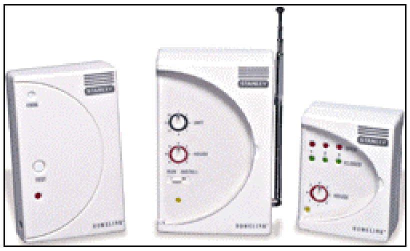
Controlling the Status Indicator Module
# of the Stanley Garage Door Opener Set

1. Introduction

This document describes how it is possible with Extended X10 commands in HomeVision to control the 3 status indicator lights (LEDS) on the **Status Indicator Module** of the Stanley Garage Door Opener Set (see Picture 1; see also for example: http://www.smarthome.com/7463.html for pricing information). I am in no way affiliated with the businesses that manufacture or sell these modules, or HomeVision. I am just an interested user.

Picture 1



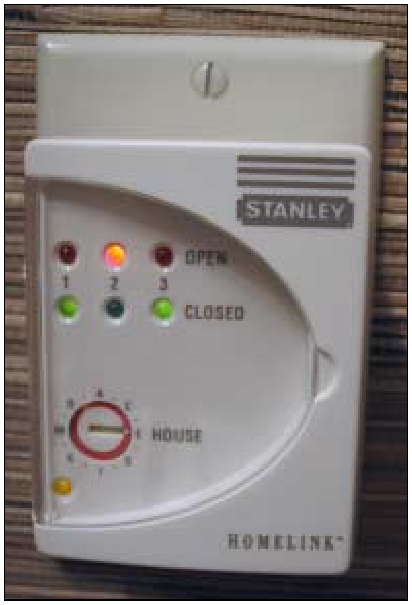
The original setup and intended use of the whole set (3 modules) is as follows. The **Sensor Unit** (device on the left in Picture 1) is attached to the garage door you want to monitor. The **RF Base Module** (middle device in Picture 1) is plugged into a nearby power outlet. Finally, the **Status Indicator Module** (device on the right in Picture 1) is plugged into any other power outlet in your home. A red LED tells you that the garage door is open, a green LED shows you it is closed. Users can add as many indicator modules as they like in case they want to monitor garage doors from several locations in their home. Three sets of LEDs on each Status Indicator Module allow monitoring up to 3 garage doors simultaneously. The garage door sensor determines the open/close position by sensing if it is horizontal or vertical. It then transmits any change of status to the RF Base Module, which broadcasts the change in position to the Status Indicator Module via a power line signal. The sensor can be used to monitor other devices that change position horizontally/vertically.

My idea was to use the Status Indicator Module alone (this module is for sale by itself) to show the status of anything (not just garage doors). I therefore ordered the Status Indicator Module, and tested it in combination with HomeVision. Several users in several discussion groups on the Internet were helpful in providing me tips and suggestions regarding the X10 commands that were used to control the Status Indicator Module. This document contains the results of my testing, opening many possibilities for this module. Examples of items to be indicated:
garage door open/close (although this module can accomplish this in conjunction with the two other hardware devices as shown in the Picture 1, I already have a different system with a magnetic switch that can used with this), doors/windows open/close, remote light on/off, alarm system on/off, motion detected, etc.

2. Description of the Status Indicator Module

Picture 2 shows a larger photo of the Status Indicator Module plugged into a wall power receptacle in my home. The module contains a dial to set the X10 House Code (Set to E in Picture 2). The LED below the House Code Dial turns yellow when the module is plugged in and power is applied to it. The important parts are the 3 sets of lights, numbered 1, 2, and 3. For each number, there are 2 lights: one red, and one green. For each number, there are 3 possible states: both lights OFF, RED on, or GREEN on.

Picture 2



3. Control of Status Indicator Module with HomeVision

The lights are controlled by Extended X10 commands. See ftp://ftp.x10.com/pub/manuals/xtc798.doc for a technical discussion of X10 Extended Code Formats. Although the TW-523 module that interfaces the HomeVision box with the power line cannot interpret Extended X10 commands, HomeVision is able to transmit Extended X10 codes through this interface. Therefore, it is possible to control the lights on the Status Indicator Module (It is not possible in HomeVision to monitor the Extended X10 codes that are send by the RF Base Module of the garage door set, for example. Again, this is not a limitation of HomeVision but of the TW-523 module).

The lights can be controlled from the HomeVision software with the "Xmit extended code ## and ##" command (right-hand side of the screen labeled "Custom Commands" in X10 Control). In addition, the 32-bit version of the HomeVision software contains identical commands that can be added into a HomeVision schedule. There are 2 extra bytes that need to be provided, labeled ***Command*** and ***Data***. In addition, the proper ***House Code*** and ***Unit Code*** need to be selected.

The values and the significance of all codes needed are as follows:

House Code
Same letter as set on the house code dial of the module

Unit Code
Determines which lights are being controlled (i.e., will be set to RED or GREEN). If the light number is not included in the selection, both lights for that number will be off, irrespective of the data byte in the Extended X10 code (see Data below). The meaning of the Unit Code is as follows:

Unit Code	Selected Lights
1 or 2	2 and 3
3 or 4	2
5 or 6	1
7 or 8	1 and 3
9 or 10	1, 2 and 3
11 or 12	1 and 2
13 or 14	none
15 or 16	3

Command
Always 29 (decimal)

Data
Determines whether the lights will be RED or GREEN, as follows:

Data Byte	Light 1	Light 2	Light 3
112	Green	Green	Green
113	Red	Green	Green
114	Green	Red	Green
115	Red	Red	Green
116	Green	Green	Red
117	Red	Green	Red
118	Green	Red	Red
119	Red	Red	Red

From this table, it is clear that bit1 (rightmost bit) controls light 1 (set to 1 for RED, set to 0 for GREEN), bit2 controls light 2, and bit 3 controls light 3. It's important to know that the lights are only turned to red or green when the proper Unit Code is selected to control any particular light (otherwise they are both off). Bits 4 and 8 are ignored.

Bits 5-7 have some special meaning, explained in this paragraph. If the plugged-in Status Indicator Module doesn’t receive a status update within 4 hours or so, the lights that were on before, start blinking. This is actually a nice feature that would tell you that the module wasn't updated properly, in which case the validity of the status is questionable. Some further testing revealed that the blinking can only be turned off when the corresponding bit of the ***Data*** byte is set to 1 (bit5 for light 1, bit6 for light 2, and bit7 for light 3). Therefore, the data bytes in the ***Data*** table above have bits 5-7 always set to 1 (112 decimal added) to stop or prevent blinking, if it would appear. I know of no way to intentionally initiate the blinking.

4. Control of the Status Indicator Set in a HomeVision Schedule

With all this information, it is possible to have 3 flags in a HomeVision schedule to indicate the status of almost anything tracked in HomeVision, and to have that status reflected into the 3 lights of this module (see Schedule examples below). Actually, it is possible to have such a module for each House Code, if one would want to indicate the status of more than 3 items.

Some X10 code examples:
Unit Code=9, Data=112: turns all 3 lights GREEN
Unit Code=9, Data=119: turns all 3 lights RED
Unit Code=11, Data=112: turns lights 1 and 2 GREEN, light 3 OFF
Unit Code=11, Data=119: turns lights 1 and 2 RED, light 3 OFF

Examples in a schedule:
Any Extended X10 code appropriate to change the lights on the Status Indicator Module affects all the lights. This means that there are no commands to separately change one single light, leaving the other lights on the module unchanged. Therefore, a schedule need to keep track of the status with variables or flags, from which the Extended X10 command will be constructed that contains the composite codes to command all 3 lights at once. The first thing to decide is how many of the lights will be used. With that decision up front, a user can then include the appropriate codes into a schedule, with subsequent calculation of the composite code.

Let's assume that we want to use all 3 lights to indicate 3 separate items (the most complicated case). My schedule contains 3 variables (renamed **Led1**, **Led2**, and **Led3**) that indicate the status of the 3 lights. Each variable is set to 0 to turn on the GREEN light, or set to 1 to turn on the RED light. My macro "Update 3Leds" (see below) then calculates the composite ***Data*** byte (first 5 lines) into a variable **Result1**. The programming statements in the schedule calculate the composite code as follows:

Result1 = **Led3** * 4 + **Led2** * 2 + **Led1**

Which is equivalent to

Result1 = ((**Led3** * 2) + **Led2**) * 2) + **Led1**

And which gives the composite decimal value of the lower 3 bits in the ***Data*** byte (eight possible values, ranging from 0 to 7). If we would be able to define the ***Data*** value in the Extended command in the HomeVision schedule by a variable (an idea for future additions to HomeVision), we would use Result1 with 112 added to it. Instead, we need to define the arguments in the Extended X10 command (Transmit Extended Code) as fixed numbers. This is done in the schedule, and selected by eight IF-THEN statements. The House Code is E since the House Code wheel on my Status Indicator Module is set to E. The Unit Code is 9 since we are controlling all 3 light numbers. The ***Command*** byte is always 29 for controlling the Status Indicator Module.

```
MACRO EVENT # 20 'Update 3Leds'

Var  #3 (Result1) = var #2 (Led3)
Var  #3 (Result1) = var #3 (Result1) × 2
Var  #3 (Result1) = var #3 (Result1)  +  var #1 (Led2)
Var  #3 (Result1) = var #3 (Result1) × 2
Var  #3 (Result1) = var #3 (Result1)  +  var #0 (Led1)
If
        Var #3 (Result1) = 0
Then
        X-10: E 9 (StatusLight1&2&3) Transmit Extended Code: Command=29, Data=112
End If
If
        Var #3 (Result1) = 1
Then
        X-10: E 9 (StatusLight1&2&3) Transmit Extended Code: Command=29, Data=113
End If
If
        Var #3 (Result1) = 2
Then
        X-10: E 9 (StatusLight1&2&3) Transmit Extended Code: Command=29, Data=114
End If
If
        Var #3 (Result1) = 3
Then
        X-10: E 9 (StatusLight1&2&3) Transmit Extended Code: Command=29, Data=115
End If
If
        Var #3 (Result1) = 4
Then
        X-10: E 9 (StatusLight1&2&3) Transmit Extended Code: Command=29, Data=116
End If
If
        Var #3 (Result1) = 5
Then
        X-10: E 9 (StatusLight1&2&3) Transmit Extended Code: Command=29, Data=117
End If
If
        Var #3 (Result1) = 6
Then
        X-10: E 9 (StatusLight1&2&3) Transmit Extended Code: Command=29, Data=118
End If
If
        Var #3 (Result1) = 7
Then
        X-10: E 9 (StatusLight1&2&3) Transmit Extended Code: Command=29, Data=119
End If
```

The macro “Update 3Leds” can be tested with the following macro, called “Test Status Indicator”. When this macro is started from the HomeVision software, with the Status Indicator Module plugged in nearby, one can see consequently all the 8 possible combinations of green and red lights. The whole test takes about 10 seconds. The sequence starts with all 3 lights green, and changes then one light at a time.

```
MACRO EVENT # 19 'Test Status Indicator'

Var  #0 (Led1) = 0
Var  #1 (Led2) = 0
Var  #2 (Led3) = 0
Do macro #20 (Update Leds) once
Var  #0 (Led1) = 1  -  var #0 (Led1)
Do macro #20 (Update Leds) once
Var  #1 (Led2) = 1  -  var #1 (Led2)
Do macro #20 (Update Leds) once
Var  #2 (Led3) = 1  -  var #2 (Led3)
Do macro #20 (Update Leds) once
Var  #1 (Led2) = 1  -  var #1 (Led2)
Do macro #20 (Update Leds) once
Var  #0 (Led1) = 1  -  var #0 (Led1)
Do macro #20 (Update Leds) once
Var  #1 (Led2) = 1  -  var #1 (Led2)
Do macro #20 (Update Leds) once
Var  #2 (Led3) = 1  -  var #2 (Led3)
Do macro #20 (Update Leds) once
```

The statements wherein a variable is subtracted from 1 are an easy way to switch its value from zero to one, or from one to zero. The macro “Test Status Indicator” is for testing purposes only.

An example of a real application is to monitor the status of a remote light. The following two X10 signal events (A14-On and A-14-Off), detected when HomeVision sees the X10 signal that changes my office lights (not really remote, but good enough for testing), triggers an evaluation of the **Led1** variable, followed by an update of the status lights. The programming code below is executed after any power line transmission of A14-On or A14-Off. This works fine for remotes, or for devices such as the Maxi Controller, which all work through the power line. But in order to intercept the local control at the wall switch, the switch needs to be able to send a power line signal if switched. As an example, this is possible with the SwitchLinc X-10 two-way switches that send out equivalent power line signal when switched at the wall. In addition, one could also periodically request the status from these types of switches in a HomeVision schedule.

```
X-10 SIGNAL 'ON' EVENT # A 14 'Office Light'

Var  #0 (Led1) = 1
Do macro #20 (Update Leds) once


X-10 SIGNAL 'OFF' EVENT # A 14 'Office Light'

Var  #0 (Led1) = 0
Do macro #20 (Update Leds) once
```

To also intercept more global X10 commands such as A-All Lights On and A-All Units Off (buttons that I very often use on a Maxi Controller), the following programming code is needed as Periodic Events with an event rate of “each loop”.

```
PERIODIC EVENT # 1 'Check X10 Sequences'

If
        X-10 sequence [A-All Lights On] received within 5 seconds
Then
        Var  #0 (Led1) = 1
        Do macro #20 (Update Leds) once
End If
If
        X-10 sequence [A-All Units Off] received within 5 seconds
Then
        Var  #0 (Led1) = 0
        Do macro #20 (Update Leds) once
End If
```

This is just a simple illustration of possible use of the Status Indicator Module. In addition, to avoid blinking lights on the Status Indicator Module, the lights will need to be updated periodically with a periodic event every 4 hours, for example:

```
PERIODIC EVENT # 0 'Periodic LED Update'

Do macro #20 (Update Leds) once
```

In case that only 2 lights are used, the following macro (“Update 2Leds”) commands only lights 1 and 2 (Unit Code 11), and light 3 is off.

```
MACRO EVENT # 21 'Update 2Leds'

Var  #3 (Result1) = var #1 (Led2)
Var  #3 (Result1) = var #3 (Result1) × 2
Var  #3 (Result1) = var #3 (Result1)  +  var #0 (Led1)
If
        Var #3 (Result1) = 0
Then
        X-10: E 11 (StatusLight1&2) Transmit Extended Code: Command=29, Data=112
End If
If
        Var #3 (Result1) = 1
Then
        X-10: E 11 (StatusLight1&2) Transmit Extended Code: Command=29, Data=113
End If
If
        Var #3 (Result1) = 2
Then
        X-10: E 11 (StatusLight1&2) Transmit Extended Code: Command=29, Data=114
End If
If
        Var #3 (Result1) = 3
Then
        X-10: E 11 (StatusLight1&2) Transmit Extended Code: Command=29, Data=115
End If
```

In the simple case that a user is only using the first light, the following macro (“Update 1Led”) commands only light 1 (Unit Code 5), and lights 2 and 3 are off.

```
MACRO EVENT # 22 'Update 1Led'

If
        Var #0 (Led1) = 0
Then
        X-10: E 5 (StatusLight1) Transmit Extended Code: Command=29, Data=112
End If
If
        Var #0 (Led1) = 1
Then
        X-10: E 5 (StatusLight1) Transmit Extended Code: Command=29, Data=113
End If
```

This macro can be used with using flags instead of variables, as follows:

```
MACRO EVENT # 23 'Update Led1withFlag'

If
        Flag #1 (Light1Flag) is cleared
Then
        X-10: E 5 (StatusLight1) Transmit Extended Code: Command=29, Data=112
End If
If
        Flag #1 (Light1Flag) is set
Then
        X-10: E 5 (StatusLight1) Transmit Extended Code: Command=29, Data=113
End If
If
        Flag #1 (Light1Flag) is neutral
Then
        X-10: E 13 (StatusLightsNone) Transmit Extended Code: Command=29, Data=112
End If
```

In addition, the last IF-THEN statement interprets a neutral state of the flag as turning light 1 OFF (neither GREEN or RED, coded with a House Code of 13). This way, the flag indicates 3 different states of the status light.

5. Fun and Games

It’s not all serious Home Automation stuff. One can have fun with the Status Indicator Unit as well. Here is a HomeVision schedule macro called “Game Play” that randomly turns one of the 3 lights RED. Let your kids guess which of the 3 lights will come on, and the winner gets to mow the lawn (just kidding). The macro goes through the sequence of lights using the test macro to keep the suspense going (where did I say that this test macro was for testing only?). It then saves the current centisecond value in variable **Results2**, which is then tested and used as a simple random generator. The range of the centisecond value is 0 to 99. Since the one hundred possible values cannot be easily divided in three sets, a value of zero is rejected. For the value of zero, there are no winners indicated by all lights GREEN (Dad gets to mow the lawn; not a bad deal since there is only one chance in a hundred that this value will be selected). The other 99 values decide who is the winner: for values in the range 1-33, light 1 turns RED; for the range 34-66, light 2 turns RED; and for the remaining values 67-99, light 3 is turned RED. You could activate this game macro from a remote, and play it over and over.

```
MACRO EVENT # 24 'Game Play'

; Flashing lights to keep suspense going
Do macro #19 (Test Status Indicator) once
; Reset all light flags to GREEN
Var  #0 (Led1) = 0
Var  #1 (Led2) = 0
Var  #2 (Led3) = 0
; Use centisecond as random generator
Var  #4 (Result2) = current centisecond
If
        ; Exclude centisecond = 0 -> nobody wins (all 3 light flags stay GREEN)
        Var #4 (Result2) > 0
Then
        If
                Var #4 (Result2) < 34
        Then
                ; centisecond 1-33 -> light 1 wins (turns RED)
                Var  #0 (Led1) = 1
        Else
                If
                        Var #4 (Result2) < 67
                Then
                        ; centisecond 34-66 -> light 2 wins (turns RED)
                        Var  #1 (Led2) = 1
                Else
                        ; centisecond 67-99 -> light 3 wins (turns RED)
                        Var  #2 (Led3) = 1
                End If
        End If
End If
; Update lights to indicate winner
Do macro #20 (Update Leds) once
```

6. Limitations

I can think of one limitation of the system outlined in this document. It is not possible to combine some of the lights on a Status Indicator Module controlled by the other two units of the Garage Door Opener Set, with some other lights on the same module controlled by HomeVision as described in this document. The reasons are the limitation of the TW-523 module as explained above, and the fact that all lights on the Status Indicator Module are controlled by one Extended X10 command. In order for the combination to work, HomeVision would need to keep track of the Extended X10 commands sent by the RF Base Module to combine these setting of the lights with the custom control setting of the other lights on the same module. And the TW-523 is incapable of interpreting Extended X10 codes for the tracking of the Set. It is possible however to use multiple Status Indicator Modules, on different House Codes, for different purposes.

Rene Braeckman
e-mail: rbraeckman@ceptyr.com
April 6, 2000