

X-10 Control Program for Windows (X10CTL) Ver. 2.6

**** NOTICE

This program is **available at no cost** from the Author, Larry E. Fosdick [CIS 70105,1142]. No warranty, either expressed or implied, covers losses of data, fingernails, or hair from the use or abuse of this software. This program may be freely distributed. No money is to be collected for this software, especially by the author, who did this for recreation. **The source code is NOT available.**

**** Changes from Version 2.4 & 2.3

- Bugs in sunrise/sunset routine resulting in "Subscript Out of Range Errors" fixed
- Uses SPIN.VBX, but renames it as SPINX10.VBX to avoid compatibility issues
- Code to test for correct .VBX dates removed, since it caused problems.
- Batch sequences can be executed from the startup command line (see Batch Mode)

**** Changes from Version 2.2

- Events can be offset from sunrise or sunset
- Comments can be added to Batch Command Sequences
- Unit descriptors can be used in the Immediate Mode
- Tests for correct .VBX files. Eliminated Gauge.VBX from program.
- Automatically stores current event list on disk

INTRODUCTION

This program, written in Microsoft Visual BASIC 2.0, provides complete access and control for the X-10 Powerhouse Interface (X-10 USA Inc.). Features include:

- True multiple window application
- X-10 clock can be automatically synchronized to computer system clock
- Console window for immediate control
- Descriptive unit definitions entered by user
- Descriptive unit definitions accessible from Immediate Console (new in 2.3)
- Timed events are easily added, deleted, or modified, with changes downloaded automatically to the interface
- Timed event sets can be saved to and restored from disk files
- Timed events can be set to
Today, Tomorrow, Everyday, Weekdays, Weekends, or Specific Days
- Event list can be printed for reference
- 16-position dimmer function for lamp units or light switches
- Powerful Batch facility added (new in Version 2)
- Comments can be added to Batch Sequences (new in 2.3)
- Events can be offset from sunrise or sunset (new in 2.3)
- Events are stored on disk for power-failure recovery (new in 2.3)
- .VBX files checked for correct versions (new in 2.3, removed in 2.4)

X10CTL requires the X-10 Powerhouse interface from X-10 (USA), Model CP290. It can be purchased from Radio Shack, CompUSA, Crutchfield, and probably others. I do not endorse any equipment vendor. This information is provided to answer queries Revision 1.00 generated. There may be other vendors. I bought my X-10 Powerhouse from DAK in 1986, and it works very well.

Sunset/Sunrise Offsets

Offered after popular requests, events can be scheduled as offsets to sunrise or sunset. Offsets can be positive (later) or negative (earlier) than sunrise or sunset, up to 23:59 either way. Sunrise and sunset are calculated from an algorithm developed by Kevin Walker, a user of X10CTL. Kevin graciously granted permission for its inclusion in X10CTL, for which we thank him.

PROGRAM INSTALLATION

1. Make sure that VBRUN200.DLL is in your Windows System directory. It can be found in the MS Basic forum as VBRUN2.ZIP.
2. If you still use Windows 3.0, you also need COMMDLG.DLL. This file is part of Windows 3.1. It may be on CompuServe, but I don't know where. This file is also in the Windows System directory.
3. Unzip X10C26.ZIP into a directory on your hard disk, such as C:\X10CTL. Of course, you must have done this to read these instructions, but I wish to be complete.
4. Execute X10Setup.EXE from within Windows. You can do this by clicking on the File Menu in Program Manager, then Run. You can type the complete file path or use the browse function to find X10Setup.EXE.
5. X10Setup copies the VBX files into your Windows System directory. The following files, included with the X10CTL package, are installed in the Windows SYSTEM directory:
 - spinx10.vbx
 - cmdialog.vbx

If older versions are found, X10Setup reports this and quits, to allow you to save these files somewhere else. If newer versions are found, X10Setup leaves them. If X10CTL fails to run properly, then the VBX files are not backward compatible, and I need to hear from you.

If X10Setup fails for any reason, you can manually copy these files into the Windows\System directory.

X10Setup does not move X10CTL.EXE, nor does it create a program group. Tests with alternate shells have convinced me that Microsoft's Setup utility for VB works poorly in these cases. Consult your Windows manual if you need help adding the X10CTL icon to a Program Manager group.

In your Startup group, add X10CTL with the command line

X:\dirname\X10CTL update
for example, C:\X10CTL\X10CTL update

The update switch will execute X10CTL as a minimized application, where any sunrise/sunset events are updated for changes in sun time. X10CTL will terminate when the updates are complete. The frequency of updates defaults to 7 days, and can be changed in the Setup Window of X10CTL, described later.

No changes are made to WIN.INI or SYSTEM.INI. A private .INI file, X10CTL.INI is created in the Windows directory.

PROGRAM OPERATION

The first time the program is started, X10CTL.INI is created in your Windows directory. This file maintains initialization parameters for X10CTL, including the COM port number, window positions, and unit descriptions. You will be required to enter the COM port for the X-10 Interface

the first time the program is run. If you change the port in the future, you will have to edit X10CTL.INI, since the program will not operate without the Interface active.

Main Window

The Main window provides access to the other windows in the application. Access can be gained by double-clicking the appropriate icon, or through the menu system. The Unit, Event, Batch, and Console windows can be open concurrently, with free movement between these windows.

SetUp Window

The Setup window allow you to set the X-10 Base HouseCode, set the day and time, turn on/off Daylight Savings Time, or enter your geographic location for sunrise/sunset calculations. Both Daylight Savings Time and location (latitude and longitude) are required for calculation of sunset and sunrise.

Your geographic location is expressed as a latitude and a longitude. Latitude defines your distance from the equator (latitude = 0), either north or south. Longitude defines your distance from the Greenwich Meridian (longitude = 0), either east or west. I know that some programs provide handy city lists with this information in disk files, but I didn't. If I could do this programming (for free) to satisfy requests, then you can find out where you are on a globe. Latitude and longitude can be found on a good map or in an almanac. Your local librarian should be able to help you.

Latitude should be expressed as dd'mmD, where dd are degrees, mm are minutes, and D is direction (N or S) with respect to the equator. Similarly, longitude should be expressed as dd'mmD, where D is now E or W of the Greenwich Meridian.

Unit Window

This window allows the user to describe up to 256 units (A1 - P16). The descriptions are stored in X10CTL.INI in the format

HUU=Description String

where H is the House Code (A-P) and UU is the unit number (01-16). The leading zero is required. Unit descriptions are added by first selecting House and Unit codes, then clicking the Add/Edit button. A dialog box is opened, allowing the user to enter a series of characters describing the function of the unit. Changes are saved when the application is terminated.

Unit descriptions can be edited by first selecting the unit to be edited, either using the House and Unit buttons, or by clicking the entry directly on the list, followed by clicking the Add/Edit button.

Unit descriptions can be deleted by selecting the unit of interest, followed by clicking the Delete button.

Unit number 17 (which does not exist physically) is predefined as "All Lights On/Units Off". This function was added at the request of users, who wished to be able to program "All Off" functions as events. Note that any "On" events are translated automatically to "Dim00" (Highest lamp unit output) so that "On" functions will only work on lights connected to lamp or wall switch modules. The "Off" functions will work with all units assigned to the given House Code. This behavior mimics an X-10 Command Center.

Event Window

When the Event window is first opened, the events stored in the X-10 Interface are uploaded to the computer, then displayed. Events can be added, updated (modified), or deleted. All changes to the event list are sent to the X-10 Interface immediately. The Event window thus always represents the current action list in the X-10 Interface. The Event List can be printed for

reference or easy review.

Events are added by selecting the following components in any sequence:

- Unit
- Mode (Today, Tomorrow, Everyday, Weekdays, Weekends, or Specific Days)
- Action (On, Off, or Dim)
- Time (24-hour format)
- Time Basis (Normal, Security, Sunrise, Sunset)

The Unit and Mode are selected from drop-down lists. The Unit list is constantly updated from the Unit window. If a desired unit is not defined, simply switch to the Unit window, add the unit description needed, then switch back to the Event window.

If Mode is Specific Days, a Day Selection control box is displayed. Check the days you wish to have the action take place. The box is not displayed in the other modes, since the days are implicit in the mode definition.

If Action is Dim, a Dim Level control is displayed. Use the spin buttons to select the dim level between 0% (full bright) to 100% (full dim).

To change the Time, click on the hours or minutes areas, then use the spin buttons to change the selected value.

If Time Basis is Normal, the time is the actual event time. If it is Security, then the first occurrence is the set time, and subsequent occurrences are up to an hour later, varying in a pseudorandom fashion. If Time Basis is Sunset or Sunrise, the listed time is an offset from the calculated sunrise or sunset time, displayed at the top of the form. A sign, "+" or "-", is added to the time display. Click on the sign to reverse it. The Time Basis is displayed in the event list by "\$" for Security, "S" for sunSet, and "R" for sunRise. Though it is, in theory, possible to combine a time offset with the security feature, I elected to make them mutually exclusive.

Events can be updated by first clicking on the line describing the event, then changing controls as needed, followed by clicking on the Update button. When an event is selected by clicking on it, all controls are updated to match. This simplifies changes, since most controls will be correct. Note that the days in the Specific Days mode appear gray to differentiate the days selected in the interface from those being changed.

Events can be deleted by clicking on the event, followed by clicking the Delete button. There is no undelete function.

Event files are accessed through the Event window by clicking on Save or Load. The current list of events can be saved to disk (default extension is .X10). Events loaded from a disk file will completely replace the events in the interface, i.e., any events in the interface are erased when an event file is loaded.

Note on file formats: The file format has changed in Version 2.3. Files created by earlier versions can be read, but do not contain sunset/sunrise offset information. File translation is automatic. Any files saved in version 2.3 (or later) can not be read by earlier versions.

Batch Window

The Batch window contains commands you can send to the X-10 whenever you want. The batch sequences can be stored on disk, and there are no limits to the number of batch files you can have. Starting with Version 2.1, if batch commands are in the list box, they are automatically saved to the most recently accessed batch file when the program ends. If no file was specified, the commands are stored in AutoLoad.X1b. When the Batch window is opened during a subsequent program run, the most recent file is loaded automatically.

Each line in the Batch window is a separate batch sequence having the form

```
H:OP.U1.U2.U3/ 'Comment
```

where H is the House Code
OP is the operation (on, off, dim)
U1 is a unit descriptor
' Denotes a comment, and the rest of the line is ignored.

Examples

```
L:ON.1.2.3.4/L:D4.5.6/ ' turns on units 1,2,3 and 4, then dims units 5 and 6 to Dim04 (D0 is full bright, D15 is full dim).
```

```
L:OFF.1.2.4/L:ON.6/L:D9.11.13/'turns 1,2, and 4 off; turns 6 on; dims 11 and 13 to Dim09.
```

Batch sequences are defined in a separate window, using simple point and click operation. Note that multiple units can be selected in batch mode. Units having letters large and bold will be included in the batch sequence. Starting with Version 2.3, a text box below the batch list contains a comment field. The field can be edited, and is then appended to the end of the sequence after the Close button is clicked.

The batch files are stored as simple ASCII files. You can create batch files with a text editor. If you do, any typos are your responsibility. The BatchAdd function is easy to use, and it ensures proper syntax.

TECHNICAL ISSUES

1. To all of you claiming that your fast CPU was messing up the program: NO WAY! I upgraded to a 486/40 motherboard, and all versions back to 1.01 worked as well as they did on my 386SX system. The DOS software from X-10 USA no longer works with my 486.

2. COMM ports, the X-10, and Windows: You must understand that the DOS X-10 software supplied with the X-10 console does not use interrupts when communicating with the X-10 interface (this is true for Version 1.00 2/86, and probably for newer versions). Windows does, so all the limitations of sharing IRQ's applies. In the IBM PC and AT world (any ISA bus system), COM1 and COM3 share IRQ4, while COM2 and COM4 share IRQ3. COM1 and COM3 can NOT be used at the same time, nor can COM2 and COM4 be used at the same time, PROVIDED that the software is using interrupts. All COMM programs, mouse drivers, and Windows applications use IRQs, while the DOS-based X-10 software does not.

My system can serve as an example:

COM1	X-10 interface using IRQ4
COM2	internal modem using IRQ3
COM3	not attached at the moment, using IRQ4
COM4	not installed (damn ATI, anyway!)
MOUSE	InPort (bus) mouse, using IRQ5

You can see that my modem, mouse, and X-10 all have separate interrupts assigned to them. If I connect anything to COM3, then I can only use it when I am not accessing the X-10 interface.

If you have a serial mouse, modem, and X-10 interface, then you should have at least three COM ports (maybe four). My suggestion for configuring this system is:

COM1	X-10 interface	(IRQ4)
COM2	mouse	(IRQ3)

COM3 modem (IRQ4)

Note you can't use the modem and X-10 at the same time. If you have nothing using IRQ5, then you might want to reconfigure one of your ports to use IRQ5. Most mouse drivers and Windows 3.1 will accept a nonstandard setup (provided you tell the software what you've done).

The last thing you want to do is try to share an interrupt between the mouse and the X-10. It just can't work with X10CTL.

3. A user recently sent me a dump file of the commands in his X-10 unit. It reminded me that a command and an event are two different things. A command is "turn unit L01 on", while an event is "turn these units on house code L on". X10CTL assigns one command to one event, which simplifies bookkeeping, but wastes the limited memory of the CP290 (only 128 events can be stored).

The DOS-based software from X-10 (his version, anyway) automatically combined commands having the same operation, house code, and time, thus producing commands with two or three units. This has the benefit of conserving event memory.

X10CTL cannot work this way without major revisions and fundamental changes in its operation. Right now, the events listed in the window constantly reflect the X-10 CP290 memory, so you know what is going on. If you turn off the computer, you won't have any surprises (at least from the X-10). To combine commands into events, the software would have to increase greatly in complexity to remain matched to the interface memory, or the exact tracking would have to be sacrificed.

I have used the interface since 1986, and have used X-10 modules since 1980. The largest event list I ever created had about 60 events. I can't see the need to modify the code to the extent needed for commands combined into single events.

X10CTL 2.1 (and higher) will look for these events, and will let you know when it runs into one. It will display ?? instead of a unit number, and I strongly advise against updating or deleting these events from within X10CTL.

You can force the DOS version to separate the commands by changing the times by as little as one minute. If you plan to use the DOS version alongside X10CTL, just remember to avoid having two identical commands and times. When used in this way, one command = one event, and both programs should be happy.

If this seems inelegant to you, or if you feel that wasting event memory is not possible, then let me point out that X10CTL is free, and you may use it or not as you see fit. I appreciate the interest that many of you have expressed, but I wrote this for fun, and this change won't be fun.

4. Additional features:

The original program improved in robustness, performance, and features due to the interest and ideas I received from users. The Batch facility and "All Lights On/Units Off" function were ideas I got from you. I will still entertain ideas, since this can lead to uses that I would never conceive.

Sunrise and sunset offsets were very popular requests. The algorithm to calculate sunset and sunrise was provided by Kevin Walker, who graciously allowed me to translate it from C to VB. My thanks go to Kevin for his code and his assistance in understanding it.

ERROR MESSAGES AND THEIR CAUSES

Error Opening Port

Either some other program, such as a comm program or mouse driver, is using the port, or the X10CTL application suffered an uncontrolled crash. If the former, adjust the port assignments in your computer (see Tech Section, above). If the X10CTL has crashed, you must restart Windows, since the port was not closed, and access is cut off. If X10CTL crashes, record as much information concerning your actions at that point, then e-mail a summary to me.

Error with X-10 Interface

The port has been opened and configured. The initial attempt to communicate with the X-10 Interface has failed. Possible causes include:

1. Wrong com port selected
2. X-10 Interface not connected
3. X-10 Interface unplugged
4. Miscommunication with interface

The last is a tricky one. When the X-10 Interface executes a programmed event, it sends the event information out the serial port, whether or not a computer is attached at the time. If communications are attempted at the same time an event is executed, the error message will be displayed. Simply restart the program to fix this particular error.

Correct the cause(s), if found, and try again. If everything seems normal, then it may be an unknown bug in X10CTL. If you can, try using a different program to access the interface to make sure the X-10 itself is OK.

Please forward comments, compliments, complaints, bug reports, or suggestions to

Larry E. Fosdick [70105,1142]

REVISION HISTORY

***** Changes in Version 2.2**

Corrected code to support COM3 and COM4

***** Changes in Version 2.1**

Automatic save and load for batch files added.

Multi-command events are recognized and brought to your attention, but not dealt with further (see Technical Issues, above).

Text and stuff are adjusted when in 1024x768 (or higher) video mode.

Several minor tweaks and adjustments which I can't recall and you probably wouldn't notice (unless you were the one that found the bugs).

***** Changes in Version 2.0**

Batch command processing has been added, where you can predefine a series of commands and send them whenever your little heart desires.

Now that Windows has a common dialog function, I am using it. You will see minor changes in file handling for event files.

I also experimented with gauges to show dim level and spin buttons instead of scroll bars where appropriate.

***** Changes in Version 1.4**

A very helpful user determined that X10CTL crashed when first accessing an X-10 system with a clean memory, such as that caused by a power failure without battery backup. This problem has been fixed.

***** Changes in Version 1.3**

Corrected some problems with the file handler. It seems that events were uploaded to the Interface faster than it could accept them, resulting in garbage. Also corrected some strange things that crept into Version 1.2 somehow that affected the appearance but not the performance.

***** Changes in Version 1.2**

The ability to print the event list has been added.

A unit description of "All Lights On/Units Off" has been added to the Unit Window. This was primarily added for those who wish to send "All Off" commands while away from home. I have no idea who will use it to turn lights on. Please note that the code uses DIM00 instead of ON, so that this control will only turn on lights that are connected to lamp modules or wall switches. The off function affects all units on the selected house code. The unit number for this function is 17, which is not real, but gives the software a hook.

***** Changes in Version 1.1**

First of all, version numbers became two digits.

The event upload has been changed again. It should now work for any speed CPU, without adding silly delays. If you have trouble with this version, please retest the program with a slower cpu speed, if your computer allows cpu speed switching, then report the results to me.

The intro window was added once I saw how long the screen went blank while initializing the program. An odd quirk of Windows causes the focus to go to a window other than the Main window on my system, so I tried to incorporate a work-around using the timer. It caused more problems than it solved, so I took it back out. If the Main window doesn't have the focus when it first comes up, sorry.

It turns out that if you terminate a multiple window Visual BASIC program (like this one), the Form Unload subprogram is only executed on the primary window unless you unload explicitly the other windows. This "feature" caused problems with saving changes to the Unit Description List. This has now been fixed.

***** Changes in Version 1.01**

Three changes were made. The first addresses an "Illegal Function Call" error when first starting the program and setting the COM port for the X-10. This was a silly oversight. I use the timer function to ensure that the Main window gets the focus after the intro window is removed (normally, the focus goes to some other window). I now start the timer after the port configuration window is closed.

The second fix deals with Event updates not working properly in 1.00. It was simply a

bookkeeping error.

The third fix deals with uploading events from the interface. The X-10 Powerhouse does not support any form of handshaking, and so the event uploading has to be handled by a combination of timing and waiting for the right number of characters in the buffer.