UPDATE: *Additional information to previous articles*

The X-10 TW523 Two-Way Power Line Interface

A Step Toward Closed-Loop Power Line Control

by Ken Davidson

Just as I promised in Vol. 1, No. 3 of INK, I have an update on the status of an X-10 receiver module. Before I begin, though, let me fill in some information for those of you who may have missed the last article.

The X- 10 POWERHOUSE system is based on a concept known as carrier current communication. The AC wiring already strung throughout the house or building is used to carry control signals from command consoles to remote switch modules. These modules contain either a solenoid for switching heavy loads, or a dimmer circuit used to dim, brighten, and turn lights on and off. Each module can be set for any of 256 unique addresses, so a very elaborate control network can be set up (a typical control console can only address 8 or 16 modules, however).

In my previous article, I described a new module available from X- 10 (USA) Inc., the PL513, that allows a computer to directly access the AC power line. Using the X-10 transmission protocol, the computer then controls the timing and configuration of bits transmitted through the power line.

One of the biggest drawbacks of the X-10 system over the years has been its "open-loop" configuration and lack of two-way communication. There are quite a few devices around capable of transmitting X- 10 codes and plenty of receiver modules that provide control actions as a result. However, until now, there has been no convenient way to intelligently "listen" to the power line and "hear" all the X- 10 activity taking place.

With the introduction of the X- 10 POWERHOUSE TW523 Two-Way Power Line Interface, all that is changing. The TW523 not only contains the complete transmitter circuitry found in the PL5 13 module, but also contains all the necessary front-end circuitry to receive X- 10 codes as well.

Figure 1 shows the schematic for the receiver section of the

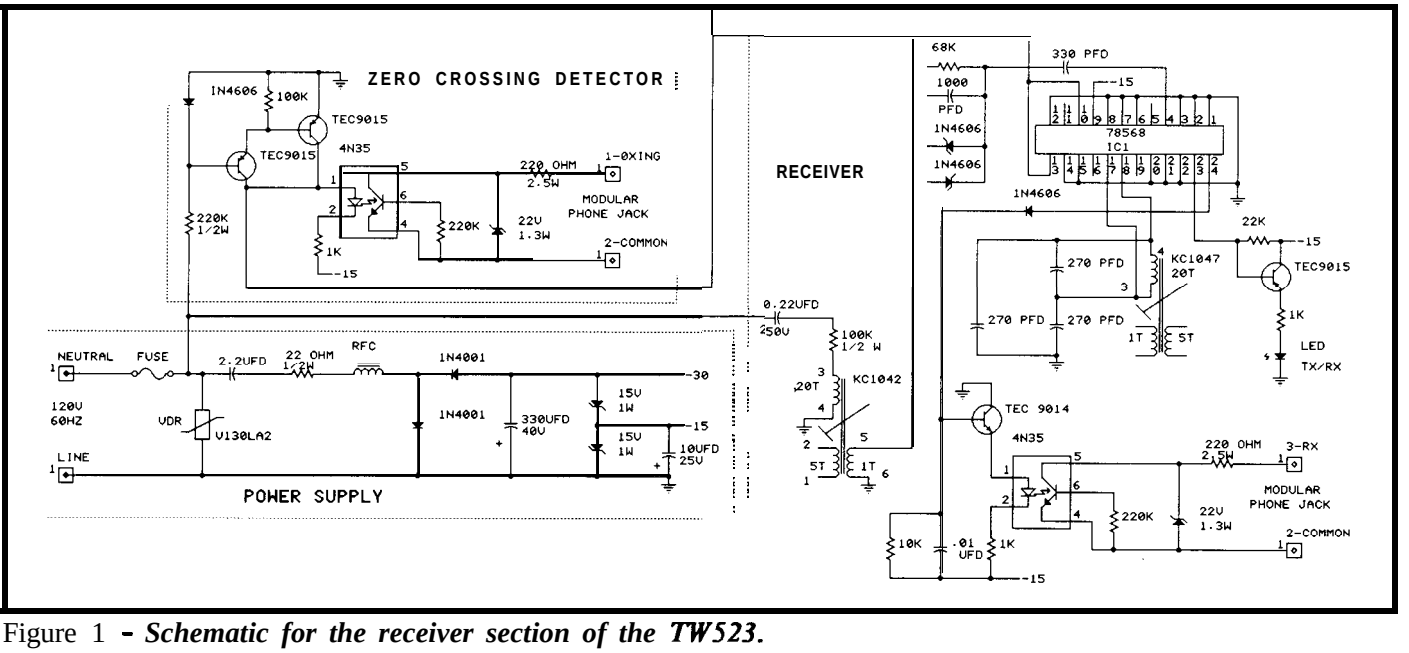

Figure 1 - ***Schematic for the receiver section of the TW523.***

TW523. Since the schematic for the transmitter section is nearly identical to that of the PL513 (which was published in the last article), I haven't reproduced it here.

As I described in detail in the last article, the X-10 protocol is made up of a start code, a house code, and a function code (which may be a module number or an actual function). One bit of the code is always sent out once immediately after the AC zero crossing and two more times to correspond with the zero crossings of two other AC phases. A "1" bit consists of a I-ms burst of a 120-kHz oscillator and a "0" bit is denoted by the lack of a burst. Except for the start code, a bit is followed on the next zero crossing by its complement. Finally, each start/house/function group is sent out twice with no delay between them. The top half of figures 2a and 2b show what a typical transmission looks like.

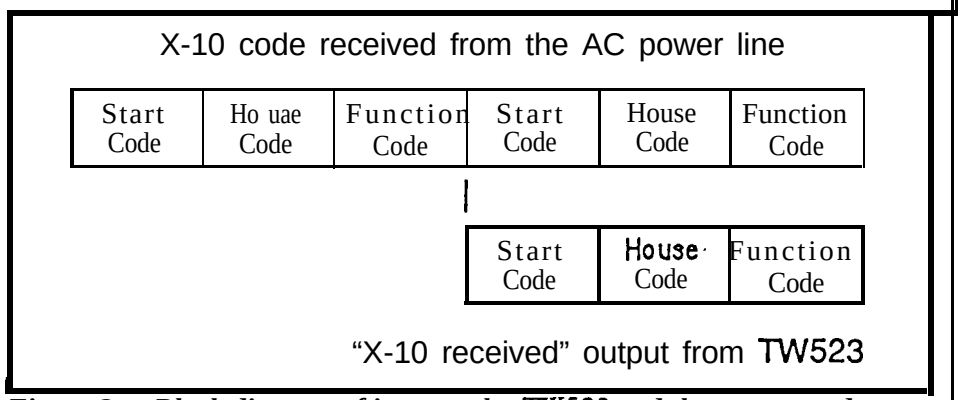

Figure 2a - Block diagram of input to the TW523 and the corresponding output.

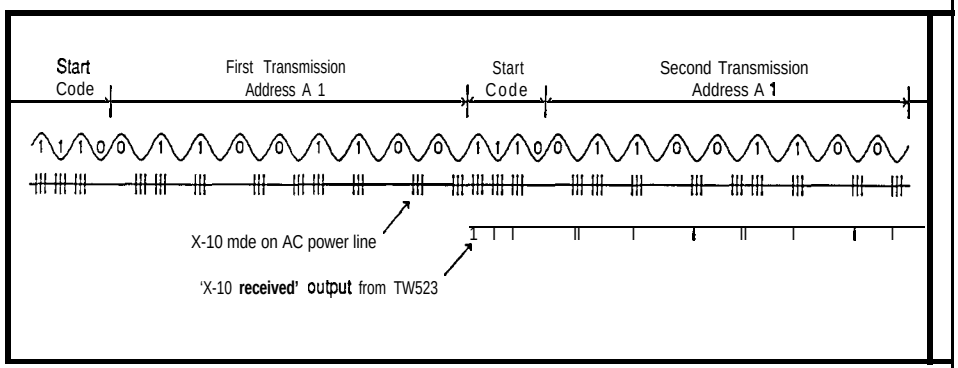

Figure 2b - ***The TW523 outputs a single I-ms pulse for every group of three 120-kHz pulses it receives.***

Imagine, if you will, a very crude receiver front end. It consists of a zero crossing detector, a filter which passes just 120 kHz, and some isolation circuitry. It is now up to the programmer to listen to the power line at the correct times and decide if a 120-kHz burst is present, if the length of the burst is proper, if the start code is good, if there is an error in the house code or function code, and so on. A great deal of time must be spent in error checking.

The TW523, however, adds some intelligence to the receiver front end. It takes care of listening for proper-length 120-kHz bursts and valid X-10 code. If it determines that what it hears isn't valid X-10 code, it won't act on it. When it does detect valid X-10 code, it presents the programmer with clean I-ms pulses coincident with the AC zero crossings. Don't expect the module to be smart enough to assemble groups of complete code transmissions, though. Something has to be left for the programmer to do.

Figures 2a & 2b show how the receiver output corresponds to the activity taking place on the power line. The TW523 listens to and validates the first occurrence of any X-10 transmission. If it determines the transmission to be error free, the receiver output will exactly match the second transmission of the code sequence. In figure 2b, each vertical "hash mark" on the transmission line represents a 1-ms, 120-kHz burst. Notice that each burst is repeated three times, as explained before. On the receiver line, each vertical hash mark represents a 1-ms pulse sent to the computer.

Since the transmitter software primitives are already written, the software to support the receiver falls into place rather quickly. We first wait for a zero crossing to occur, then wait 500 microseconds (half the duration of the l-millisecond pulse), and sample the receiver input line. That way we can be sure of sampling the center of the pulse. If it is high (no activity), we just sit back and wait for the next zero crossing. If it is low, we know a transmission is starting to come in, so we enter something of a state machine to assemble the rest of the received transmission.

Just briefly, a state machine is a hardware or software device which sits in a known state, able to move on to the next state based on one or more inputs. In the case of the receiver code, we know that we must first see a valid start code before we can move on. We can then assemble a house code, followed by a function code. If at any point, a single error is detected, we

	D1	D2	D4	D8	D16
1	0		110		0
2	1	1	1	0	0
3	0		0	10	0
4	10		10		0
5	0		0	0	10
6	10		0		10
7	0		10		10
8	1	1	0		1 0
9	0		1	1	1 0
l o		1	1	1	1 0
11	0		0		110
12	1	0		1	1 0
13	0	0	0	0	0
14	10		0	0	0
15	010		0		0

16	110	0		0
All Units Off	0	0	0	01
All Lights On	0	0		011
On	0	0	1 0	1
Off	0	0	1 1	1
Dim	0	1 0		0 1
Bright	0	10		11
All Lights Off	0	110		1
Extended Code	0	1	1 1	1
Hail Request	10	0		01
Hail Acknowledge	10	0		11
Preset Dim	1 0		1 x	1
Extended Data	110	0		1
Status=On	1	1 0	1	1
Status=Off	1	1 1 0		1
Status Request	1	1 1	1	1

Figure 3 - A *list of the existing twenty-two X-10 function codes plus ten newly defined codes. (x can be replaced with 0 or 1)*

throw in the towel and wait for the line to be quiet for at least three full line cycles before listening for valid code again. Since the front-end circuitry prechecks the incoming code, the chances of the host computer detecting an error are greatly reduced.

I'm not listing any software with this article, but I've posted on the Circuit Cellar BBS the HD64180 (280) code I wrote for the BCC180 to support the receiver section of the TW523 if anyone is interested. Support for IBM PC and BCC52 will be available when the TW523 module is actually in production.

X-10 Protocol Additions

An unexpected bonus found in the TW523 data sheet includes 10 new function codes designed to extend the usefulness of the X-10 system. I've listed these new codes along with the existing ones in Figure 3.

The more pedestrian additions are those meant to extend existing functions. "All Lights Off" (as opposed to "All *Units* Off") is one I've longed for in the past. With Preset Dim, the most-significant bit of the level replaces the "x" in the binary function code and the four least-significant bits of the level are placed where the house code is normally found. With a single code transmission, the target module can be told to what level to dim. Currently, the power line is tied up while the transmitter sends multiple dim commands.

More exciting are the completely new functions which rely, for the most part, on the existence of two-way modules. Hail Request is sent to determine whether any other transmitters are within listening range. Any such transmitters would send out a Hail Acknowledge in response.

Extended Data is sent to signal that additional 8-bit data follows. Presumably such data would be from temperature sensors or perhaps alarm contacts. Extended Code can be used to continue making additions to the X-10 command set even though all the primary command codes are used (similar to the way Zilog extended the 8080 instruction set for the 280).

All of the above additions are still only on paper, though. Computers using TW523 modules can implement most of the functions, but existing modules won't respond to, say, "All Lights Off." The last three added codes, however, have been implemented in at least one module already on the market.

When a module is selected and a Status Request is sent to it, the module is supposed to respond with either Status = On or Status = Off. The X-10 POWERHOUSE Radio Controlled System's transceiver module (RR50 1) already supports the status function. The RR501 receives codes sent via R.F. from the hand-held RT504 radio transmitter and retransmits the codes over the power lines. It, like the TW523, is a true two-way module. I've set up the TW523 to listen and have seen the RR501 respond with its status when requested, so it really works. Now if the rest of the modules would only do the same...

The Future

What does the future hold for the X-10 line of products? If the latest releases by X- 10 are any indication, we are looking at some exciting times. With the introduction of the new function codes and (hopefully) their eventual implementation, X- 10 will finally develop into a true closed-loop, full-duplex system upon which even more powerful home control systems (and control systems in general) can be based.

The TW523 is an important first step in that direction. It will now be much easier for system designers to design two-way X- 10 operation into their products, perpetuating the line (you can be sure this is X-10's main motive for developing such interface modules).

As I've said before, we plan to do a lot more with these new power line interface modules. I'll keep you informed as anything new develops. ■

Special thanks to Dave Rye for his contributions to this article.