SIS USER MANUAL

Bob Glicksman, Jim Schrempp

Date: 12/29/2015

Contents

1. INTRODUCTION.

1.1. Overview.

This document provides detailed instructions for using the SIS Client app. The SIS Client app is a javascript-enhanced web page. The SIS Client can be run on any device that has a web browser that supports javascript. The SIS Client has been tested on the following devices/browsers, but should run on virtually any device running any modern web browser. Tested environments include:

- Windows/Internet Explorer.
- Windows/ Google Chrome

- Android/ Google Chrome
- Mac/ Safari
- Mac/Firefox
- iOS/Safari

1.2.Pre-requisites.

Before you can use the SIS Client, you must have completed the following:

- Read and accepted the Terms of Use of SIS at:

https://github.com/TeamPracticalProjects/SISProject/blob/master/SISDocs/Terms_of_Use_License_and_Disclaimer.pdf

- Read and understood the SIS Theory of Operation and Installation Guide at:

https://github.com/TeamPracticalProjects/SISProject/blob/master/SISDocs/SIS_Theory_of_Operation_and_Installation.pdf

- Purchased parts and assembled an SIS Hub, per the instructions at:

https://github.com/TeamPracticalProjects/SISProject/blob/master/SISDocs/SIS_Hub_PCB_Assembly_Instructions.pdf

- Optionally, mounted your SIS Hub electronics in a plastic enclosure, per:

https://github.com/TeamPracticalProjects/SISProject/blob/master/SISDocs/SIS_Hub_Packaging_Assembly_Instructions.pdf

- Created a free cloud account at *particle.io* and used this account to upload and test the firmware to your SIS Hub:

https://github.com/TeamPracticalProjects/SISProject/blob/master/SISDocs/SIS_Firmware_Installation.pdf

- Planned your installation and purchased and installed your sensors, per the instructions in the aforementioned Theory of Operation and Installation Guide.

2. LOGGING IN TO THE SIS CLIENT.

You can obtain the SIS Client app by opening your web browser and going to the SIS home page:

http://shrimpware.com/SIS/

After selecting the "Standard SIS Client Control", you should then be presented with a screen similar to that shown in figure 1:

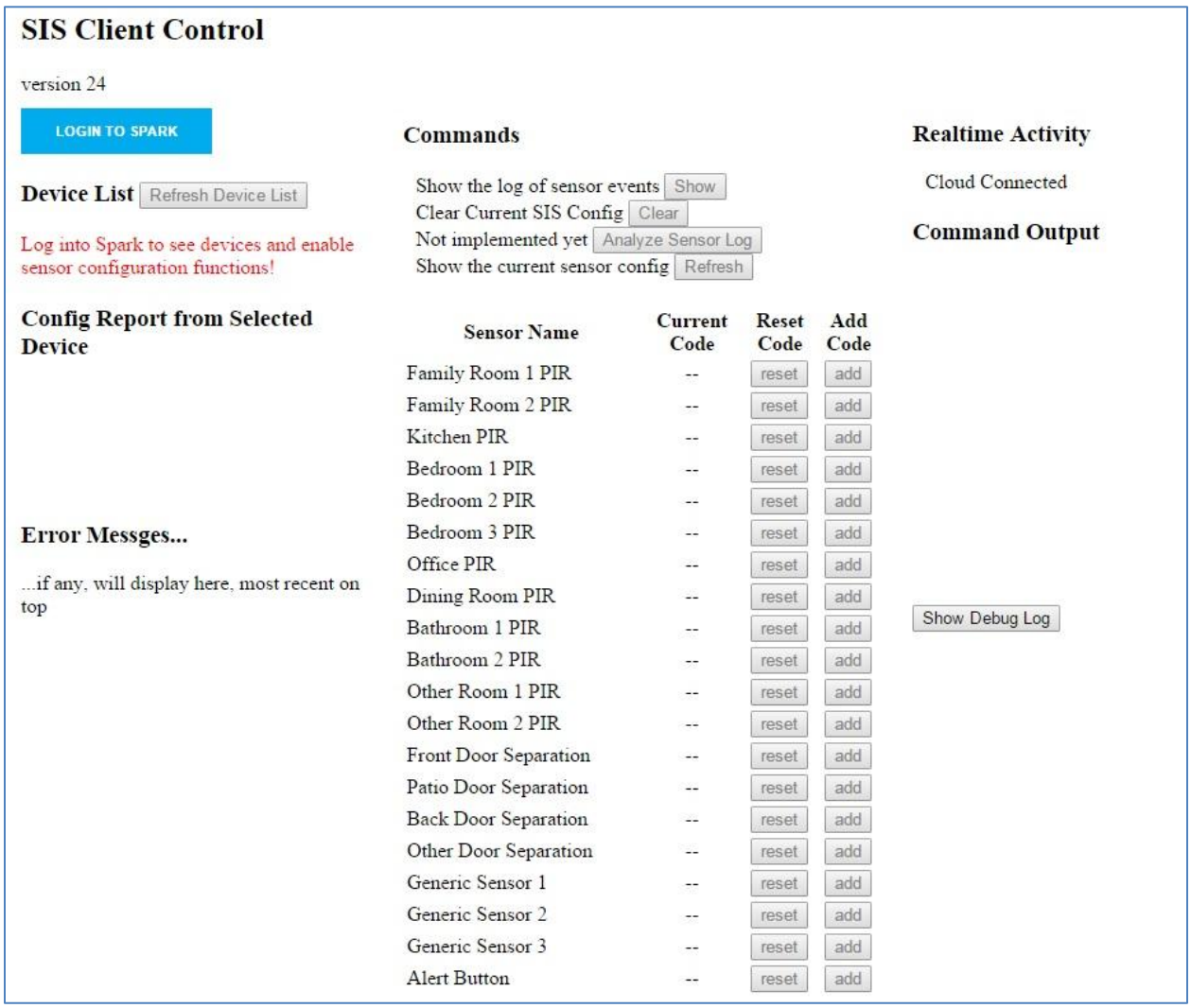

Figure 1. SIS Client Opening Screen.

The next step is to log into the *particle.io* cloud in order to establish communication with your SIS Hub device. Click the blue button at the top left of the screen marked "LOGIN TO SPARK". You will see a login panel over a grayed out screen, similar to figure 2:

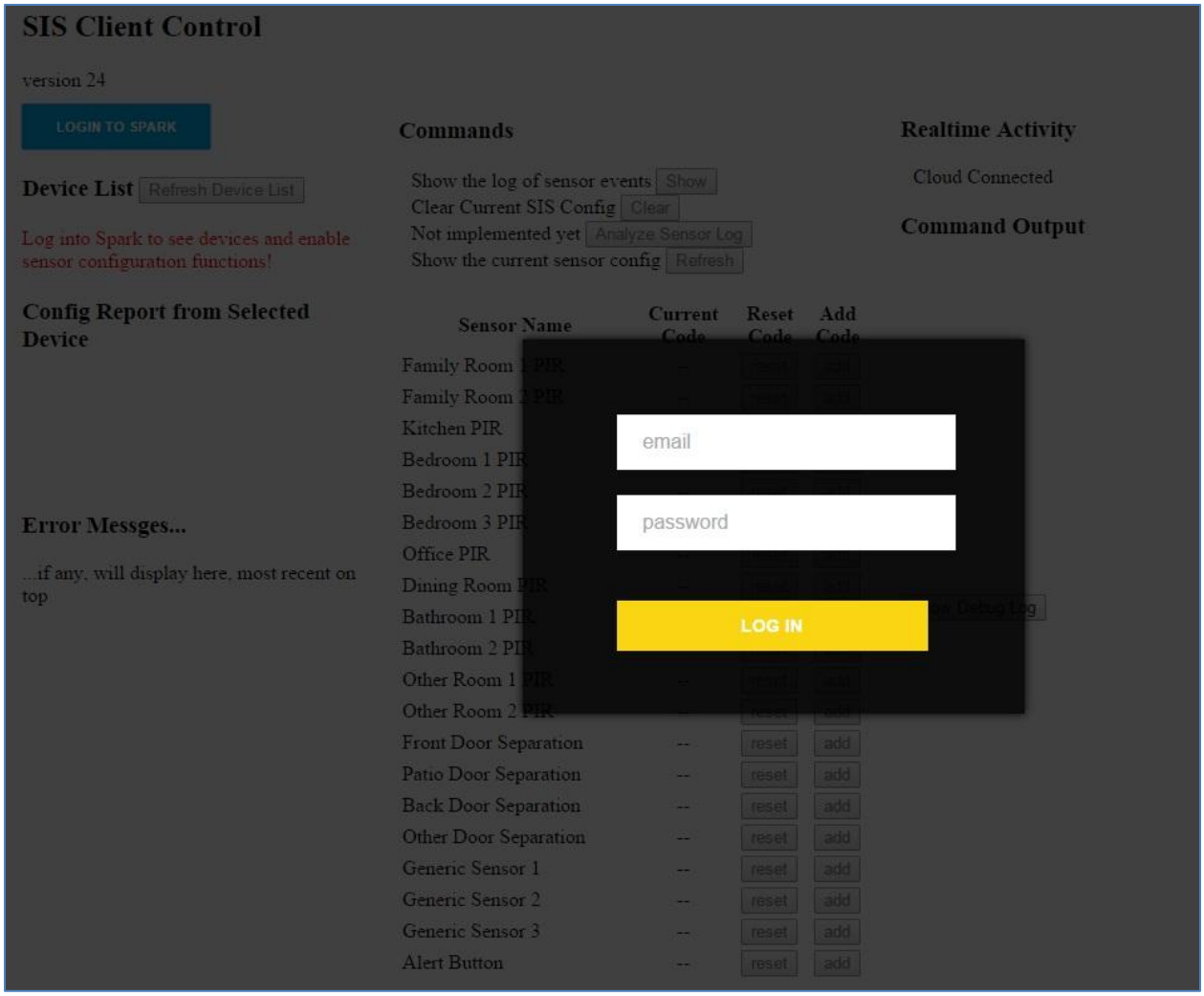

Figure 2. Log-in Panel.

Use the Login Panel to log into your *particle.io* account using your e-mail and *particle.io* password. Ten click the yellow LOG IN button to log in. You should then see a screen similar to figure 3 (cropped off):

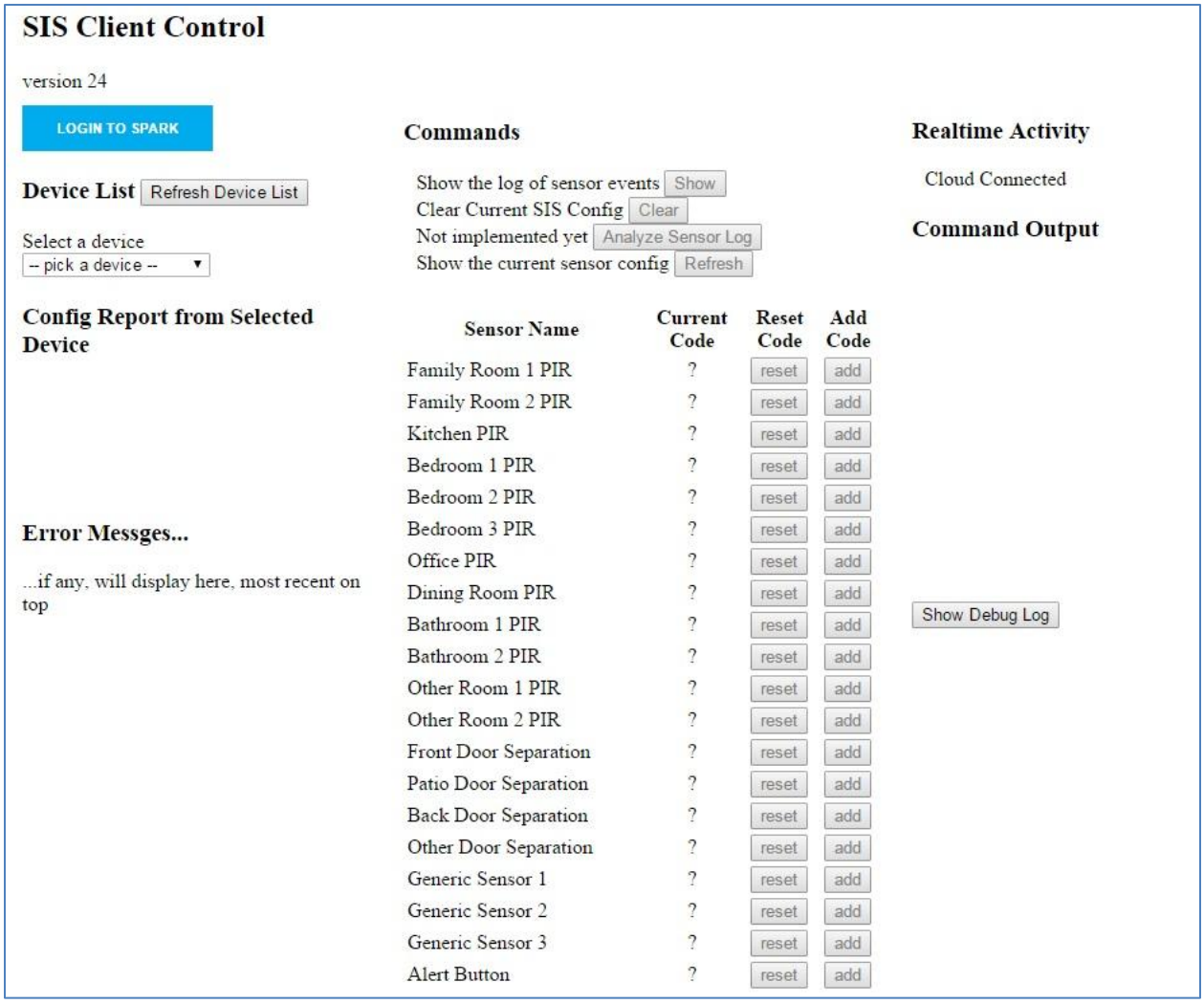

Figure 3. Device Selection Screen.

Referring to figure 3, you will see "Select a device" below the **Device List** heading on the upper left hand side of the screen. Below this is a drop down list that says " – pick a device – ". Click on this list and select the device that is the SIS Hub that you wish to communicate with. Note that you can have many SIS Hubs in your *particle.io* account.

Make sure that your SIS Hub device shows as "ON LINE" in the dropdown list. Your device will be online when it is powered up and connected to the *particle.io* cloud via the home's WiFi. After selecting your device from the drop down list, you will see a screen similar to figure 4. Note that it will take a few seconds to fully populate this screen, as the javascipt code behind the screen display is accessing your selected SIS Hub and reading out the current configuration information.

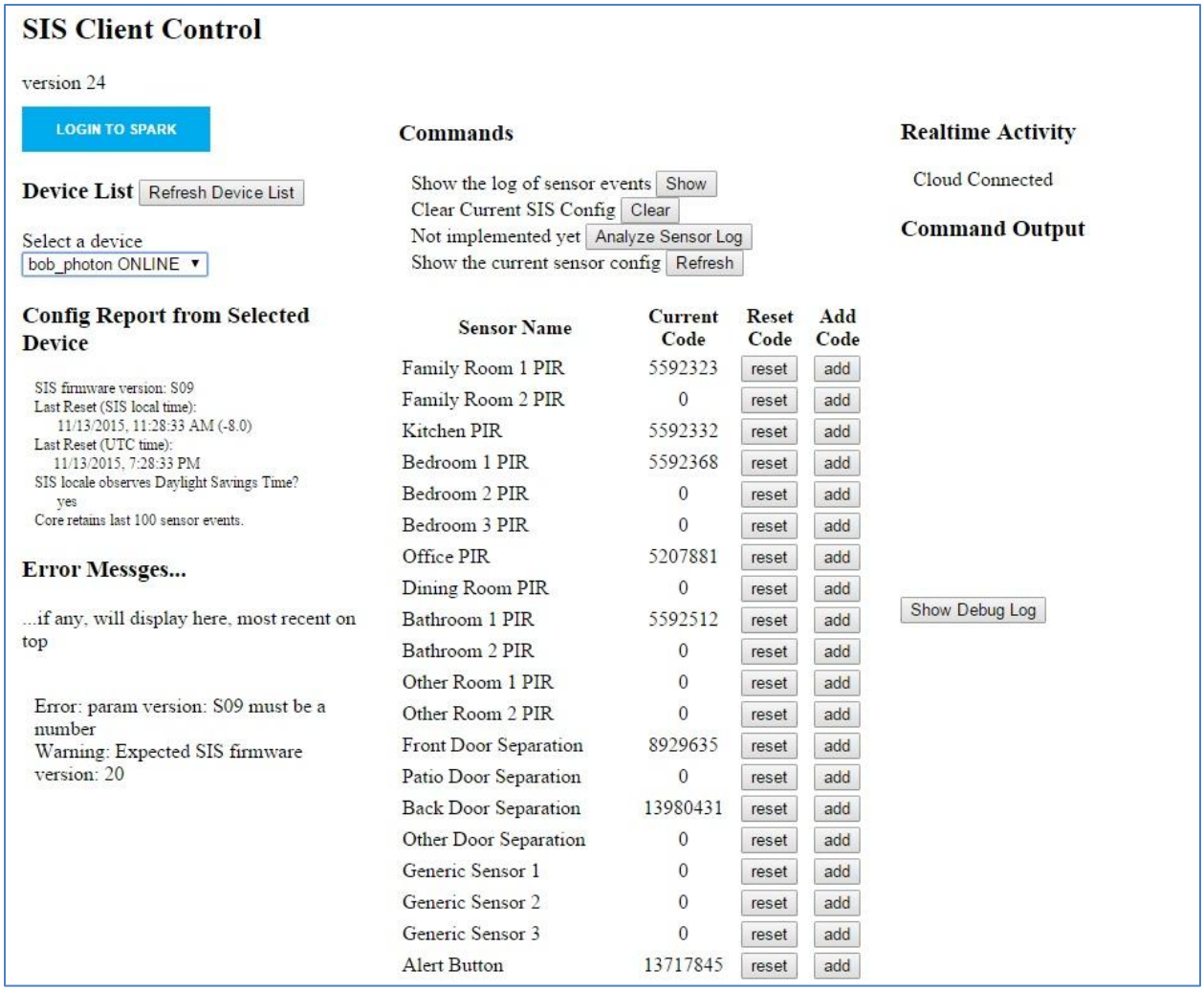

Figure 4. Device Connected Screen.

You will note several things on the screen of figure 4:

- On the left hand side of the screen, under your selected device, will be information about the device that you just connected with (**Config Report from Selected Device**). In particular, the time of the last reset will be displayed. You should take note of this time so that when you read out the log, you will know if the SIS Hub has been reset (e.g. due to a power failure) and thus the starting date/time for the log entries.

- One the right hand side of the screen, you will see the message "Cloud Connected" under **Realtime Activity**. This message will periodically blink green to indicate that cloud communication with the selected SIS Hub is ongoing. If this message is red, it means that the designed SIS Hub device is not presently connected to the *particle.io* cloud over the Internet.

- In the center of the screen, you will see the current sensor configuration for the SIS Hub. Any entry with a non-zero "Current Code" is a room/location with a sensor registered to it.

3. SENSOR REGISTRATION USING THE SIS CLIENT.

The section describes how to register sensors and manage the sensor configuration within an SIS installation using the SIS Client. You must be thoroughly familiar with the SIS Theory of Operation and Installation Manual in order to properly configure, place and register your sensors:

https://github.com/TeamPracticalProjects/SISProject/blob/master/SISDocs/SIS_Theory_of_Operation_and_Installation.pdf

Note that you need to trip sensors installed in the home as part of this registration process. Therefore, you should be running the SIS Client app on a laptop, tablet or mobile phone that is battery powered and connected to the Internet via WiFi. This way, you can go to each sensor location, trip the sensor and register it using the SIS Client app as you do so.

Referring to figure 4, above: sensor registration is managed in the center column (**Commands**) of the SIS Client display. There are two commands directly under the heading that are applicable to sensor configuration management:

- Clear Current SIS Config (**Clear**): Clicking on the **Clear** button will clear out the sensor configuration. This means that all sensor locations (rooms) will not have any sensor registered to them. **Clear** should be used only when re-using an SIS Hub at a new location, so that all previously registered sensors can be unregistered with one button click. Otherwise, the individual room resets should be used (see below).

- Show the current sensor config (**Refresh**): The table underneath the **Commands** shows the sensor configuration when the SIS Client last contacted the SIS Hub. The SIS Client attempts to keep this table up to date as you make configuration changes. However, you can always update the table to display the current sensor configuration by clicking **Refresh**.

Below these command buttons is a table that displays the current sensor configuration for the selected SIS Hub. The table also includes **Reset Code** and **Add Code** buttons. These buttons affect only the room (location) that they are associated with. These buttons should be used to remove or change a sensor at a particular location in the home.

The sensor configuration table displays the room/location under the heading **Sensor Name**. The names of these rooms/locations are fixed in the SIS Client and cannot be changed by the user. When planning an SIS sensor configuration, the user should map locations in the home with the best fit name in the table. The table has provision for 12 PIR motion sensors, 4 exterior door

separation sensors, three generic sensors and one Alert sensor. The specific meanings of each of these types of sensors in described in the SIS Theory of Operation and Installation Manual, and you must configure each sensor as the proper type in order for SIS to work properly.

The column labeled **Current Code** shows a decimal number representation of the sensor trip code associated with that location. If the location does not have a sensor registered to it, the **Current Code** will show 0.

Before you can add a sensor to a location, the location's sensor configuration must be unregistered (**Current Code** of 0). If you try to add a sensor to a location with a sensor registered to it, you will get an error message and the old sensor trip code will remain registered to that location. In order to change the registration to a new sensor, you must click on the **reset** button to reset the location to unregistered. Thereafter, you can register a new sensor to that location.

Figure 5 shows the same SIS Client display as figure 4, except that the **reset** button has been pressed on the Office PIR line in the table. After a few seconds, the display will refresh the sensor configuration table as shown in figure 5.

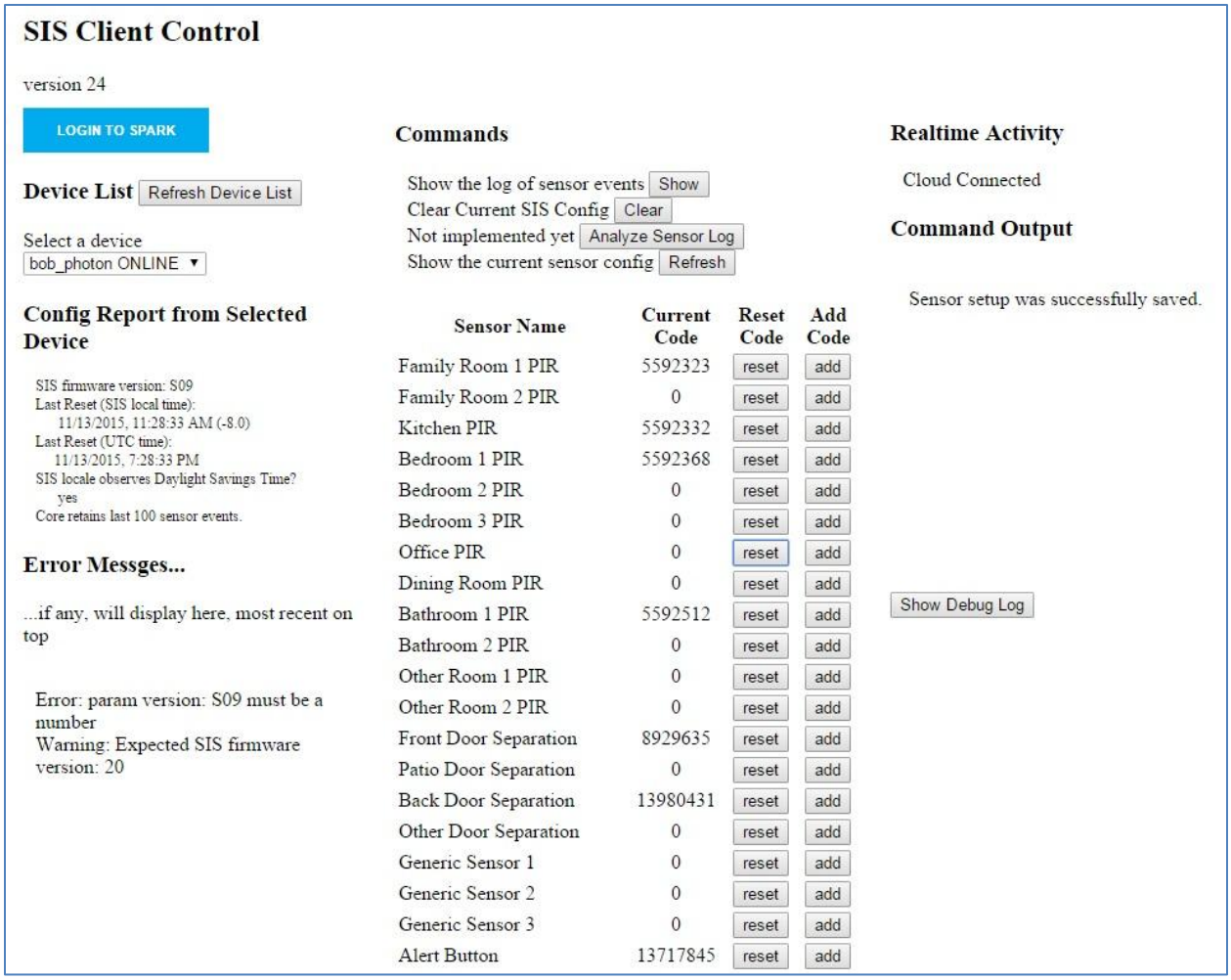

Figure 5. Reset the "Office PIR" Sensor Configuration to *unregistered*.

Now, a new sensor can be registered by pressing the **add** button in the **Add Code** column of the **Office PIR** row in the table. When you do this, you will get a popup window, as shown in figure 6:

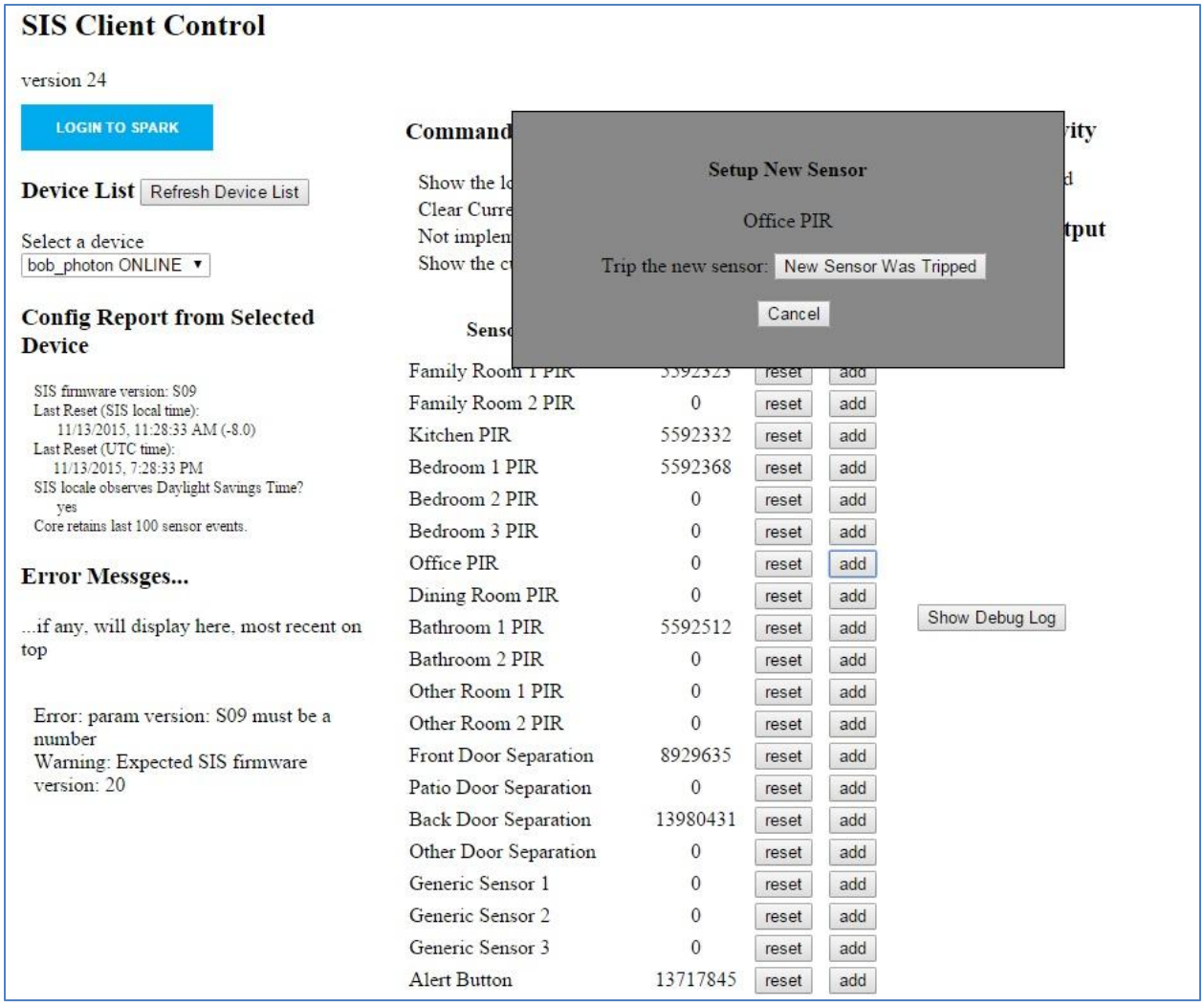

Figure 6. Adding a Sensor to the Office PIR Location.

When you see the popup window in figure 6, trip the sensor at that location. You can verify that the sensor was tripped by observing the LED on the sensor. All SIS sensors have an LED that lights up for a few seconds after the sensor is tripped.

After tripping the new sensor, click the **New Sensor Was Tripped** button in the pop up dialog box. You will then see a display similar to figure 7:

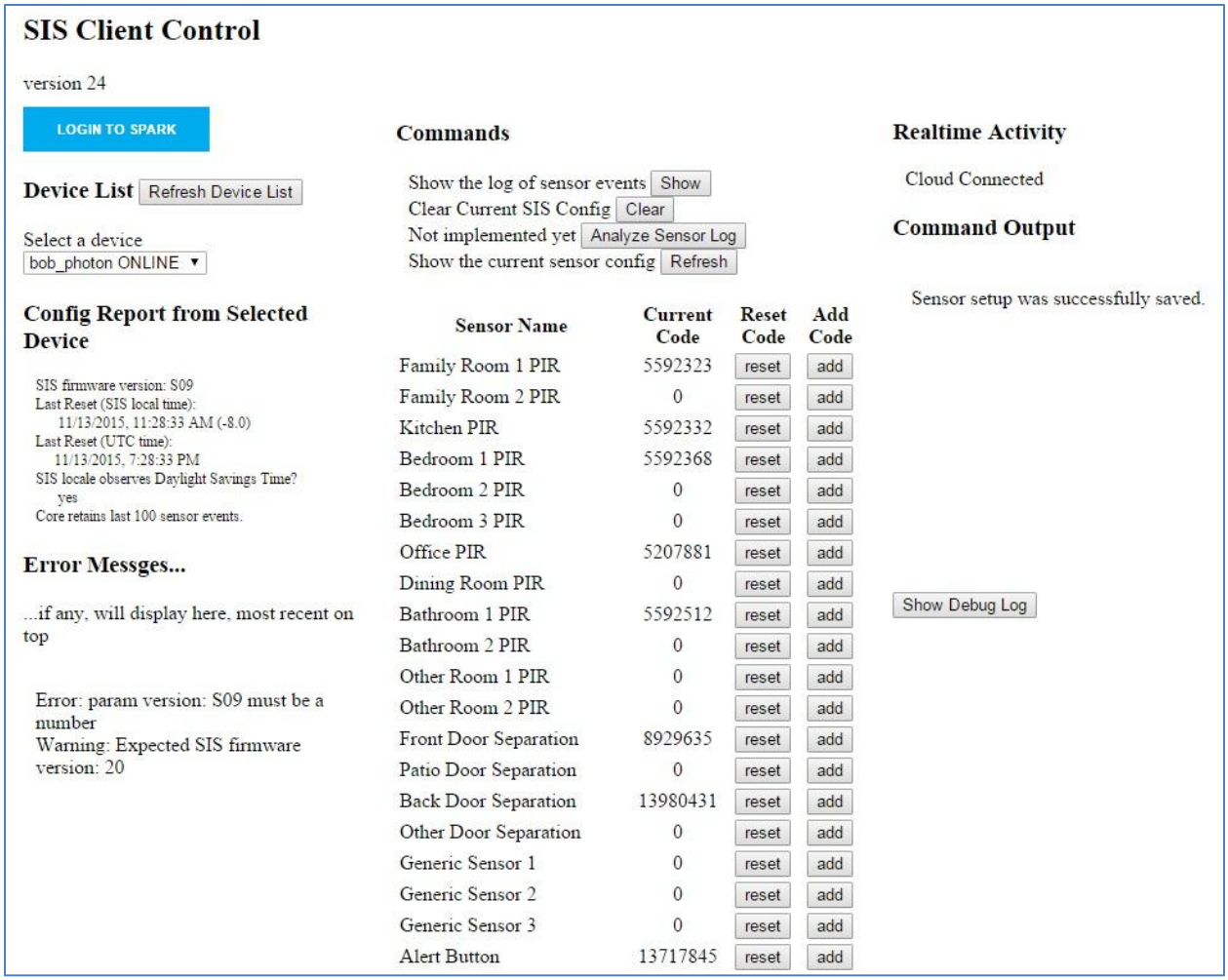

Figure 7. Registration of a New Sensor.

If the operation was successful, you will see the new sensor trip code listed in the table and the message "Sensor setup was successfully saved." under the **Command Output** heading on the right hand column of the SIS Client display.

4. READING AND INTERPRITING THE SIS LOG.

The center column of the SIS Client app screen, under **Commands**, contains a button:

"Show the log of sensor events" **Show**.

Clicking on **Show** displays the SIS Hub log, under **Command Output** in the right hand column of the screen; similar to figure 8.

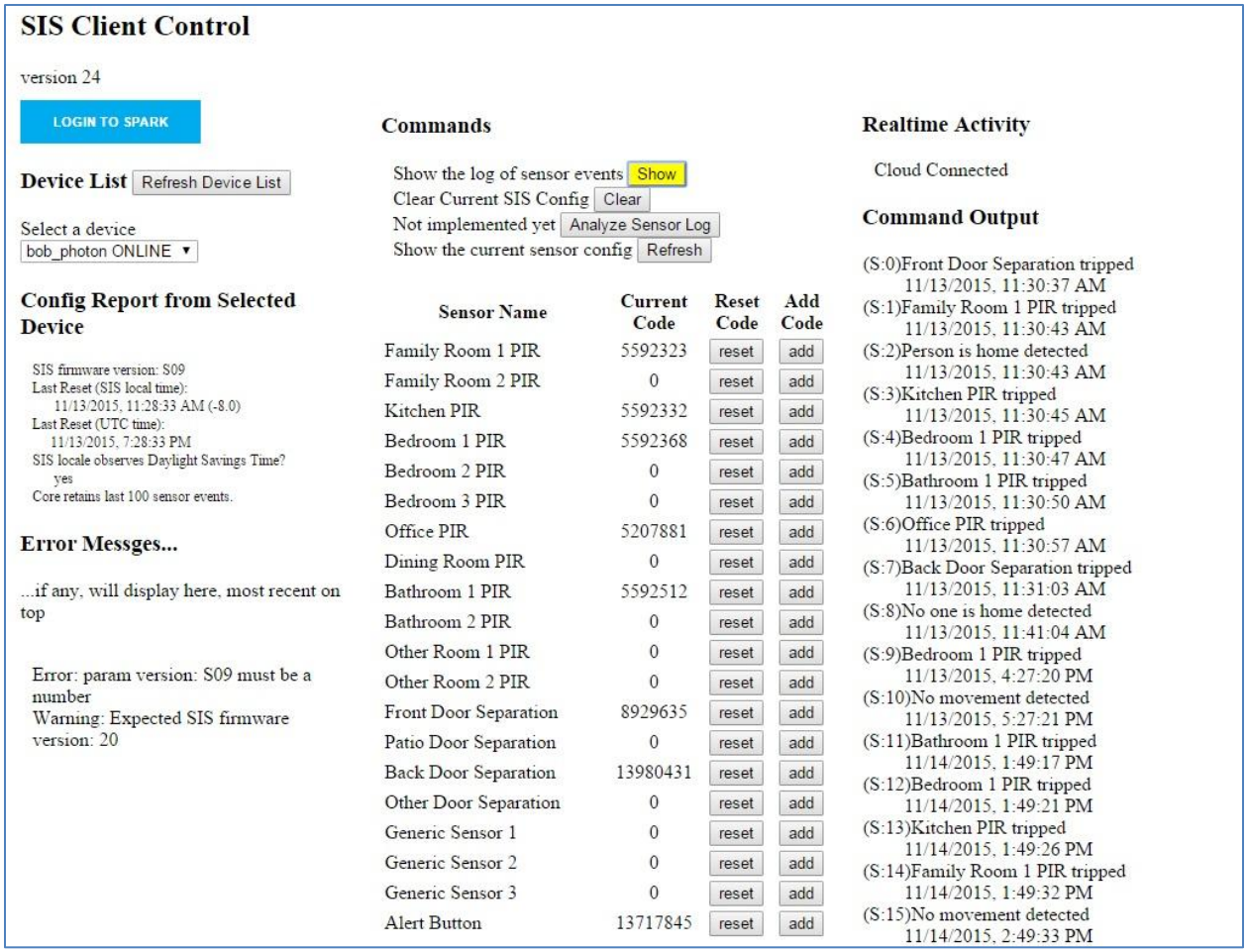

Figure 8. Displaying the SIS Log.

The log entries are displayed below the **Command Output** with the most recent log entry shown first (at the bottom) and the last (oldest) entry in the log shown at the top. Note that it may take about one second per log entry to populate this display. Note too that the log will display the last (latest) 100 entries (or fewer, if there are fewer than 100 entries in the SIS Hub log).

The log entries are formatted to show a one-up "sequence number", followed by a message about the log entry, followed by a time stamp of when the log entry occurred. For example:

(S:3)Kitchen PIR tripped
11/13/2015, 11:30:45 AM

The "(S:3)" is the sequence number of the log entry. Sequence numbers begin at 0 and increment sequentially up to 9999, and then roll back to 0. Sequence numbers are useful in determining if an entry has been missed and also in tracking the last log readout to determine which entries are new. If the SIS Hub is reset (e.g. due to power failure), the sequence numbers will reset to zero. Thus, if the last sequence number that you read out from the log was (S:1534)

and now you read out the log and it begins with (S:0), that means that the SIS Hub was reset in between your last readout and the present readout of the log. You should consult the time of last reset in the left hand column of the display in order to understand what times may have been missed due to the resetting of the SIS Hub.

When using the SIS Client app to read the log out from the SIS Hub directly (in real time), the sequence numbers should be sequential with no gaps (no missing numbers). If you use IFTTT to send log entries to a Google spreadsheet in the cloud (section 6, below), there may be missing sequence numbers, indicating that there was some Internet communication problem when the log entry was published to the Internet and that the data was lost. The IFTTT mechanism has a number of advantages over reading out the SIS Hub internal log in real time; however, it can be unreliable in the face of Internet communication issues. A gap in the sequence numbers will indicate that some entries are missing from the Google spreadsheet; however, the internal SIS Hub log should have no gaps and can therefore fill in any missing information.

After the sequence number comes the message; in this example "Kitchen PIR tripped". This is an example of a logged *event*, as described in the SIS Theory of Operation and Installation Guide. After the message, on the next line, is the time stamp of the message (in this example "11/13/2015, 11:30:45 AM").

The time stamp requires further discussion. When the SIS Hub logs a *sensor trip*, an *event*, or an *inference*, a time stamp is applied. The SIS Hub software uses Internet time (in Universal Coordinated Time -- UTC) for the time stamp. The SIS Client app converts UTC to the actual time where the SIS Hub is located using the configuration parameter "utcOffset". You can see the utcOffset in the entry:

"Last Reset (SIS local time):
11/13/2015, 11:28:33 AM (-8.0)"

This entry is found on the left side of the screen (figure 8), under **Config Report from Selected Device.** Note that the "(-8.0)" is the currently configured utc offset, and is used to convert UTC to local time in the log displayed by the SIS Client app.

At present, the SIS Client app does not support changing the utcOffset from the default "-8.0" value (Pacific Standard Time). If you need to change this default (e.g. to -5.0 for US Eastern Standard Time), you may do so using the SIS Debug Client. The procedure to do this is described in section 7 of this document. You should only have to do this once per SIS installation.

At present, the SIS Client does <u>not</u> have capability to adjust the displayed log time stamp for Daylight Savings Time. The SIS Hub configuration does have a parameter for whether the location observes DST or not, but the SIS Client does not currently make the DST adjustment when the log is displayed. Therefore, the user must make the time adjustment manually, if needed.

5. USING IFTTT TO BE NOTIFED WHEN THE ALERT SENSOR IS TRIPPED.

SIS supports registration of one sensor or wireless "button" (e.g. keyfob) as an "Alarm/Alert" sensor. Tripping this sensor (pressing the button) causes the SIS Hub to log the action as a *sensor trip,* in the same manner as other "misc" sensors. However, tripping this sensor or button also sends a special message out over the Internet to the *particle.io* cloud. It is possible to use a free web service called "IFTTT" to pick up this message and use it to notify someone of the event in near real time. This "notification" can be anything that IFTTT can generate. One specific example is to send an SMS text message to a registered (to IFTTT) mobile phone.

Note: IFTTT response to a cloud publication in this manner is not a reliable service. There can be considerable delay in delivery of the message and it is possible, under various circumstances outside of the SIS' control, to have the notification dropped altogether. Therefore, this capability *should be used as a convenience only* and ***not relied upon for emergency services***.

In order to use this capability you must register at:

https://ifttt.com

You must then create a "recipe", as described below.

These instructions apply strictly to sending a text message to your mobile phone. However, feel free to explore all of IFTTT's many channels for other things that you can do in response to the Alert message from SIS.

Once you register with IFTTT, you can create an "IF" recipe. When you click on "Create a Recipe", you will be presented with a message in large type saying "If This than That". In IFTTT, you select "This" to create the *trigger* (in this case, the cloud message from your SIS Hub), and "That" to create the *action* (in this case, to send an SMS text message to your mobile phone).

When you get the first "If This than That" message, "This" will be in underlined blue, indicating that you need to specify the *trigger*. Click on "This" and it will say "Choose a channel". The channel that you need to choose is "Particle". You will then be prompted to "Choose a trigger". You will select "New event published". Note that if you have not previously registered the Particle channel to your IFTTT account, you will have to do so first – follow the prompts from IFTTT to put your *particle.io* account information into IFTTT.

Once you have registered your *particle.io* account to IFTTT and selected the "New event published" trigger, you will be presented with a number of fields to fill in:

- IF (Event Name): type in "SISAlarm". Type this in exactly as shown here, without the quotes.
- Is (Event Contents); leave this field blank.

- Device Name or ID: select the device registered to your *particple.io* account that you want this IFTTT recipe to apply to (the device name for your SIS Hub).

After completing the "This" part of the recipe, IFTTT will again present you with "If This than That" in large letters and with "That" underlined in blue. If you want the *action* to send you an SMS text message, you select the "sms" channel (the IFTTT native SMS channel). Once again, you will have to register your mobile information to this channel if you had not previously subscribed to it. You select the *action* entitled "Send me an SMS". IFTTT will present you with a default message to send to your mobile phone, which you can edit or leave as is (your choice).

Finally, IFTTT will assign you a recipe name, which you can leave as is.

IFTTT provides you with tools to enable or disable your recipes, edit your recipes, and delete recipes. See the IFTTT on-line documentation for more information. You can test this new recipe by triggering the registered Alarm sensor and seeing that you get the SMS message that you specified. Note that it may take some time – a minute or more – before the message comes though. If, at any time in the future, you wish to disable getting these messages, you can turn the recipe off in your IFTTT account.

6. USING IFTTT TO LOG TO A GOOGLE SPREADSHEET IN THE CLOUD.

Every time that the SIS Hub adds an entry into its internal log, it also publishes a message to the *particle.io* cloud. You can use the free service IFTTT (https://ifttt.com) to enter these log messages into a Google spreadsheet. In order to do this, you also need a free Google account.

Note: IFTTT response to a cloud publication in this manner is not a reliable service. There can be considerable delay in delivery of the message and it is possible, under various circumstances outside of the SIS' control, to have the logging dropped altogether.

The advantages to using a Google spreadsheet to create a duplicate log in the cloud are (1) the log is persistent and won't disappear with an SIS Hub reset or power failure, and (2) the log is much longer – it is 2000 entries per spreadsheet and when one spreadsheet fills, a new one is automatically created. The internal SIS log is 100 entries long and overwrites the oldest entries with newer ones once 100 entries have been received.

The disadvantages to using a Google spreadsheet for a cloud-based log are: (1) it may take some time (several minutes or more) for an entry to appear in the Google spreadsheet and (2) some entries may be completely missing, owing to Internet related communication events outside of SIS' control. Note that a missing entry can easily be determined by a gap in the sequence numbers. There will be no gap in sequence numbers in the internal SIS log, which can be read out in real time as described in section 4, above. It is thereby possible to use both logs – the SIS internal log for timeliness and completeness and the Google spreadsheet log for long term data retention.

In order to obtain a log in a Google spreadsheet in your Google account, you need to have a Google account (free) and also an IFTTT account. If you don't have an IFTTT account, proceed as described in section 5, above.

After creating both your Google account and your IFTTT account, proceed in IFTTT to create a new "IF" recipe, similarly to section 5. The "This" channel will again be the "Particle" IFTTT channel and the trigger will (again) be "New event published". You will be presented with a number of fields to fill in:

- IF (Event Name): type in "LogEntry". Type this in exactly as shown here, without the quotes.
- Is (Event Contents); leave this field blank.
- Device Name or ID: select the device registered to your particple.io account that you want this IFTTT recipe to apply to (the device name for your SIS Hub).

After completing the "This" part of the recipe, IFTTT will again present you with "If This than That" in large letters and with "That" underlined in blue. You want the channel to be the "Google Drive" IFTTT channel. If you have not previously registered this channel to your IFTTT account, IFTTT will prompt you to do so. After registering the Google Drive channel, the IFTTT *action* you select will be "Add row to spreadsheet". IFTTT will prompt you to fill in the following fields:

- Spreadsheet name: You can name this anything that you want, such as "Grandma's activity log".
- Formatted row: You can leave the defaults here or change them. The defaults create 4 columns in the spreadsheet with the event name ("LogEntry"), Event Contents (the message, which is similar to the contents of the SIS Hub's internal log), your chosen device name, and a Google generated time stamp (which is in the timezone that Google recognized you to be in – not necessarily the timezone where the SIS is located!).
- Drive folder path: You can specify a folder path on your Google Drive to put the spreadsheet in or just leave this field blank (in which case, IFTTT will place a folder called "IFTTT" on your Google Drive and put the spreadsheet inside of this folder).

Finally, IFTTT will assign you a recipe name, which you can leave as is.

A typical log entry on your Google Spreadsheet will look something like this:

LogEntry	(S:3)Kitchen PIR tripped at Fri Nov 13 19:30:45 2015 Z (epoch:1447443045Z)	bob_photon	November 13, 2015 at 11:30AM

Note that the logged message has the SIS time stamp in UTC but that Google's time stamp is in local time (which will be corrected for daylight savings time, by Google).

IFTTT provides you with tools to enable or disable your recipes, edit your recipes, or delete recipes. See the IFTTT on-line documentation for more information. You can test this new recipe by triggering some registered sensors and then looking at the entries in the Google Spreadsheet on your Google Drive. Note that it may take some time – perhaps several minutes – before the spreadsheet entries appear. If, at any time in the future, you wish to disable logging entries to your Google Spreadsheet, you can turn the recipe off in your IFTTT account.

7. USING THE DEBUG CLIENT FOR ADVANCED OPERATIONS.

The SIS Client app currently has certain limitations:

- It cannot be used to set the time zone offset (from UTC) of the location where the SIS Hub is installed. The default time zone offset is -8.0, which corresponds to Pacific Standard Time. As a result, the time stamps for SIS log entries will be displayed in Pacific Standard Time. If you wish to change this, you must use the Debug Client to do so, as explained below.

- The SIS Hub defaults to a configuration entry that says that the location of the SIS Hub observes Daylight Savings Time. The SIS Client does not presently adjust for Daylight Savings Time; however, the Debug Client may be used to change the SIS Hub configuration to indicate that it is in a location that does not observer Daylight Savings Time.

- The Debug Client may be used to obtain detailed information from the SIS Hub that is not available using the SIS Client app. It may also be used to register sensors to any location (any location name) and is not limited to the default rooms/locations shown on the SIS Client display. A full description if the Debug Client is beyond the scope of this document and changing sensor registration locations is not described herein.

To change the time zone offset configuration parameter in the SIS Debug Client, you must first load the client from:

http://shrimpware.com/SIS/DebugSIS.html

You must then log into the particle.io cloud via the Debug Client in exactly the same way that you logged into the SIS Client app. Likewise, you select the SIS Hub device in exactly the same way that you did using the SIS Client app. At this point, you will see a screen similar to figure 9, below:

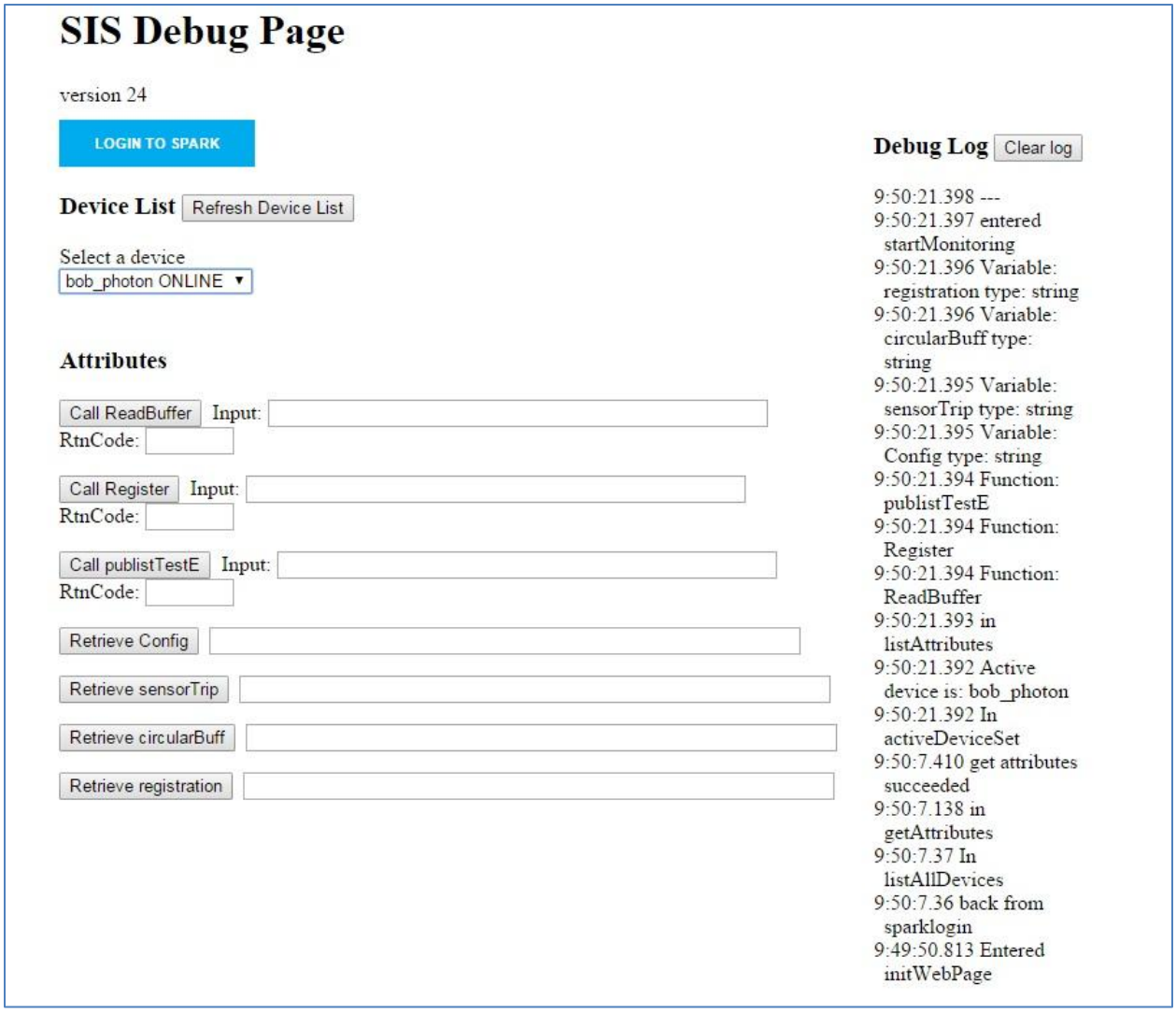

Figure 9. Debug Screen.

Referring to figure 9: there are a series of command buttons under **Attributes** on the left hand side of the screen. Click on the button **Retrieve Config** to retrieve the configuration from the selected SIS Hub device. You will then see something similar to figure 10.

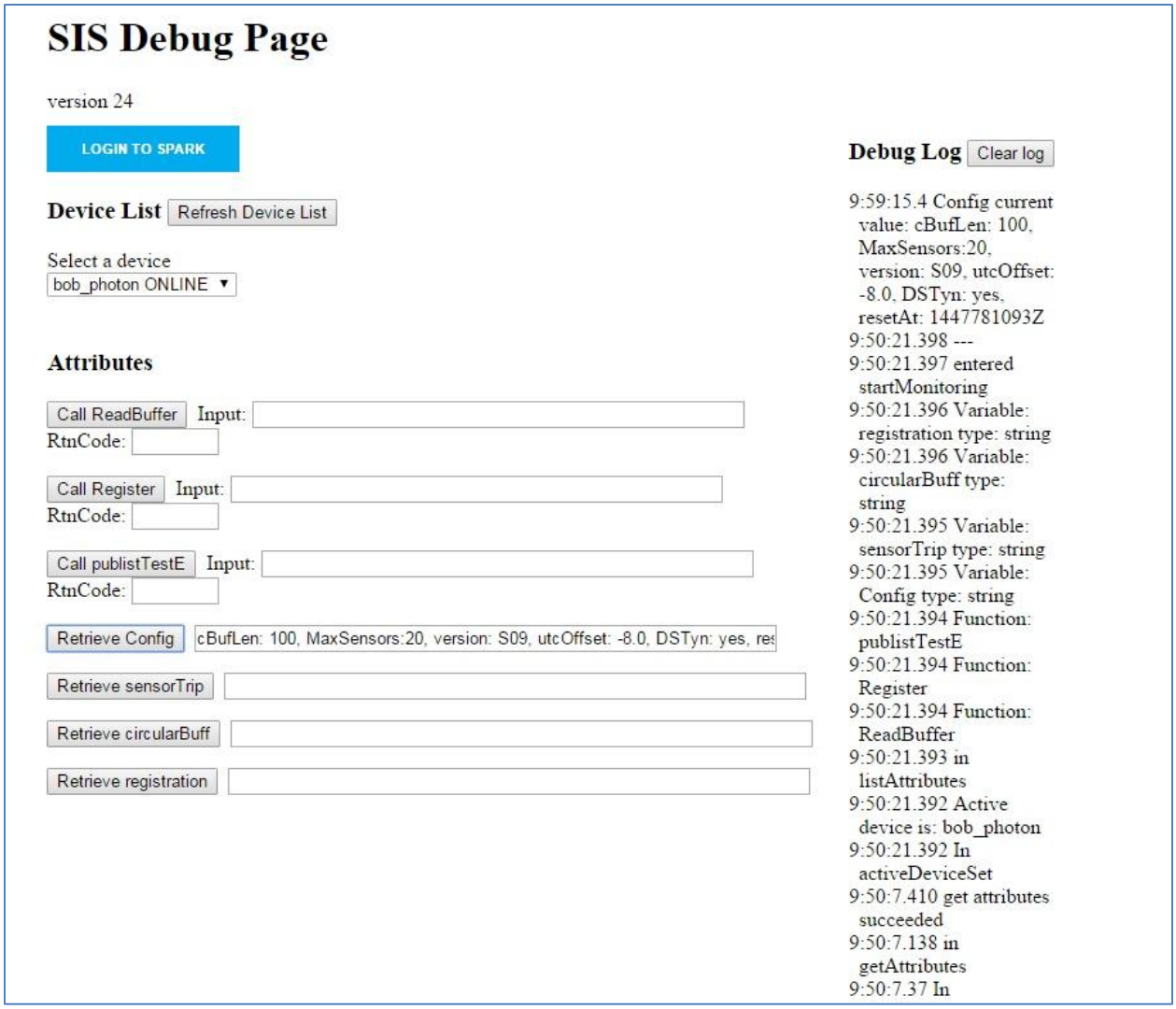

Figure 10. Configuration Information from the SIS Hub.

Referring to figure 10, you can see information in the text box next to the **Retrieve Config** button. Examining this text box you can observe the "utcOffset" is set to "-8.0" and that "DSTyn" is set to "yes".

These configuration parameters can be changed using the **Call Register** button and the text box directly to the right of it. To change the utcOffset parameter to Eastern Standard Time (offset = -5.0), type the following into this text box:

offset,-5.0

Make sure to type this exactly as shown above; no spaces, all lower case, and you must put the offset in as a number with a decimal point. Do not type the "enter" button, just the text, as shown. After typing this text in the text box, click the **Call Register** button. You will see something similar to figure 11.

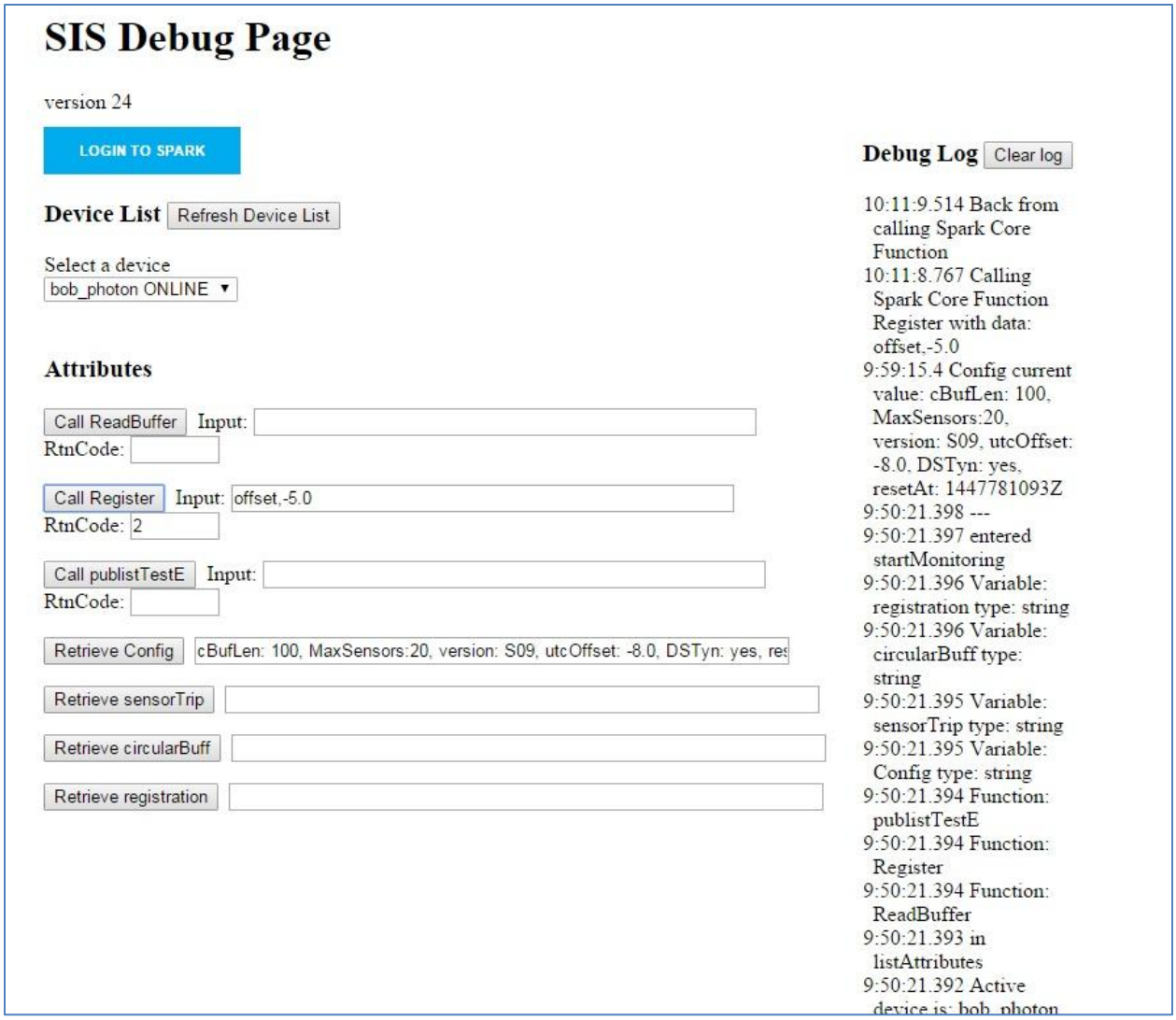

Figure 11. Changing the UTC Offset.

A "RtnCode" of "2" indicates success. In order to verify this, click on **Retrieve Config** again. You should see something similar to figure 12.

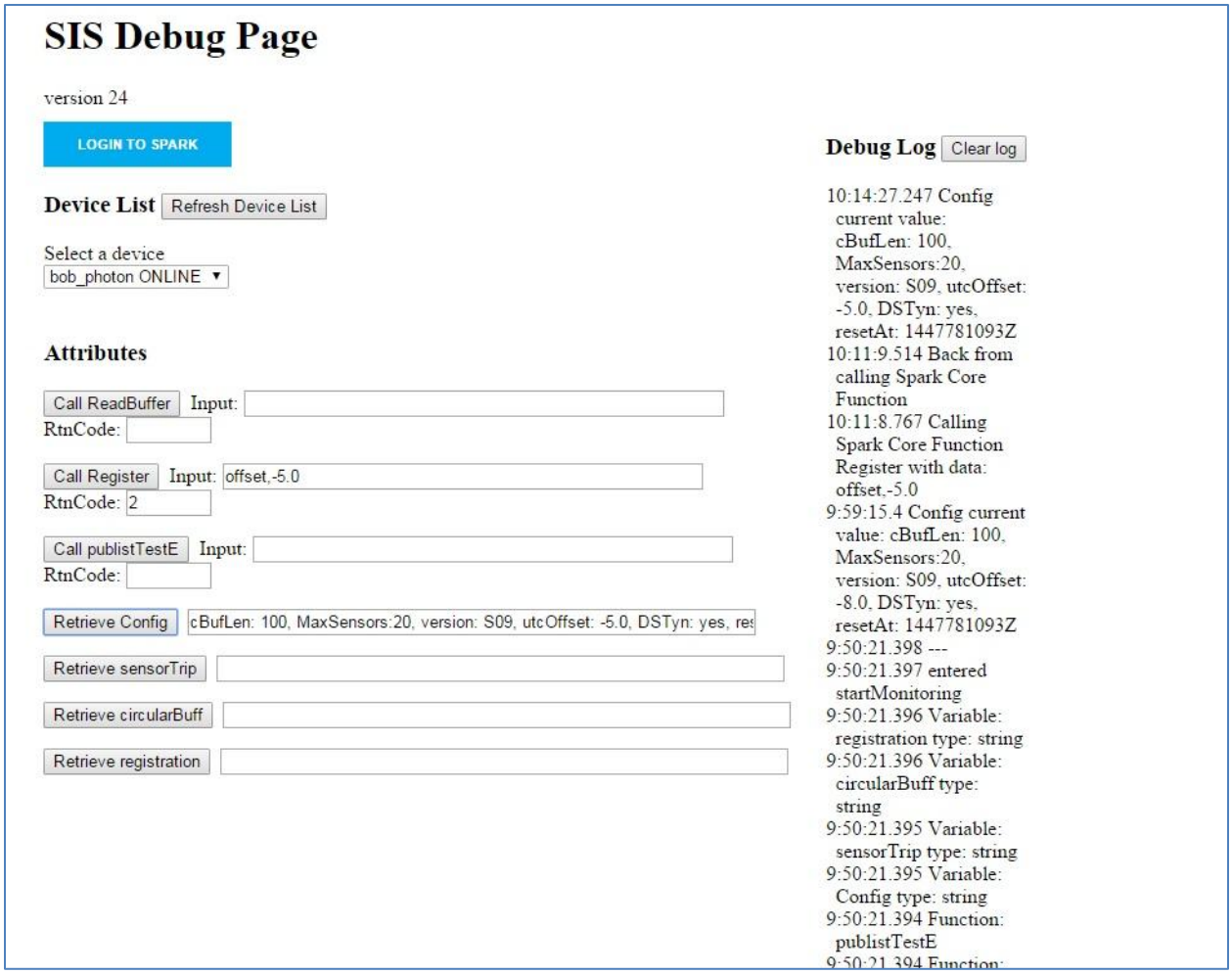

Figure 12. Verifying the UTC Offset Change.

You can see that the UTC Offset has been changed to -5.0. However, this change is not yet stored in non-volatile memory, so the next time that the SIS Hub is reset, the old UTC Offset value will come back. If you want to store this new configuration persistently, replace the "Input" next to **Call Regist**er with the word:

store

Again, you must type this exactly as shown – all lower case and nothing else on the text box. You do not type the enter key. You click on **Call Register** and a "RtnCode" of "1" indicates success. See figure 13.

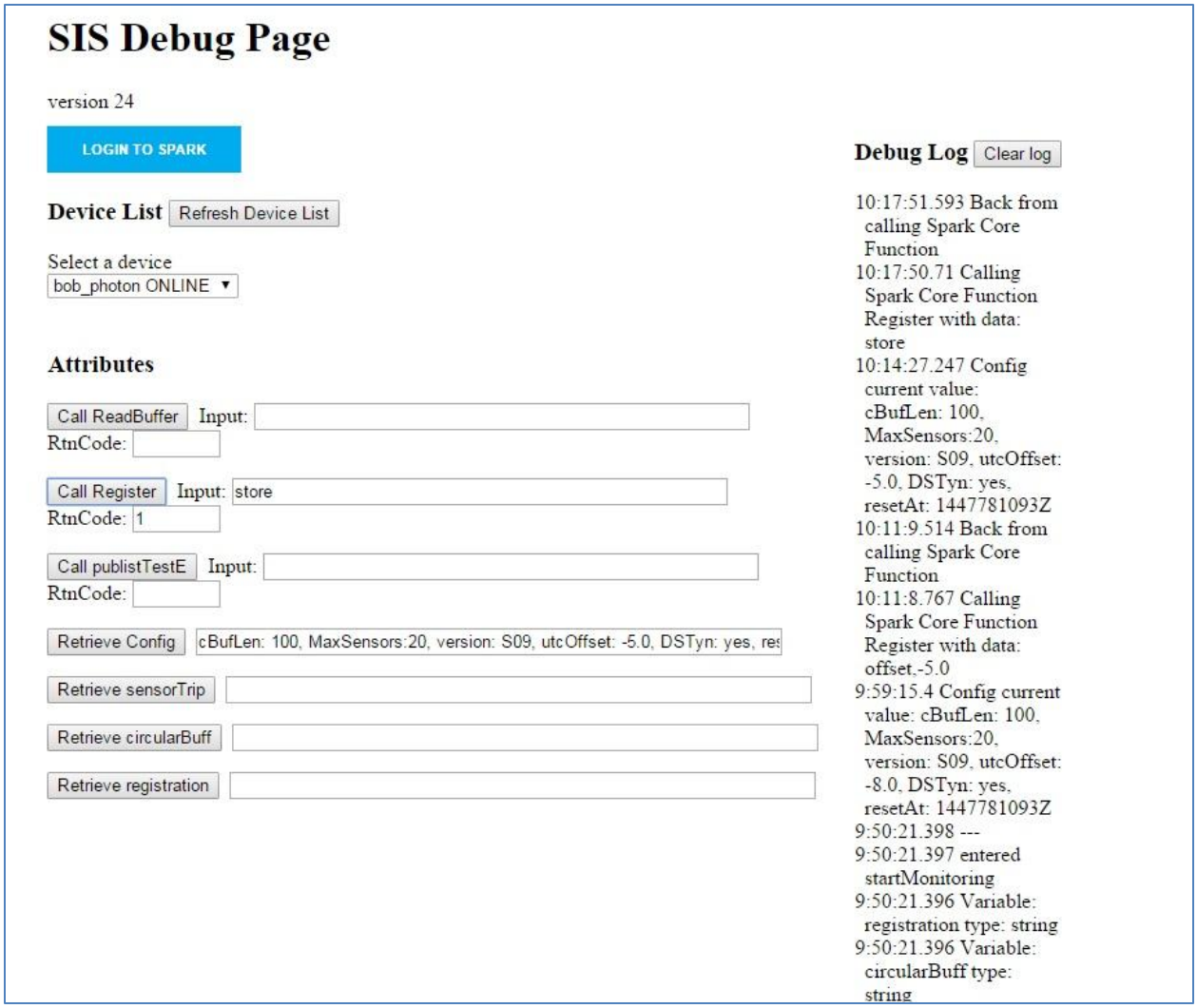

Figure 13. Storing the New Configuration in Non-Volatile Memory.

If you want to change the DST parameter (e.g. from "yes" to "no"), type the following into the "Input" text box adjacent to **Call Register**:

DST,n

Type this exactly as shown, overwriting anything else in the text box. Do not type the enter key. Click on **Call Register**. A "RtnCode" of "2" indicates success. You can then click on **Retrieve Config** to verify the changed configuration, as shown in figure 14.

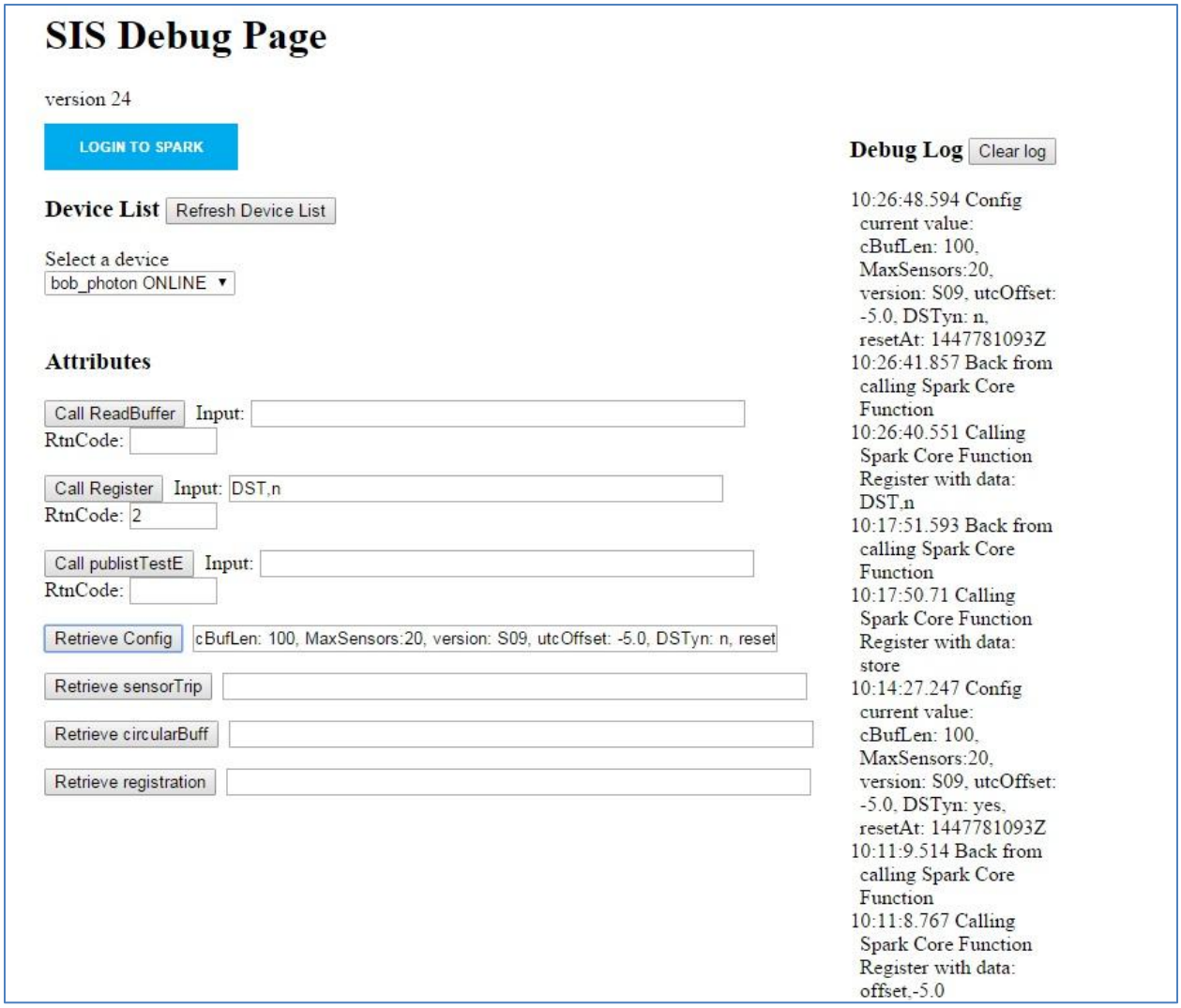

Figure 14. Changing the DST Configuration.

Very Important: if you wish this new configuration to be stored in non-volatile memory, you must "store" it, as described above. Each time that you "store" the configuration, you are storing the entire configuration of the SIS Hub into non-volatile memory. Therefore, you may make multiple configuration changes (e.g. change the "utcOffset" and the "DST") and then "store" it once after verifying all of your changes in the **Retrieve Config** text box.

The SIS Debug Client can be used to exercise all low level communication to/from the SIS Hub, including reading out the circular buffer, retrieving the latest sensor trip (registered or unregistered), changing/adding/deleting sensor registration, and logging all raw communication in the **Debug Log**. Instructions to perform these additional functions are beyond the scope of this document.