Bill Cheek
bill.cheek@f731.n202.z1.fidonet.org

Digital Logic Simplified

I am painfully aware of the intense fear some of you dear readers may have for things digital. I myself was a confirmed holdout against the digital transition. I still have boxes and bins full of old tubes, 2N404 transistors and selenium rectifiers. Neverthe-less, I am a convert to the world of digital technology. I have to be, if I want to squeeze the most performance out of the new receivers coming out today.

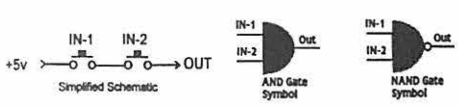
We will take it slow and easy, though, with a little something thrown in each month for those of you who haven't made up their minds about the world of 1's and 0's.

You digital gurus will have to bear with me for a couple of months while I introduce the fundamental circuits of digital technology to those who are just waking up to it. Sorry, it can't be helped. Digital novices should sit up and pay attention because we're going to have some fun as we learn. I will skip as much "theory" as possible and by next month, we'll be into hard core, hands-on, down and dirty circuit building.

Digital technology offers cheap and effec-tive ways to inject raw horsepower into our monitoring stations. In most cases, digital IC's are priced at around a buck or less. What's more, the necessary peripheral com-ponents for each IC usually amount to no more than a few cents. In many cases, the hobbyist can concoct a fairly sophisticated digital cir-cuit for well under $5.00 and reasonably ex-pect it to work the first time out. Now let's get to the groundwork; fun stuff just ahead!

The fundamental building block IC's of digital electronics include: AND, OR, and BUFFER circuits followed by their logical opposites: NAND, NOR, and INVERTER circuits. Add to this list, the EXCLUSIVE OR (XOR) and EXCLUSIVE NOR (XNOR) cir-cuits to round out the first lot. There are a few other basic logic circuits but those listed here will do for now. Figure 1 lists some examples of each IC with pinouts for the 74HCxx series. Symbols and simplified schematics are given in the following introduction of the more common logic circuits: AND, NAND, OR, NOR, and buffers/inverters.

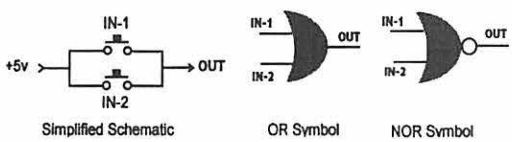
AND and NAND Gates



The AND circuit is easiest to comprehend and be understood as two switches in series, such that Switch 1 AND Switch 2 have to both be closed in order to get an output. Closing just one switch won't do. That's why this circuit is called an AND gate. Suppose you had a heat sensor on one input and a smoke sensor on the other. The AND gate would not output to an alarm until BOTH conditions existed. This "intelligent" logic helps prevent false alarms.

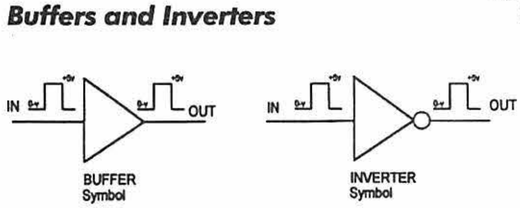
The opposite logic circuit of the AND gate is called a NAND (Not AND). Don't get hung up on the NOT term here; it is only a Boolean expression that means logical opposite of what is to follow. A NAND isn't as easily represented with switches as the AND gate, but is understandable described as an inverted AND function. If both switches are closed, the output is low, but if either switch is open, then the output is high.

OR and NOR Gates



Next, we have the OR gate. Either switch closed (or both: it doesn't matter), will pro-duce an output high. An OR gate could be used to trigger an alarm if either smoke OR heat existed, see? The opposite logic of the OR is the NOR and like the NAND, it is more difficult to depict than to describe. Neither an input on 1 nor an input on 2 are required to get an output high. Conversely, there is no output high if there is an input high on either 1 or 2 or both.

Buffers and Inverters



BUFFERS and INVERTERS are the sim-plest of all. First, both are properly called "buffers" with specific terms to identify each type: non-inverting buffers and inverting buff-ers. I shall refer to them as "buffers" and "inverters," however. Both of these circuits completely isolate the output side from the input side. Buffers provide only isolation while inverters first isolate and then invert the input to its opposite state in the output. Logic inversion turns 1's into 0's and 0's into 1's. There are instances of non-isolating inverters, but we'll not use them in our projects.

Applications

Now, let's look at some real world applica-tions of AND and OR logic. The simplest AND circuit exists in your home with Switch 1 as one of the circuit breakers in the AC service entrance and another as the switch on your radio. Both the circuit breaker AND the On/Off switch on your radio must be ON before the radio will work.

Simple OR logic exists in most two-story homes where there is a light switch at the top of the stairs and another at the bottom, and where either switch being on will satisfy the OR wiring to light the bulb.

AND and OR logic are not complicated....not in the real world and not in electronic logic circuits. On the other hand, NAND and NOR logic can be confusing because it is hard to draw a picture of how they work. This is a case where a few words are worth more than a hundred pictures. The best way to understand NOT logic is to first view the function in its positive logic state; (analyze it in terms of AND and/or OR) and then invert the output! For example, a NAND gate is only an AND gate with an inverted output. Same with a NOR gate being only an inverted OR, you see. Think of NOT logic as in the case where an indicator LED signifies trouble (oil pressure idiot light in your car) as opposed to another light indicating normal operation (pilot lamps).

Spend a little time with Fig-1 to come up to speed for our next several projects. You'll notice some peculiar pin markings in Fig-1, but relax, because all these chips offer the economy of several identical circuits on a single chip. One or more sections can be used in any project. For example, look at the pinout for the 74HC08A NAND gate. NAND "A" consists of two inputs at pins 1-2 with an output at pin 3. NAND "B" has two inputs at pins 4-5 with an output at pin 6. Ground is at pin 7 followed

FIGURE 1: *Common CMOS Logic Chips*

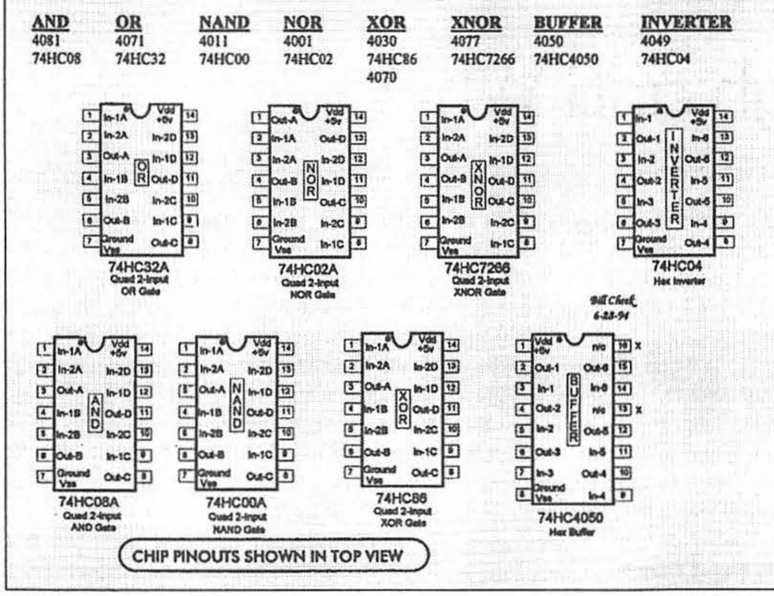
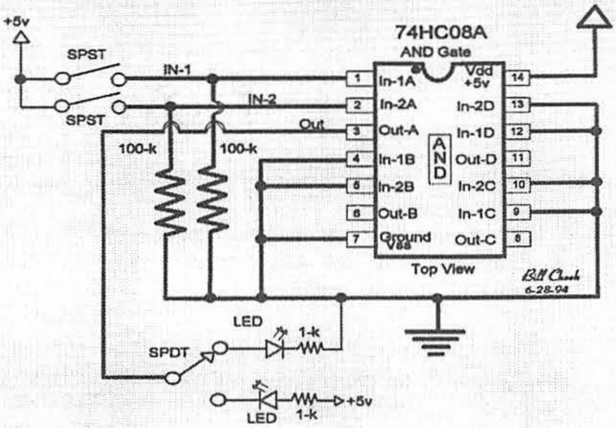


FIGURE 2: *Trainer Circuit for -AND- Logic*



Note: Bold traces are the required AND circuit. Light traces are the optional control interface. The SPST switches can be a simple 2-seg DIP switch. The SPDT toggle switch and LEDs demonstrate the two different ways to derive logical outputs for this type of AND gate. Eliminate the light-trace circuits and add your own In/Out requirements as desired.

by two inputs for NAND "C" at pins 9-10 with an output at pin 8. NAND "D" inputs are at pins 12-13 with an output at Pin 11. +5v feeds Pin 14.

None of these four NAND gates interact with each other. As with all CMOS circuits, the INPUT pins of all unused sections must be grounded to prevent oscillations. For example, in Fig-2, a test circuit for the AND gate, we'll use section 1 only, so you'll see that the other inputs—Pins 4-5, 9-10, and 12-13—will be grounded. Unused outputs should be left alone, unconnected.

Closing Notes:

Two different logic families are shown in Fig-1: the 4000 series and the 74HCxx series. As a rule of thumb, logic families are not to be mixed, though we can sometimes mix these particular two families without adverse consequences. You should avoid the mixture where possible, though. The 4000 series IC's draw more current; operate slower; and are best suited for +8v and even +15v logic though they will work at +5v. The 74HCxx series are limited to +5v logic, but are faster with extremely low current requirements. The 74HCxx series will be the logic family of choice for most of our forthcoming projects.

Use the 4000 series only if you must, and then be sure to have a data sheet on each one, because the pinouts can differ from the 74HCxx series. In all cases, standard CMOS handling precautions should be exercised with these two logic families. Static charges can zap them. You probably can't go wrong with laying in a small supply of 74HC00A, 74HC08A, 74HC4050, and 74HC04 for future projects. Also stock a supply of LEDs, 0.1-µF capacitors, and resistors in the range of 680-□ to 100-k□. It won't hurt to peruse my recent columns in back issues of *MT* and build, buy or have available a +5v regulated power supply for bench and testing needs.

By the way, for the most timely and expedient technical support for projects in this column, you can always reach me direct at my BBS, the Hertzian Intercept, (619) 578-9247, after 5:30pm and before 1:30pm, weekdays, PDT; 24-hrs, weekends. Replies usually posted by noon the following day. I am available by E-Mail over the Internet at the address above.

A Hidden Feature of the Sony ICF-2010?

Gerald S. Busch of Coderre, Saskatchewan, Canada, has an interesting shortwave listening tip to share with us this month. Gerald says owners of the Sony ICF-2010 receiver may be disappointed by the audio quality of the USB and LSB modes. The narrow SSB filter can make signals sound very muddy. There is a workaround, probably unintended by the set's designers, since it involves use of the apparent AM-only function. The owner's manual makes no mention of the technique, nor does it explain single-sideband reception.

Gerald says you can clarify those muddy sideband signals by using the synchronous detector! First, tune a sideband station in the usual manner. As soon as you have adjusted the frequency to bring the voice to normal pitch, switch on the "SYNC" mode. You may have to toggle back and forth between this and the SSB mode a couple of times until the circuitry locks in properly and the voice becomes intelligible. Once it does, you are still using the narrow filter, but you now have the option of switching to the wide filter, which was not available in sideband mode.

If adjacent frequency interference is not a problem, reception with the wide filter yields clearer and natural sounding audio, for a more pleasant monitoring session!