUT		OUTPUT
S1	S2	LED7
0	0	1
0	1	1
1	0	1
1	1	0

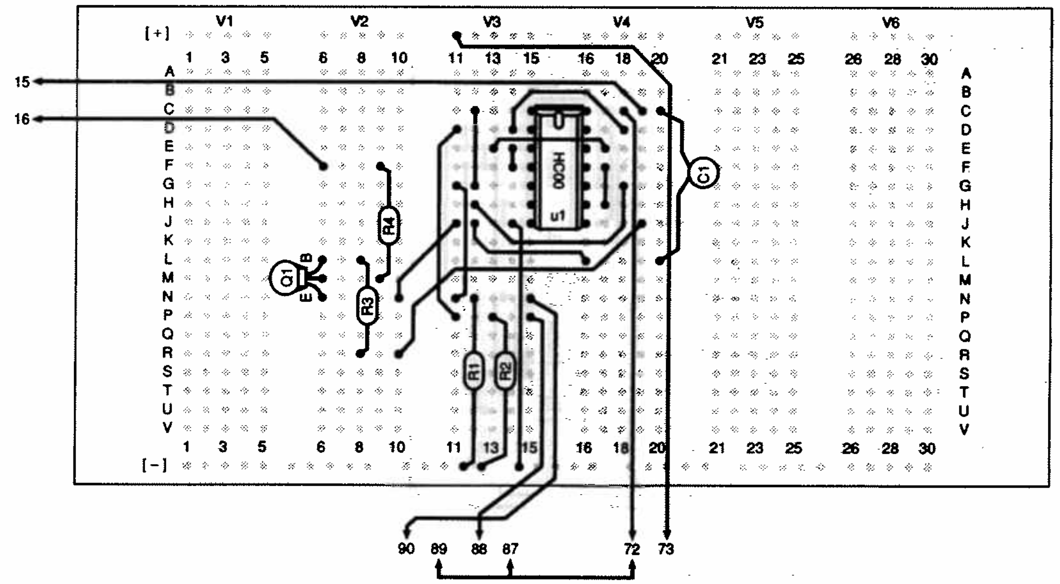
Table 1 NAND Circuit

NOR

INPUT		OUTPUT
S1	S2	LED6
0	0	1
0	1	0
1	0	0
1	1	0

Table 2 NOR Circuit

PROJECT 151. C-MOS XOR GATE



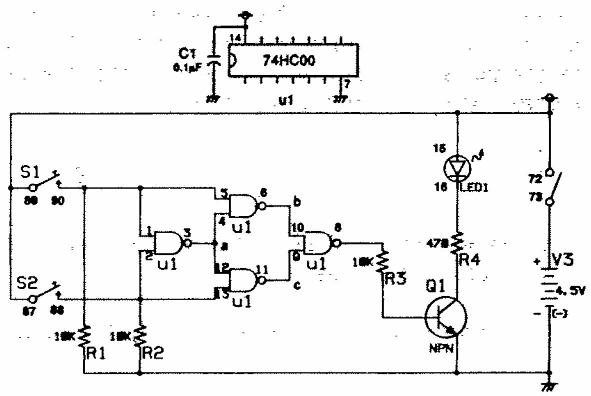
U1	74HC00	R1	10KΩ	R3	10KΩ	C1	0.1μF
Q1	NPN	R2	10KΩ	R4	470Ω		

Back in project 133 you saw how an XOR gate works. Since we've made up other digital circuits by combining NAND gates, you might suspect we could make XOR gates as well. We can, as this project shows.

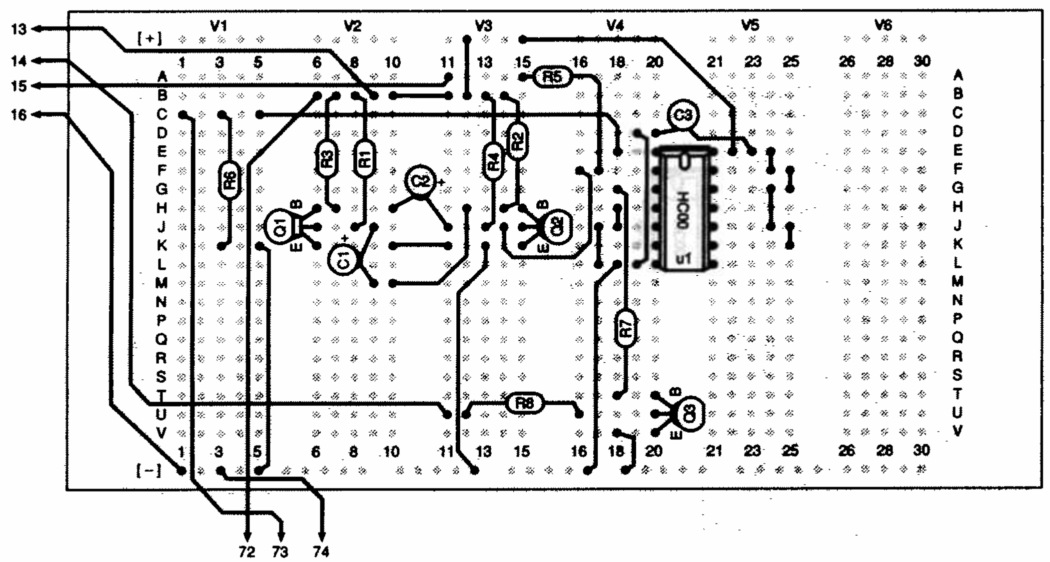
After you finish the wiring connections, turn power ON and press **S1** - does anything happen to **LED 1** ? Now release **S1** and press **S2**. What does **LED 1** do now? While keep pressing **S2** press **S1**. What happens to **LED 1**?

You can see this circuit behaves like the XOR gate you built back in project 133. The output is 1 as long as the inputs are different. But if both inputs are the same - either 0 or 1 - the output of the gate is 0.

Put on your thinking cap and try to follow each 0 or 1 input through the circuit until you reach the output. It will help if you mark "0" or "1" on the schematic at the input and output of each NAND gate.



PROJECT 152. C-MOS NAND ENABLE CIRCUIT



U1	74HC00	Q3	NPN	R3	22KΩ	R6	10KΩ	C1	100μF
Q1	NPN	R1	1KΩ	R4	100Ω	R7	10KΩ	C2	100μF
Q2	NPN	R2	10KΩ	R5	470Ω	R8	470Ω	C3	0.1μF

NAND gates can also be "electronic sentries." If you don't want a signal to pass, a NAND gate can make sure it doesn't.

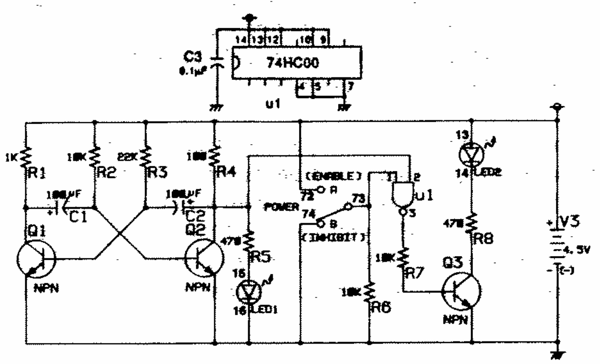
We call this project a "NAND enable circuit" because that what the NAND gate does - it enables signals to pass through a channel. The two **LEDs** let you see whether or not the signal shown at **LED 1** is allowed to pass to **LED 2**.

You probably recognized one circuit in the schematic right away - the multivibrator. You can see the output of the multivibrator by watching **LED 1**. You'll also notice that the multivibrator provides one of the inputs to the NAND gate. What do you suppose happens when the **select switch** is up or down? Be sure to make some mental notes because we're about to find out.

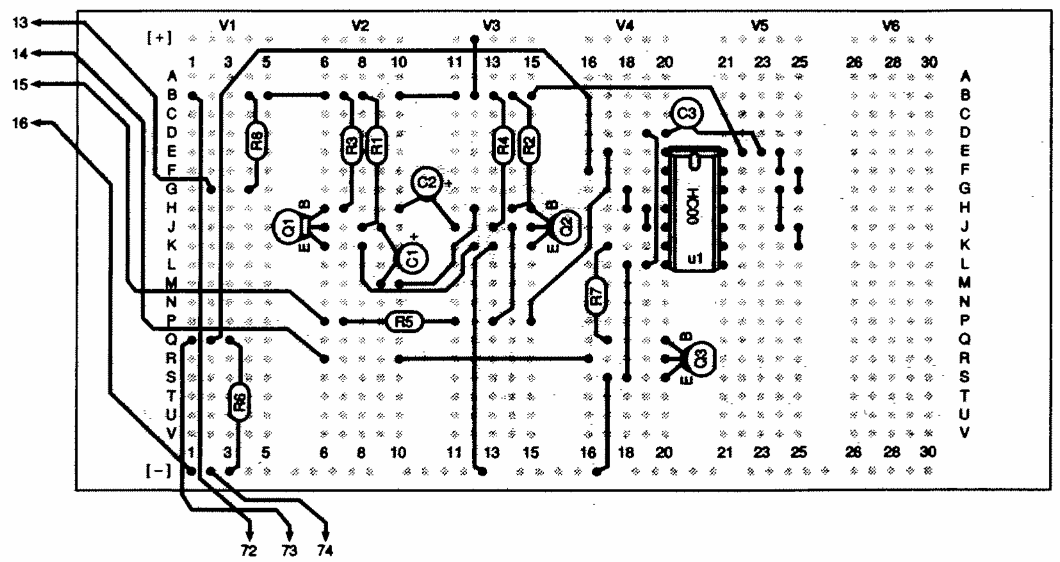
As you build this circuit, set the **select switch** down, and look at **LEDs 1** and **2**. You'll see **LED 1** "blink" to indicate the output of the multivibrator. But look at **LED 2**. You'll see that it stays lit all the time, indicating that something's preventing the signal at **LED 1** from reaching **LED 2**. Now set the **select switch** up and observe **LED 2**. What is happening? Is the same thing happening to both **LED 1** and **LED 2**?

You can see that **LED 1** and **LED 2** "take turns" going on and off. This is because we set one input of the NAND to 1 when we set the **select switch** up. The multivibrator sends 0 and 1 signals to the other input. When the signal is 1, **LED 1** lights but both input signals to the NAND are then 1. That means the NAND's output is 0, and **LED 2** goes out. Then one of the inputs to the NAND becomes 0, its output goes to 1 and **LED 2** lights.

(Now be honest - did you figure all that out before building the circuit?? We hope so.)



PROJECT 153. C-MOS AND ENABLE CIRCUIT



U1	74HC00	Q3	NPN	R3	22KΩ	R6	10KΩ	C1	100μF
Q1	NPN	R1	1KΩ	R4	100Ω	R7	10KΩ	C2	100μF
Q2	NPN	R2	10KΩ	R5	470Ω	R8	470Ω	C3	0.1μF

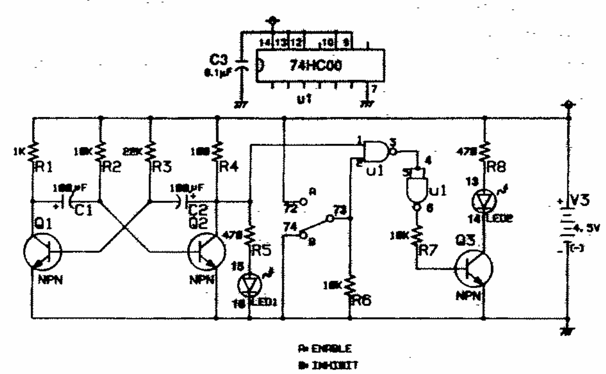
Our last project had a characteristic that might be a problem in some situations. **LEDs 1** and **2** take turns lighting on and going off. We might want both **LEDs** to light on and off together. Did you figure out how to do this when you played with the last project? If not, this circuit shows you how.

If you look carefully at the schematic for this project and the schematic for the last project, you'll notice that they're almost identical. The only change is the addition of another NAND gate... can you guess what effect this has on the operation of the circuit?

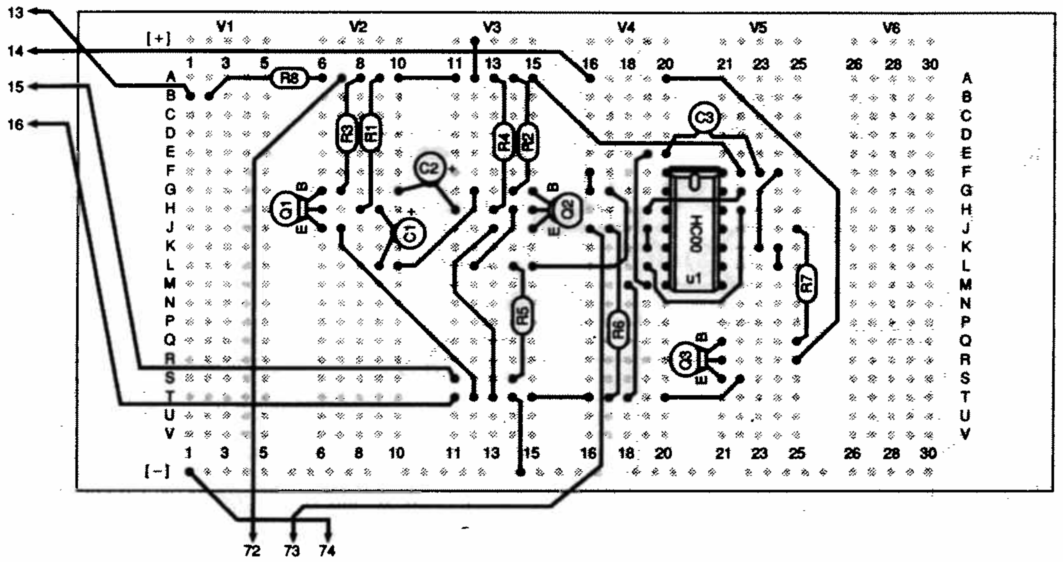
Just like our last project, setting the **select switch** down blocks the channel from **LED 1** to **LED 2**. But when you set the **select switch** up, you'll see **LED 2** light and go off together with **LED 1**. The two NAND gates together make up an AND gate (remember from project 143).

In a circuit like this, **LED 1** is often referred to as the data input. **LED 2** is often called the output. These terms are often used with enable circuits and pop up from time to time when we talk about digital electronics.

You might suspect by now that we can use other digital circuits to perform an enable function. Can you figure out how? Be sure to keep notes of what you figure out... especially if you figure out how to use an OR gate in an enable circuit. (There's a reason why, as you'll discover in the next project.)



PROJECT 154. C-MOS OR ENABLE CIRCUIT



U1	74HC00	Q3	NPN	R3	22KΩ	R6	10KΩ	C1	100μF
Q1	NPN	R1	1KΩ	R4	100Ω	R7	10KΩ	C2	100μF
Q2	NPN	R2	10KΩ	R5	470Ω	R8	470Ω	C3	0.1μF

Could you figure out how to make up an enable circuit using an OR gate? If you could, here's a chance to check your ideas against an OR enable circuit we've cooked up.

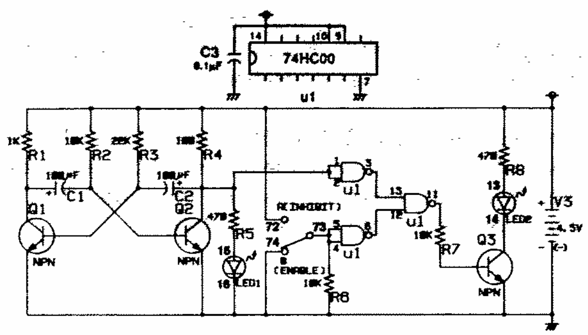
Like our last two projects a multivibrator "feeds" input to the OR gate. You can see the output of the OR gate when you look at **LED 1** - it flashes on and off according to the output of the multivibrator. Can you tell what happens once the multivibrator's input is applied to the OR gate by looking at the schematic? Give it a shot before building the project.

As you build this circuit, try setting the **select switch** up instead of down like we did for the last two projects. What does **LED 1** do? And what is **LED 2** doing? Now set the **select switch** down. What happens to **LED 1** and **LED 2** now?

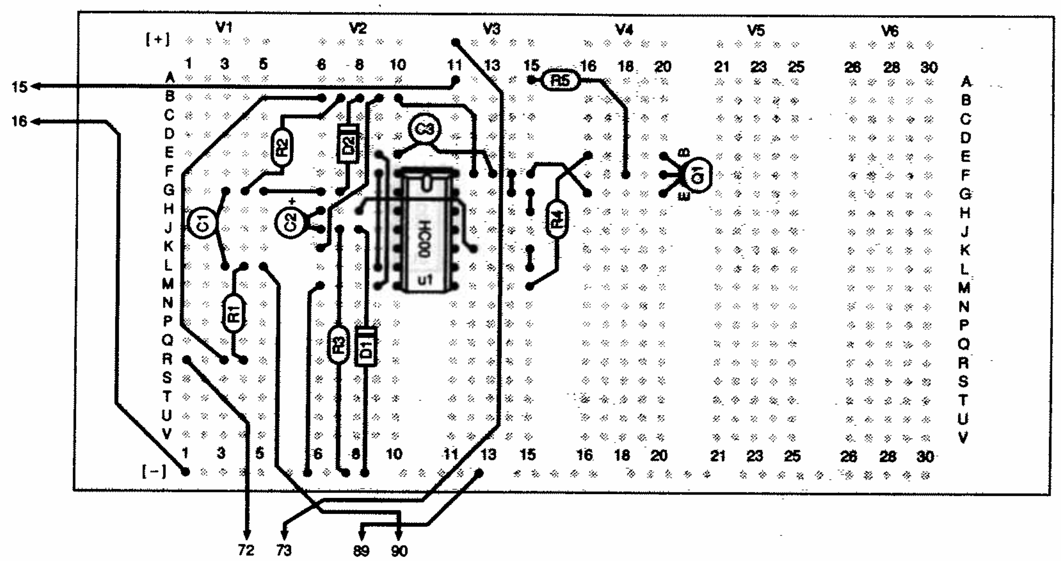
You say that in this circuit setting the **select switch** up blocks the flow of data from **LED 1** to **LED 2** (this is called the inhibit status). But when the **select switch** is down, data can flow from **LED 1** to **LED 2**. This is called the enable status.

This circuit works like it does because a NAND gate can give an output of 1 only if both inputs are 0. When you set the **select switch** up, the inputs to one NAND are at 1. The output is 0, and this supplies one of the inputs of the NAND gate controlling **LED 2**. This 0 input never changes as long as the **select switch** is left at the up position. But when the **select switch** is set to the down position, both inputs become 0 and the output becomes 1. This means one of the inputs to the NAND controlling **LED 1** becomes 1 all the time, and **LED 2** can now blink on and off depending on whether the other input is 0 or 1.

(Now there was nothing difficult about that, was there?)



PROJECT 155. A ONE-SHOT NAND GATE



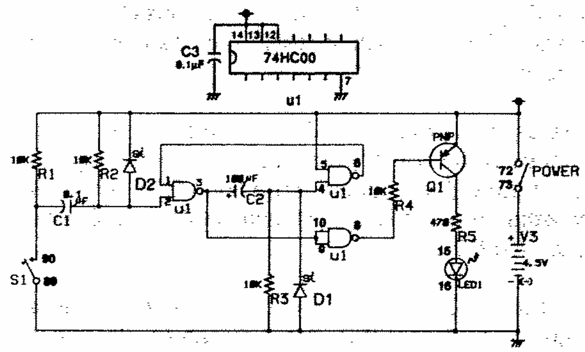
U1	74HC00	R1	10KΩ	R4	10KΩ	C1	0.1μF	D1	Si
Q1	PNP	R2	10KΩ	R5	470Ω	C2	100μF	D2	Si
		R3	10KΩ			C3	0.1μF		

Does the term "one-shot" mean anything to you? (No, it's not a nickname for some cowboy ... or a gun that holds just one bullet!) If you're stumped, take a look back at project 30.

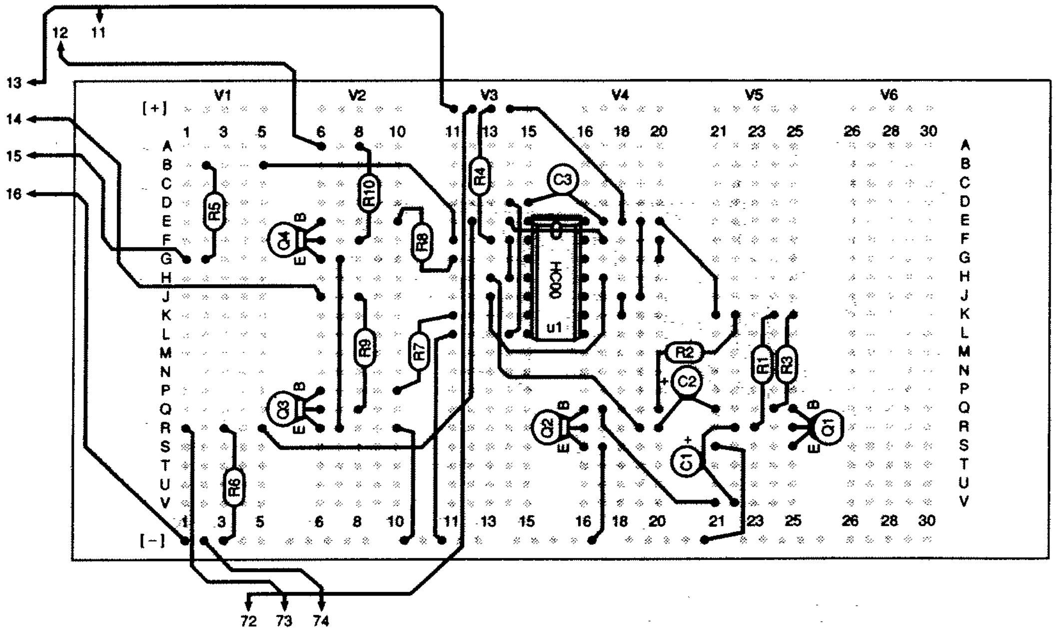
When you finish the wiring connections, turn power ON. Press **S1** once and watch what happens to **LED 1**. Try holding **S1** down for different periods of time ... does **LED 1** stay on the same length of time or does it vary?

You see that a one-shot multivibrator has an output for a certain length of time regardless of the length of the input (it "fires one shot" !). This means that it can be used in many circuits as a timer. You might also see this circuit called a monostable multivibrator.

Since this is a multivibrator, you might suspect that there's some way to vary the time it produces an output. You're right - there is a way - and we'll let you try to discover what it is. (Actually, you shouldn't have much trouble discovering which parts you need to change. Be sure to make notes about the effect of higher and lower values on circuit operation.)

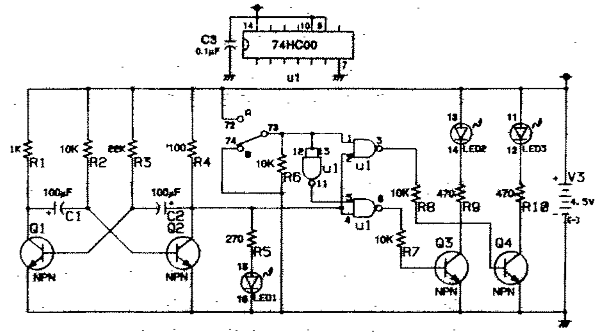


PROJECT 156. C-MOS LINE SELECTOR



U1	74HC00	Q4	NPN	R4	100Ω	R8	10KΩ	C1	100µF
Q1	NPN	R1	1KΩ	R5	270Ω	R9	470Ω	C2	100µF
Q2	NPN	R2	10KΩ	R6	10KΩ	R10	470Ω	C3	0.1µF
Q3	NPN	R3	22KΩ	R7	10KΩ				

It isn't too hard to think of situations where we might want to send input data to two or more different outputs. This project shows how we can use a network of NAND gates to help us do just that.

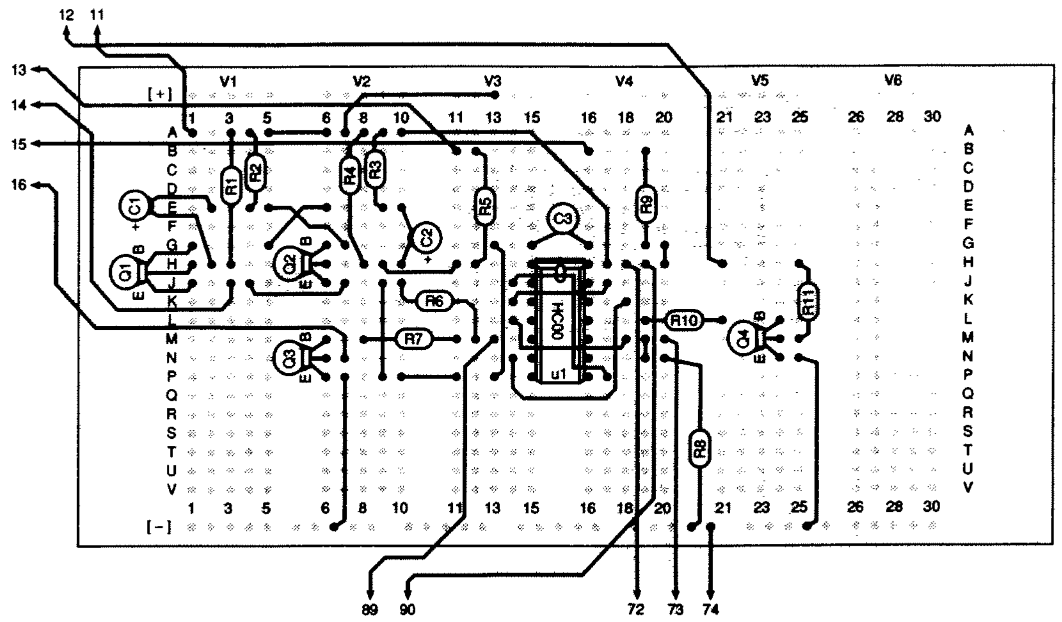


You can see that we use a multivibrator and three NAND gates in this circuit. You can leave the **select switch** at either up or down position when building this project. You'll see that **LED 1** is blinking. But what are **LED 2** and **LED 3** doing? If the **select switch** is at the down position, **LED 2** is blinking; if it is up, **LED 3** blinks.

As you can see on the schematic, setting the **select switch** up or down controls the inputs to the two NANDs that light **LED 2** or **LED 3**. With the **select switch** at the down position, the NAND controlling **LED 2** gets one steady input of 1. The output of the multivibrator supplies the other input. As the multivibrator's output switches from 0 to 1, the NAND controlling **LED 2** also switches its output from 0 to 1.

The opposite happens when you set the **select switch** up. Now the NAND controlling **LED 3** gets a steady input of 1 so that **LED 3** can go on and off according to the input from the multivibrator.

PROJECT 157. C-MOS DATA SELECTOR



U1	74HC00	Q4	NPN	R4	100Ω	R8	10KΩ	C1	100µF
Q1	NPN	R1	1KΩ	R5	470Ω	R9	470Ω	C2	100µF
Q2	NPN	R2	10KΩ	R6	10KΩ	R10	10KΩ	C3	0.1µF
Q3	NPN	R3	22KΩ	R7	10KΩ	R11	470Ω		

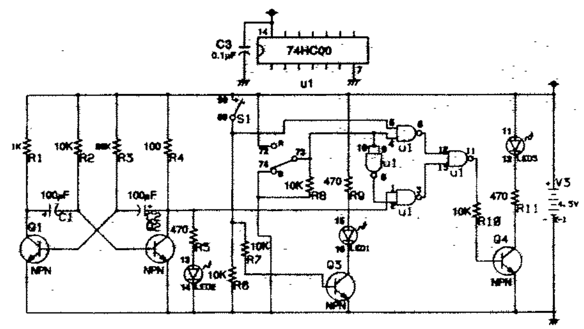
In our last project you saw how data could be sent to two or more different outputs. You can probably think of some situations where we might want to (or need to) do the opposite - send data from two or more different sources to one output. This circuit lets us see how this can be done.

When you look at the schematic for this project, you'll notice two different input sources. One is provided by the multivibrator circuit when the **select switch** is set to the down position. But when the **select switch** is set to the up position, the input signal is provided by ..... can you guess? (And don't peek at the answer.)

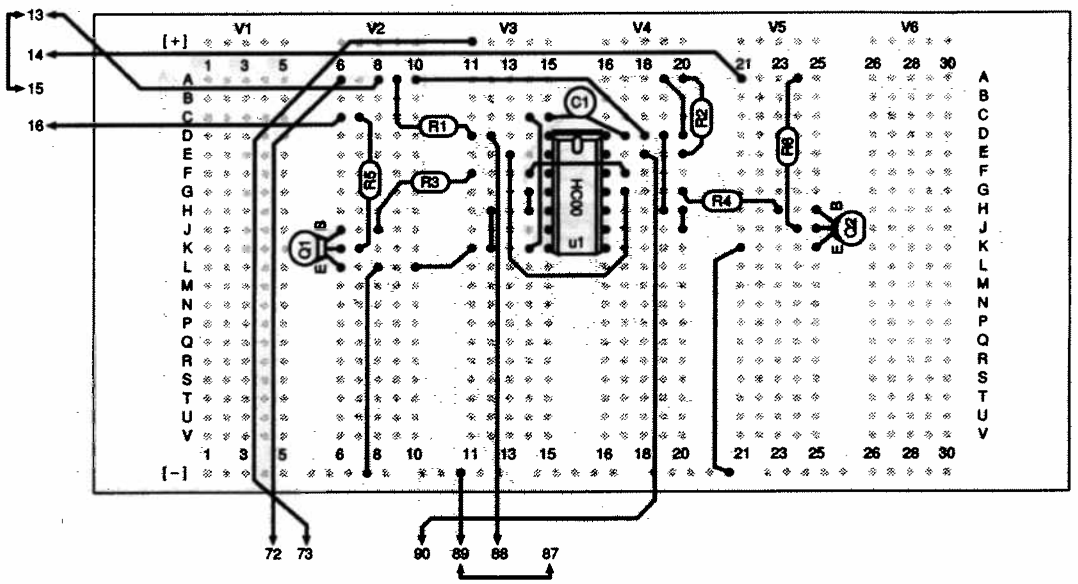
Yourself! You provide the input signal by pressing and releasing **S1**.

To see how this circuit works, set the **select switch** up when you build this project and watch both **LED 1** and **LED 3**. Is anything happening? Now press **S1** and see what happens to **LED 1** and **LED 3**. Does **LED 3** go on in step with **LED 1**? Now set the **select switch** down. **LED 3** start to go on and off in step with **LED 2**. Try pressing **S1** now - does it have any effect on **LED 3**? What does **LED 3** do when you press **S1** while the **selector switch** is down?

More complex versions of these circuits are found in computers and other highly advanced digital circuits. And - as you probably suspect - switching from one input channel to another is done electronically in most cases.



PROJECT 158. C-MOS R-S FLIP FLOP



U1	74HC00	R1	10KΩ	R4	10KΩ	C1	0.1μF
Q1	NPN	R2	10KΩ	R5	470Ω		
Q2	NPN	R3	10KΩ	R6	470Ω		

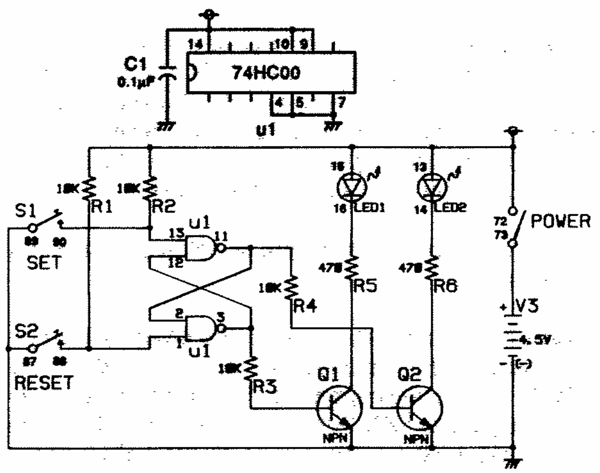
Even though your kit includes a J-K Dual Flip-Flop IC, it's possible to build an R-S flip-flop using NAND gates. This project proves it! (If your memory needs refreshing about how an R-S flip-flop works, take a look back at project 31).

After you finish the wiring connections, turn power ON. You will see either **LED 1** or **LED 2** light up. Press **S1** and **S2**. What happens to **LED 1** and **LED 2**? Can you guess which represents the set state and which represents the reset state?

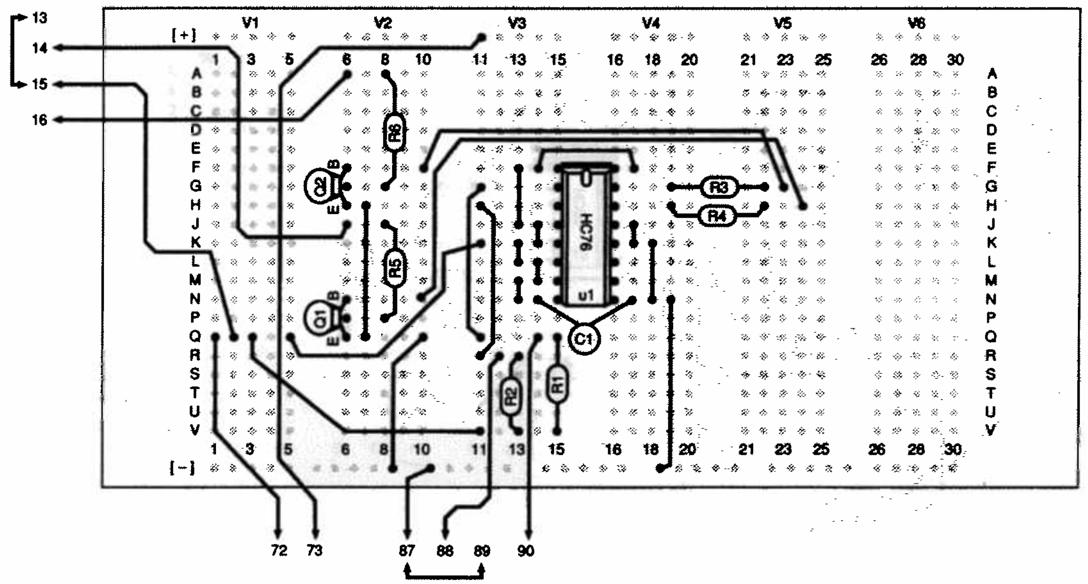
When **LED 2** lights up, the R-S flip-flop is in the set state. When **LED 1** lights up, the R-S flip-flop is reset. When you get the flip-flop set or reset, try pressing the key and see what happens.

You see one of the prime characteristics of the R-S flip-flop - once the circuit is set or reset, the circuit keeps that state until an input signal causes it to change. This means the R-S flip-flop can "remember" things. More advanced circuits like this one are used in computers so they can "remember" things as well.

We'll tell you one thing. Try to press **S1** and **S2** at the same time. What happened to LEDs? This is a weak point of the R-S flip-flop in a sense. So you should pay attention to this operation when you design a circuit using R-S flip-flops.



PROJECT 159. C-MOS R-S FLIP-FLOP II



U1	74HC76	R1	10KΩ	R4	10KΩ	C1	0.1μF
Q1	NPN	R2	10KΩ	R5	470Ω		
Q2	NPN	R3	10KΩ	R6	470Ω		

One of the integrated circuits in your kit is known as a "Dual J-K Flip Flop." Inside of that tiny IC are two J-K flip-flop circuits, like the one you built back project 31! ICs are true wonders, letting us use several circuits in a small space.

Here's a project that lets you see how to use this particular one.

When you finish wiring this project, turn power ON. Press **S1** and release it. Does anything happens to **LED 1** or **LED 2**? Now press **S2** and release it. Is there any change in **LED 1** or **LED 2**? Can you figure out which terminal is the set input and which is the reset input? (Okay, here's a hint... do you remember what Q and Q̄ stand for?)

This shouldn't have been too hard for you... IC terminal 2 is the set terminal while 3 is the reset terminal.

You can see by the schematic that R and S aren't the only inputs to the flip-flop - there's also a J and K input. (So that's why it's called a J-K flip-flop!) What do you think these J and K inputs are used for? Make some notes about what you think... because we're soon going to find out.