xapmcsMultimeter

Radio Shack 220-0812 xAP Interface

Digital Multi-Meter Telemetry

Michael McSharry
Feb 18, 2010

Table of Contents

1 Introduction 3
2 xapmcsMultimeter Setup 3
3 Operation 4
4 Browser Pages 5
4.1 xAP Data / Setup 5
4.1.1 Hide Reading 7
4.1.2 Accept Reading 7
4.1.3 Status 7
4.1.4 ID 7
4.1.5 Source 7
4.1.6 Changed 7
4.1.7 Setup Settings 7
4.2 Messages Page 9
4.3 Serial IO 9
5 xAP Setup 10
5.1 Background 10
5.2 mcsXap Installation 11
5.3 mcsXap Initial Setup 12
5.4 mcsXap Initial Use 14
5.5 xapmcsHub Setup 17
5.6 xAP Message Viewer 18
6 xAP Schema 22
6.1 xapmcsMultimeter 22
6.1.1 Receive xAP Schema 22
6.1.2 Transmit xAP Schema 22

Table of Figures

Figure 1 xapmcsMultimeter Tray Icon 4
Figure 2 Browser Header and Links 4
Figure 3 Multimeter Reading Viewing 6
Figure 4 Setup Settings 8
Figure 5 Homseer Setup Interfaces Page 12
Figure 6 mcsXap Primary Setup Options 13
Figure 7 Message Selection Filters 17
Figure 8 xAP Hub Activity Form 18
Figure 9 xAP Viewer 20
Figure 10 xAP Viewer Message Detail 21

Table of Tables

None

1 Introduction

xapmcsMultimeter provides an interface to the Radio Shack 220-0812 digital multimeter to provide telemetry of the multimeter readings in the form of LAN messages using xAP protocol. The multimeter provides 26 measurement modes (e.g. AC voltage, DC voltage, ohms, etc.). Each of these, when selected on the meter, is transmitted as the raw reading, the rate of change of the reading, the maximum reading and the minimum reading. The content of these messages can be recorded in a database such as is supported by xapmcsDatabase, or accepted in other automation applications such as Homeseer where event logic can be used to trigger other actions.

xapmcsMultimeter will sense the user selections made on the multimeter and automatically report the measurement being made. It will identify the type of measurement, include measurement units, and scale the value as the meter auto-ranges from pico, micro, kilo, and mega. If a capacitance measurement is being made it will be reported in Farad units with values typically less than 1.0 (e.g. 0.0000000123). If a resistance measurement is being made it will report in ohms with values typically greater than 1.0 (e.g. 123456.7)

xapmcsMultimeter will compute the minimum and maximum readings unless the meter is setup to report only maximum or minimum readings. In this case the maximum and minimum reported by xapmcsMultimeter will be maximum or minimum from the meter. When the multimeter is in this mode xapmcsMultimeter will not report a current reading or a rate of change of the reading.

2 xapmcsMultimeter Setup

The contents of the zip package should be unzipped into a convenient location such as C:\Program Files\xAP\xapmcsMultimeter\. The executable program is xapmcsMultimeter.exe. It is a .NET application and provides no ActiveX/COM interface. It is interfaced to other automation via xAP messages using the xAPBSC schema.

The install will deposit a cascading style sheet in the \HTML subfolder. This can be edited, if desired, to achieve a browser display that fits better with other applications. At the same location there will also be an .htm file that contains clickable links that will be shown at the top of the browser display to provide quick access to other applications. This will need to be edited to reflect any applications that are appropriate for your environment.

xapmcsMultimeter is started by running xapmcsMultimeter.exe. It will present itself as a tray icon such as is shown in Figure 1. Clicking on the Icon will bring up a menu from which various browser pages can be selected. The icon will blink as new images are created or updated.

Figure 1 xapmcsMultimeter Tray Icon

The tray icon, when left-clicked will make available four options. The first, xAP Data/Setup, is the page where setup options are selected and DMM readings selected for transmission. The second, Serial IO, is a view into the raw RS-232 data from the multimeter. The third, Messages, contains log messages from xapmcsMultimeter. The last page is the help page which will show this manual.

The top of each browser page contains a title, links to access other locations and links to access subsets of xapmcsMultimeter pages that are available from the tray icon. This is shown in Figure 2. The application links are user-defined in a file specified on the Setup page (Alias/Setup tray icon). The xapmcsMultimeter links are built based upon the number of RSS feeds that have been defined. In a normal session the tray icon will be used to gain initial access to any of the browser pages then the links used to select the specific segment that is being edited.

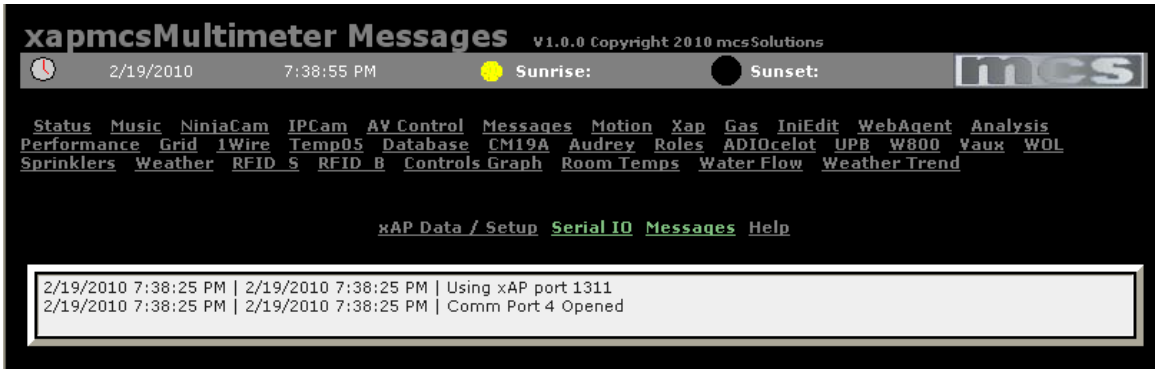


Figure 2 Browser Header and Links

3 Operation

An RS-232 connection is made between the computer and the multimeter. The selected port is identified on the setup page. This can be a COM port number or an IP address for IP/Serial raw data communications. The meter communicates at 4800 baud, 8 bits, no parity.

The meter probes will be connected to the entity being measured. The meter mode selection will be made so the desire measurement is selected. xapmcsMulitmeter will recognize which mode the meter is operating and provide the data in a manner that will associate the mode with the readings.

The xAPBSC schema is used for the xAP protocol. The subaddress is encoded as X.Y.Z where X is DMM; Y is the mode and type of data; Z is the measurement unit. The Y component is 16 characters starting with the name of the mode. The middle contains a

numeric equivalent of the mode. The right will be one of the four types of data (Reading, Rate, Max, or Min).

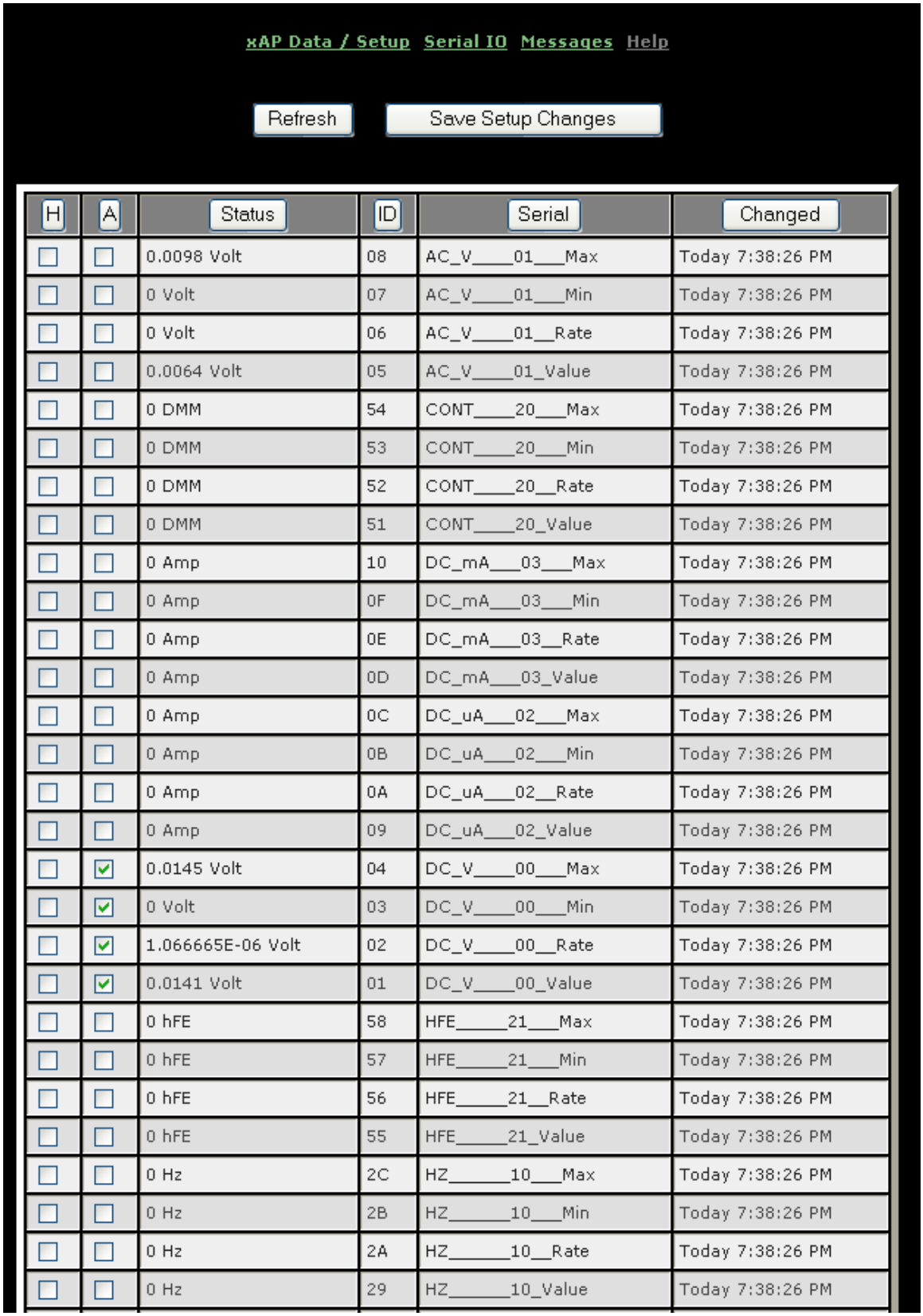
Any change in the value, or derived value from the meter will be transmitted using xapbsc.event message. The max and min will be computed until a xapbsc.cmd message is received with an address that resolves to the max or min address/subaddress. This reset of max or min can be wildcarded so multiple resets can be done with a single xapbsc.cmd message. The last reading of any end point can be obtained using the xapbsc.query message request.

4 Browser Pages

4.1 xAP Data / Setup

The measurement values are shown on the top of the page and the setup parameters are at the bottom of the page.

At the top, measurement data is presented in six sortable columns as shown in Figure 3. The user has two selections that can be made on each row.



H	A	Status	ID	Serial	Changed
☐	☐	0.0098 Volt	08	AC_V____01___Max	Today 7:38:26 PM
☐	☐	0 Volt	07	AC_V____01___Min	Today 7:38:26 PM
☐	☐	0 Volt	06	AC_V____01__Rate	Today 7:38:26 PM
☐	☐	0.0064 Volt	05	AC_V____01_Value	Today 7:38:26 PM
☐	☐	0 DMM	54	CONT____20___Max	Today 7:38:26 PM
☐	☐	0 DMM	53	CONT____20___Min	Today 7:38:26 PM
☐	☐	0 DMM	52	CONT____20__Rate	Today 7:38:26 PM
☐	☐	0 DMM	51	CONT____20_Value	Today 7:38:26 PM
☐	☐	0 Amp	10	DC_mA___03___Max	Today 7:38:26 PM
☐	☐	0 Amp	0F	DC_mA___03___Min	Today 7:38:26 PM
☐	☐	0 Amp	0E	DC_mA___03__Rate	Today 7:38:26 PM
☐	☐	0 Amp	0D	DC_mA___03_Value	Today 7:38:26 PM
☐	☐	0 Amp	0C	DC_uA___02___Max	Today 7:38:26 PM
☐	☐	0 Amp	0B	DC_uA___02___Min	Today 7:38:26 PM
☐	☐	0 Amp	0A	DC_uA___02__Rate	Today 7:38:26 PM
☐	☐	0 Amp	09	DC_uA___02_Value	Today 7:38:26 PM
☐	☑	0.0145 Volt	04	DC_V____00___Max	Today 7:38:26 PM
☐	☑	0 Volt	03	DC_V____00___Min	Today 7:38:26 PM
☐	☑	1.066665E-06 Volt	02	DC_V____00__Rate	Today 7:38:26 PM
☐	☑	0.0141 Volt	01	DC_V____00_Value	Today 7:38:26 PM
☐	☐	0 hFE	58	HFE_____21___Max	Today 7:38:26 PM
☐	☐	0 hFE	57	HFE_____21___Min	Today 7:38:26 PM
☐	☐	0 hFE	56	HFE_____21__Rate	Today 7:38:26 PM
☐	☐	0 hFE	55	HFE_____21_Value	Today 7:38:26 PM
☐	☐	0 Hz	2C	HZ______10___Max	Today 7:38:26 PM
☐	☐	0 Hz	2B	HZ______10___Min	Today 7:38:26 PM
☐	☐	0 Hz	2A	HZ______10__Rate	Today 7:38:26 PM
☐	☐	0 Hz	29	HZ______10_Value	Today 7:38:26 PM

Figure 3 Multimeter Reading Viewing

4.1.1 Hide Reading

The “H” checkbox is used to hide the row from the browser view. A setting is available at the bottom of the page where hidden values can be forced to be shown.

4.1.2 Accept Reading

Data from the meter will be visible on the browser page but will not be transmitted on the LAN until the reading has been accepted with the “A” column checkbox.

4.1.3 Status

The status is the most recent reading from the meter or derived from a meter reading. The source column identifies the reading associated with the value

4.1.4 ID

The ID column is the UID for xap end point. It is a unique value for each of the 26*4 possible multimeter end point readings.

4.1.5 Source

The source identifies the type of measurement. The first part of the name identifies the multimeter mode. The end of the name identifies the type of measurement where “Value” is the latest reading, “Rate” is the rate of change of this reading, and “Min” and “Max” are latched values. Note that the Min and Max behaves differently based upon if the multimeter is configured to report only min and max or to report only current readings.

4.1.6 Changed

The changed column identifies the time the value reported from the multimeter last changed. It does not reflect the last time a reading was reported.

4.1.7 Setup Settings

The setup setting options are shown in Figure 4.

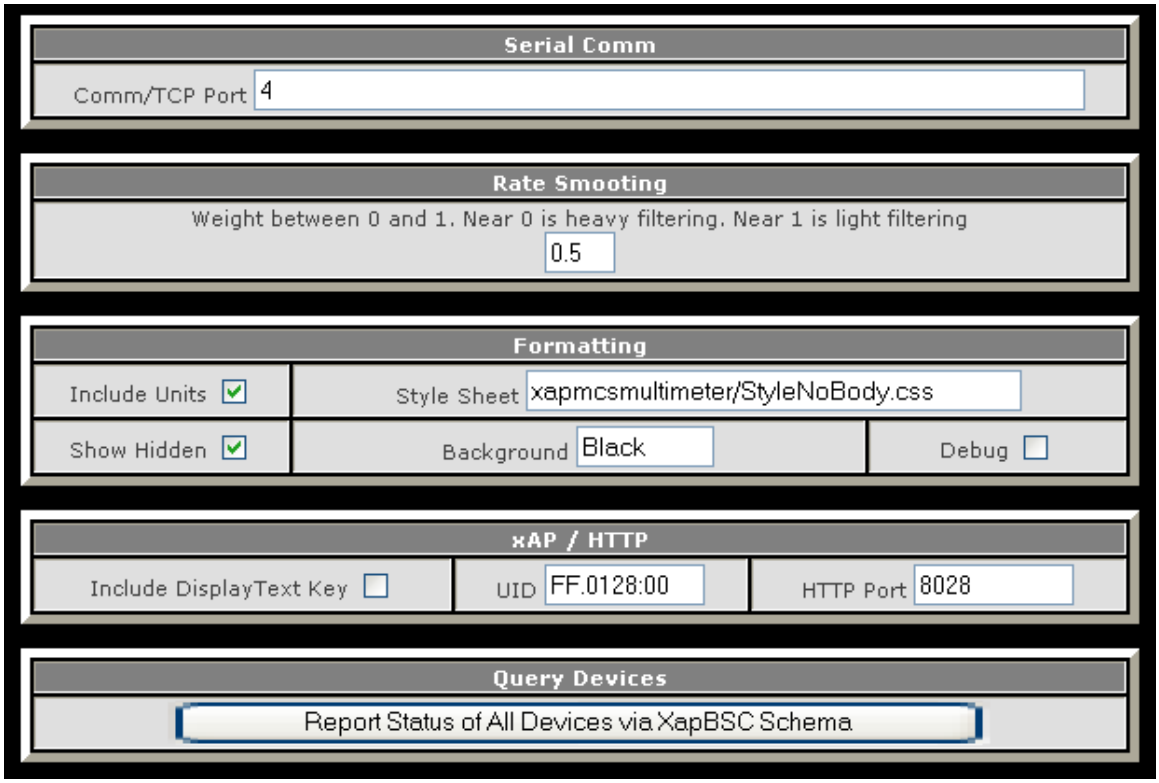


Figure 4 Setup Settings

Serial Comm

The serial connection between the multimeter and the PC can be specified as either an IP/Internet address or a COM port number on the local computer. If the TCP port is selected then both the IP address and port should be entered such as 192.168.0.40:3001 or http://myhouse.com:5000.

xapmcsMultimeter will monitor the serial port connection. If no data received from the multimeter in 60 seconds then it will try to reopen/reconnect. This is primarily a feature for TCP connection that may be lost due to change in IP or other disturbance.

Rate Smoothing

Rate smoothing is used in the rate calculation for each multimeter measurement. The sampling rate from the multimeter is rather high so very small changes in the reading will show as high rate of change unless filtering is used. The filter is specified as a weight in the range 0 to 1 where near 0 is very heavy filtering and near 1 is very light filtering. The default is 0.5.

Formatting

Formatting provides setting to control how the data appears on the browser. The Include Units checkbox will suffix a multimeter reading with the appropriate units. The Show Hidden checkbox overrides the individual Hide settings for individual meter reading data. The browser style sheet provides overall appearance with all style settings except the background color with is separately identified. The debug checkbox is used to produce additional logic output to the \data subfolder in a .txt file.

XAP / HTTP
This section allows the port and identification to be customized. Each xAP application must have a unique ID of the format xx.xxxx.xxyy. The yy aspect of this is appended with individual end points within xapmcsmultimeter. The xxx segment is user specified.

The HTTP, if changed will lose connection with the current session. If the default 8028 is not available then \Config\xapmcsMultimeter.ini needs to be edited to select an available port.

Query Devices
When the button is clicked all accepted measurement values will be delivered via xapbsc.info messages. It is equivalent to an external xAP node issuing a wildcarded xapbsc.query to xapmcsMultimeter.

This button can be useful if trying to initially accept data from xapmcsMultimeter into another node such as xapmcsDatabase or mcsXap.

4.2 Messages Page

The Messages Page is shown in Figure 2. It will contain informative information and errors raised by xapmcsMultimeter. The messages log is created new for each restart of the application. The source data for this page is available in the \Data folder.

4.3 Serial IO

The Serial IO page shows a line for each set of eight bytes received from the multimeter. The actual data is a continuous stream of nine packets where synchronization is accomplished by finding a pattern of eight followed by a checksum that match. The checksum byte is not shown on the Serial IO Page

5 xAP Setup

This section describes the initial setup of a xAP environment for Homeseer. It provides a high level description of xAP, the basic setup of the mcsXap plugin, setup of a xAP hub using the xapmcsHub application and a brief introduction to the xAP Viewer.

xapmcsMultimeter does not depend upon Homeseer, but if information from Homeseer is desired to be used in the constructed Digital Frame images then the mcsXap plugin is necessary.

If xAP message data is to be used in the construction of Digital Frame images then a xAP hub is needed on the same computer that xapmcsMultimeter is installed. If xAP data will not be used as part of the image construction then a xAP hub is not required.

5.1 Background

xAP protocol is implemented as UDP datagrams and this provides some benefits and some shortcomings. UDP datagrams have a very small protocol overhead which means they are easy to encode and decode and consume a very small footprint for low overall network utilization. The data is broadcast which means every device on the network receives the same information simultaneously. This makes for a convenient distributed computing environment. xAP is designed so it can operate on the smallest of processors such as the PIC so application in dedicated embedded applications is quite achievable. Hardware devices that natively communicate with xAP are available.

The tradeoff for UDP vs. TCP is that communications are not assured by the protocol. A UDP message is transmitted and there is no acknowledge of its receipt or retry if the transmission fails. Since messages are broadcasted every devices will receive it and process it through its recognition stack to the point of recognizing it is or is not interested in its content. UDP is also intended only for LAN and not WAN communications. It is possible to control remote locations with a TCP communication, but this cannot be done with an UDP one.

The xAP protocol layer on top of UDP does address some of the shortcomings. It provides a schema whereby message receipt acknowledgement is provided. Applications are also available that will wrap xAP messages inside of TCP communications to allow WAN control via xAP. In both of these cases the applications become responsible for communication integrity rather than having this responsibility at the transport layer.

mcsXap is a Homeseer plugin that is able to recognized LAN traffic using the xAP protocol. This protocol is just another one of the many that have been defined (e.g. SMTP, HTTP) and layered on top of the IP network.

The xAP protocol is extensible. There are core schema that provide for maximum interoperability and there are specialized one that target specific niche. mcsXap is able to

recognize all xAP schema, but is only able to translate selected ones into the Homeseer model of devices.

mcsXap will decode and encode xAPBSC, xap-x10 and xap-IR schema which is widely used core schema. xAPBSC provides the ability to communicate Homeseer DeviceStatus, DeviceValue, and DeviceString. xap-x10 provides the ability for additional X10 interfaces to be connected to Homeseer. Xap-IR provides the ability for additional IR interfaces to be connected to Homeseer.

It also support schema that are developed around the mcs environment and includes setio, role, time, wdt, homeseer, voice, and writelog. These special schema have application within a mcs xAP environment, but will not have application to those outside of Homeseer-oriented setups. For example, the homeseer.event schema is used to allow the Homeseer Log to exist on the LAN rather than within Homeseer and all xAP applications can write to this log just as one would write to the Homeseer Log using hs.Writelog.

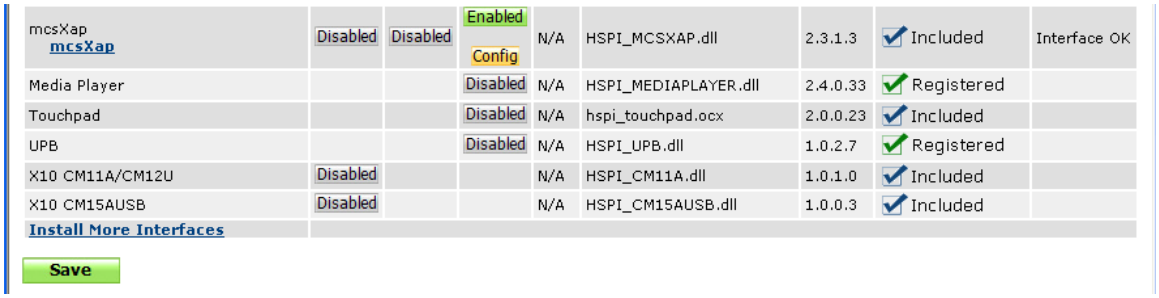
The typical initial use of xAP with Homeseer is to interface with some other xAP application such as xapmcs1wire to get 1-wire sensor readings into Homeseer or xapmcsRF to get W800 or RFXCOM RF translations into Homeseer. In most of these cases the information of interest is communicated using xAPBSC schema and that will be the only schema that needs to be considered within mcsXap.

5.2 mcsXap Installation

mcsXap is provided in two versions. One is targeted to Homeseer V1.x and is an .ocx implemented in VB6. The second is targeted to Homeseer V2.x and is a .dll implemented in VB.NET.

The set of files are obtained from the xAP Library section of the Homeseer Message Board in the form of a zip container. The contents of the zip file should be extracted into the same folder where Homeseer.exe is running. This will be something like C:\Program Files\Homeseer HS2\ but several variants are used.

The next time Homeseer starts it will recognize mcsXap and it will be available to be included as an active Homeseer plugin. Figure 5 shows a segment of the Setup\Interfaces page. mcsXap is shown as the first row. The three columns of buttons indicate that it can be used for IR, X10 and Other. For most users it will only be an Other interface with IR and X10 provided by other plugins. The selections in this figure show it being enabled only as an Other plugin. The Save button must be used to record this intent after the button is used to enable it.



mcsXap mcsXap	Disabled	Disabled	Enabled Config	N/A	HSPI_MCSXAP.dll	2.3.1.3	Included	Interface OK
Media Player			Disabled	N/A	HSPI_MEDIAPLAYER.dll	2.4.0.33	Registered	
Touchpad			Disabled	N/A	hspi_touchpad.ocx	2.0.0.23	Included	
UPB			Disabled	N/A	HSPI_UPB.dll	1.0.2.7	Registered	
X10 CM11A/CM12U	Disabled			N/A	HSPI_CM11A.dll	1.0.1.0	Included	
X10 CM15AUSB	Disabled			N/A	HSPI_CM15AUSB.dll	1.0.0.3	Included	
Install More Interfaces								

Save

Figure 5 Homseer Setup Interfaces Page

Once mcsXap is active as a Homeseer plugin then it needs to be setup to identify the scope of what it is suppose to do. mcsXap is primarily a conduit to translate information into a Homeseer view of it. It is just like the CM11A/CMU12U plugin that translates the X10 protocol on the powerline or the UPB plugin that translates UPB powerline protocol. In the mcsXap case the LAN is the physical connection and xAP is the protocol.

When mcsXap or any mcs xAP application starts it will look for the first NIC and assume that this NIC will be the one to use for xAP communications. If this default is not the correct one then a specific one can be identified by including the desired information in the file \Config\mcsXap.ini. A sample mcsXap.ini showing the setup for the NIC using address 192.168.0.117 is shown below.

[INTERFACE]
InterfaceAddress=192.168.0.117
BroadcastAddress=192.168.0.255

5.3 mcsXap Initial Setup

Figure 6 contains the settings of most interest in getting mcsXap functional. For most users the only xAP schema of interest will be xAPBSC and to enable the Receive BSC and Transmit BSC checkboxes will be checked. Figure 6 shows also the three X10 related options as being selected. Most users will not have these enabled, but no harm is one with them enabled.

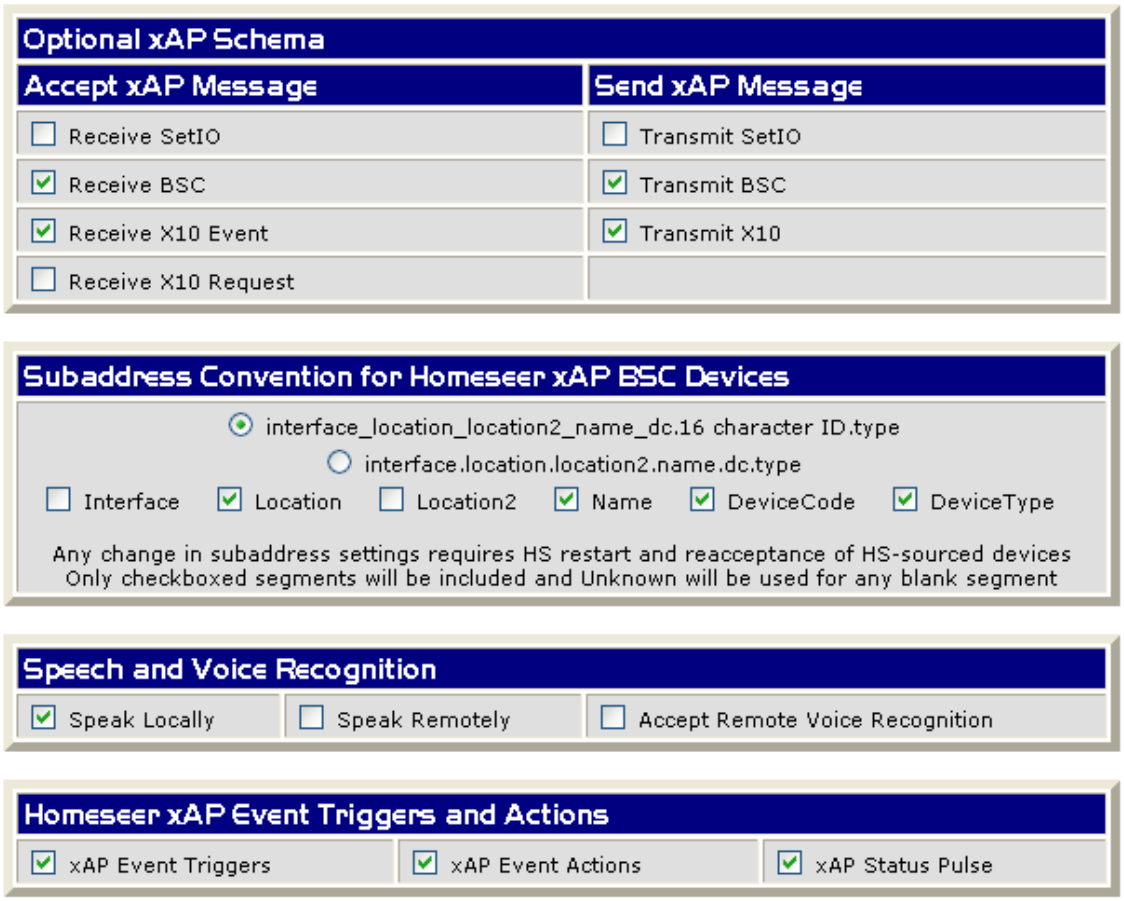


Figure 6 mcsXap Primary Setup Options

The second setup section shown in Figure 6 is used to define how the address of the xAPBSC message is constructed. The address is the mapping between a Homeseer device reference and an xAP message that will be recognized by other xAP applications. In general it does not matter what the address format is, but whatever is selected should not be changed once the xAP environment is established. A change in the xAP address would be similar to a wholesale change in all the device codes in Homeseer thus resulting in breakage of all the events that use devices.

The mcs xAP environment has a convention of Location_Name_DC.SerialNumber.Type for identification of Homeseer devices. This simply means that the default information extracted from the address can be used to automatically determine the device type, name, etc. The SerialNumber segment of the address is used by the database application within the mcs environment to define a database table or field and start collecting data about the device.

Other individuals have different conventions as to how they address devices. Two options are provided by mcsXap to accomplish the address definition. The first one forces a three segment address and the second one is a variable segment one. The checkboxes are used to select which information is included in each layout.

Unless there is a reason to change the default the default should be used to minimize downstream headaches.

Speech and Voice Recognition does not have the utility it once had before HS2.x and the speaker client. For most users none of the checkboxes should be checked in this section.

The Event Triggers and Actions will likely be of interest eventually, but are not needed initially. An event trigger can be setup to look for a specific xAP message. Since the messages that most users will be using are xAPBSC and these messages are mapped into Homeseer devices, then a standard HS event trigger on a device will be the easiest way to recognize the event.

For the same reason the Action to send an xAP message will also not be used by most since the messages they will be sending will be xAPBSC and this is sent by simply changing the value/status/string of the Homeseer device.

What some may find useful is the Event Trigger based upon a lack of data being updated by an external xAP application. This would then be used to take an action to restart the application or generate message to provide awareness.

For the initial setup the Event support within mcsXap should not bog one down. It can always be added in the future when a need is identified.

There are several other setup options available and these should all be left in the disabled or default state. When specific need exists and greater comfort exists in the role that mcsXap provides then it makes sense to revisit the other options available.

5.4 mcsXap Initial Use

The first thing that one needs to do is get visibility into xAP traffic that exists. There are three elements to this. One is the xAP Hub. The second is the xAP Viewer. The third is the mcsXap viewing filters.

When a UDP datagram is sent it must be directed to a specific port or socket. xAP has reserved port 3639 for this purpose. It is much like HTTP defaults to 80 and SMTP defaults to 25 etc. Just as you cannot have two Web servers both responding to port 80, you cannot have more than one xAP application responding to port 3639. This is where the xAP hub comes into play. The xAP hub is given the responsibility to listen for all incoming xAP messages on port 3639. When it gets one it uses the localhost (127.0.0.0) interface to retransmit that message to every other xAP application running on the same computer. When something like xapmcs1Wire broadcasts a sensor value change that broadcasted message is actually received on the computer by the xAP hub and the xAP hub retransmits it locally to mcsXap and all others running on that same computer.

Since the xAP hub is the router for xAP traffic it is desirable that it is running before other xAP applications are started. This way startup information from the other applications will be routed to others that may want to listen.

The second item is the xAP Viewer. The xAP Viewer is a diagnostic tool. It is not required for normal xAP operations, but when some investigation is needed then it is an essential tool. It shows all the xAP traffic and presents it in an easily viewable manner.

The xAP viewer is a hybrid application. If it starts and it recognizes that there is nothing providing the role of xAP Hub then it will assume this role. If it does have this role (as is indicated in its title bar) and the viewer is closed then nothing will be serving the role of xAP hub and communications will effectively stop, but while it is open there will be communications.

The third item is within mcsXap with setup as shown in Figure 7. At the bottom of this figure are two lines that each show an xAP message which has the potential to be mapped into an Homeseer device. If the "A"ccept column checkbox is checked then all future receptions of a message from this address will be decoded and the values placed in the DeviceStatus, DeviceValue and DeviceString of the Homeseer Device.

At the time the "A" checkbox is checked a specific Device Code can be entered into text box on the same row. There is also a location and a name textbox on the same line that are not visible on the inserted figure. If all three of these are left blank then mcsXap will assign a Device Code and give it a name and location. These three parameters can be later changed without implications to the functioning of mcsXap. If changes are made it is best to do if from mcsXap rather than from Homeseer so mcsXap will know about the change.

There are five checkboxes at the top portion of Figure 7. These are used to identify the segment of the population of received xAP messages that should be viewed on the browser on the lines that were discussed in the prior paragraph.

The "Show Homeseer Received BSC" will normally be checked since xAPBSC schema is the one most used and visibility into what is received will be desired.

The "Show Other Received xAP Schema" covers all other messages. These messages are referred to as "Raw". For these messages mcsXap will map their contents into Homeseer devices as best it can. This usually means the message body will be place in the DeviceString. What this means is that events based upon new values from xAP messages will not be able to be setup using the event model of Homeseer which triggers only on DeviceStatus and DeviceValue. If changes in these devices need event notification then the mcsXap Event Trigger can be used. This, however, is typically not needed since the message content is generally informative rather than actionable.

Every device in Homeseer can have its values sent as xAP messages using the xAPBSC schema. It is likely that a limited number of the devices will need to be transmitted for

other xAP applications to use. This of course depends upon the extent of the xAP network. The “Show Homeseer Sourced BSC” checkbox will make the set of device available on this page and then “A”ccept checkbox used to select the device as being active as an xAP message. The message will be sent every time Homeseer changes the DeviceStatus, DeviceValue or DeviceString of the device. It will also be sent when queries using the xAP query schema.

The query schema is xAPBSC.Query. xAP allows wildcarding with xAPBSC as well as all other xAP schema. That means that a query can be made for the status of a single device or of a class of multiple devices.

mcsXap also provides a button to simulate a fully wildcarded xAPBSC.Query message. This is “Send BSC” button. When this button is clicked then the values for all Homeseer devices that have been “A”ccepted for xAP message transmission will be sent using the xAPBSC.Info schema.

There is also a “Query BSC” button on the line labeled “BSC In”. When this button is clicked a fully wildcarded xAPBSC.Query message will be sent to all other xAP applications on the LAN and each application will respond with the current values of all their devices. When Homeseer starts mcsXap will automatically send this same request so it becomes synchronized with the entire xAP network.

The “BSC In” line also has input provisions for “BSC Target Address Mask”. When this is filled in then the scope will be reduce of the xAP endpoints that will respond. The mask can be a specific end point or partially wildcarded. Initially a fully wildcarded query will suffice. When there is more familiarity with the xAP addressesing structure then more restrictive queries can be used.

If new devices are added to Homeseer outside of mcsXap and this device is to be transmitted via xAP then it will be necessary to use the “Repopulate” button on the “BSC Out” line. mcsXap populates its image of Homeseer devices at startup so a restart will also have the same effect as using the “Repopulate” button.

The “Display” line checkboxes are used as filters based upon the “R”ejected and “A”ccepted checkboxes associated with each device in the body of the table. Normally the “Show Only Accepted” is the only one that is checked for normal viewing. When new devices are to be added it is unchecked so they become visible for selection.

The “R”ejected checkbox is used a display filtering mechanism to hide devices that are not of immediate interest. There is also a “D”elete checkbox. When it is used then all record of receiving the message will be removed. It is appropriate for invalid messages of for messages that were received from hardware that no longer exists.

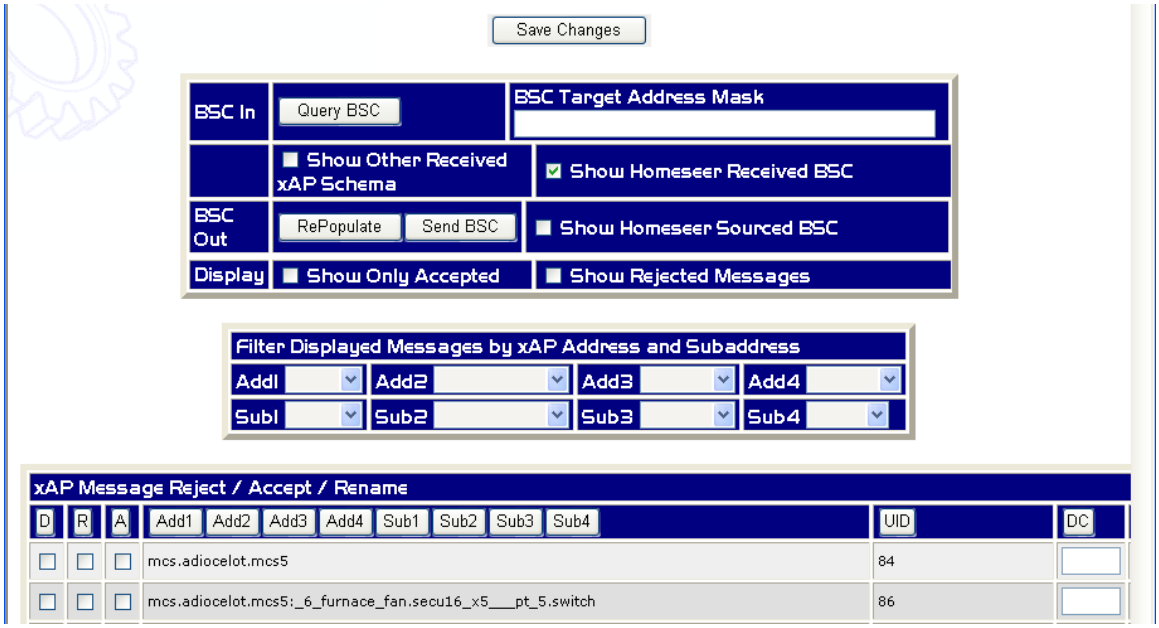


Figure 7 Message Selection Filters

5.5 xapmcsHub Setup

xapmcsHub is available from the xAP Library section of the Homeseer Message Board. It is a VB6 .exe application that provides a user interface access via a tray icon. xapmcsHub should be installed in its own folder. A structure such as C:\Program Files\xAP\xapmcsHub\xapmcsHub.exe is appropriate. Other xAP applications would also be installed in a similar manner in independent folders.

The setup form is viewed from the tray icon. Three checkboxes are provided for setup of xapmcsHub. The typical configuration will be as shown in Figure 8 where only xAP will be processed, an icon will appear in the tray and no backup hub function will be spawned. Once stable operations are achieved the tray icon may be removed if one desires a less-cluttered tray.

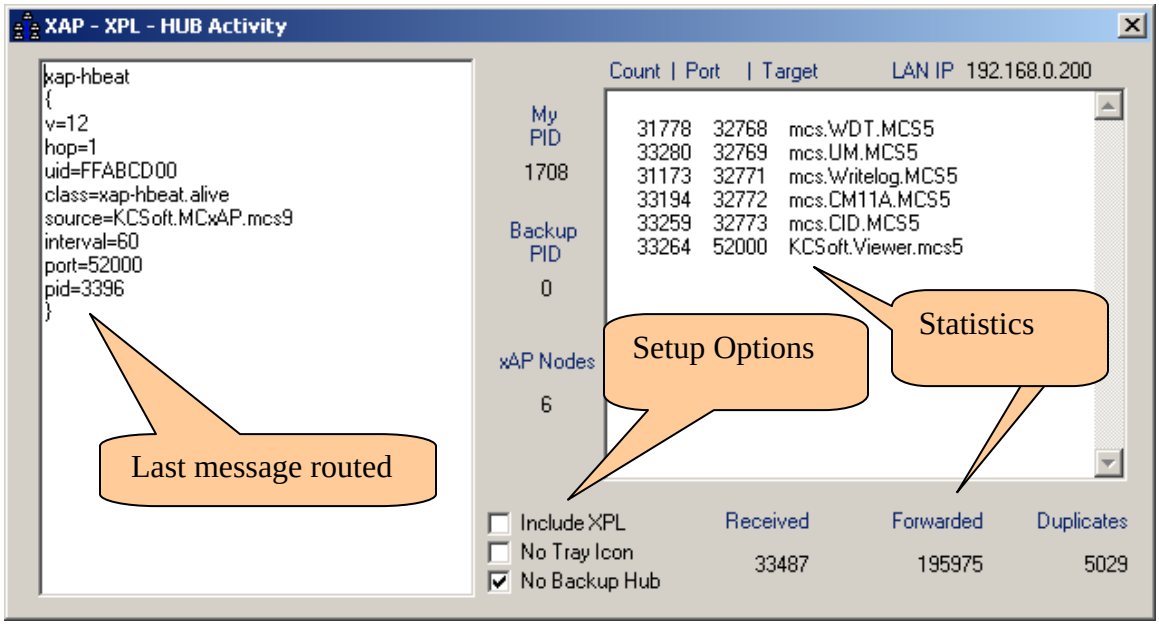


Figure 8 xAP Hub Activity Form

There are a variety of xAP hubs available and any should be able to be used. There xFx (via xapautomation.org) makes available a .NET hub that can be installed as a service. This is an attractive option as it provides the benefit of running before other desktop xAP applications.

The only issue I have with it is its lack or removal of duplicate messages. In many case duplicates do not matter, but in cases where the message triggers events then having multiple events triggers is problematic. Note in Figure 8 that 5029 duplicate messages were detected and filtered-out by xapmcsHub.

5.6 xAP Message Viewer

The xAP Viewer was originally developed by xapFramework.net and is now maintained by xFx. It is a useful tool to observe xAP message traffic and it can also serve the dual role as a xAP hub.

Figure 9 and Figure 10 show a typical display for messages summaries and message detail that is available from a Window's GUI. When using the xAP viewer and a dedicated hub one needs to be careful to start the dedicated hub before the viewer otherwise the hub will not be able to gain access to the xAP port on the primary interface.

The left panel of the viewer provides a structured list of xAP endpoints. The + and – symbols can be used the compress and expand the structure. Clicking on any level will limit the contents of the right panel to only those endpoints. The right panel provides a chronological list of xAP messages. Clicking on a message from this panel will bring up

a popup form that contains the detail of the message. An example is shown in Figure 10. The content of this second panel can be edited and it can be retransmitted. These features make it useful for diagnostic and testing activities.

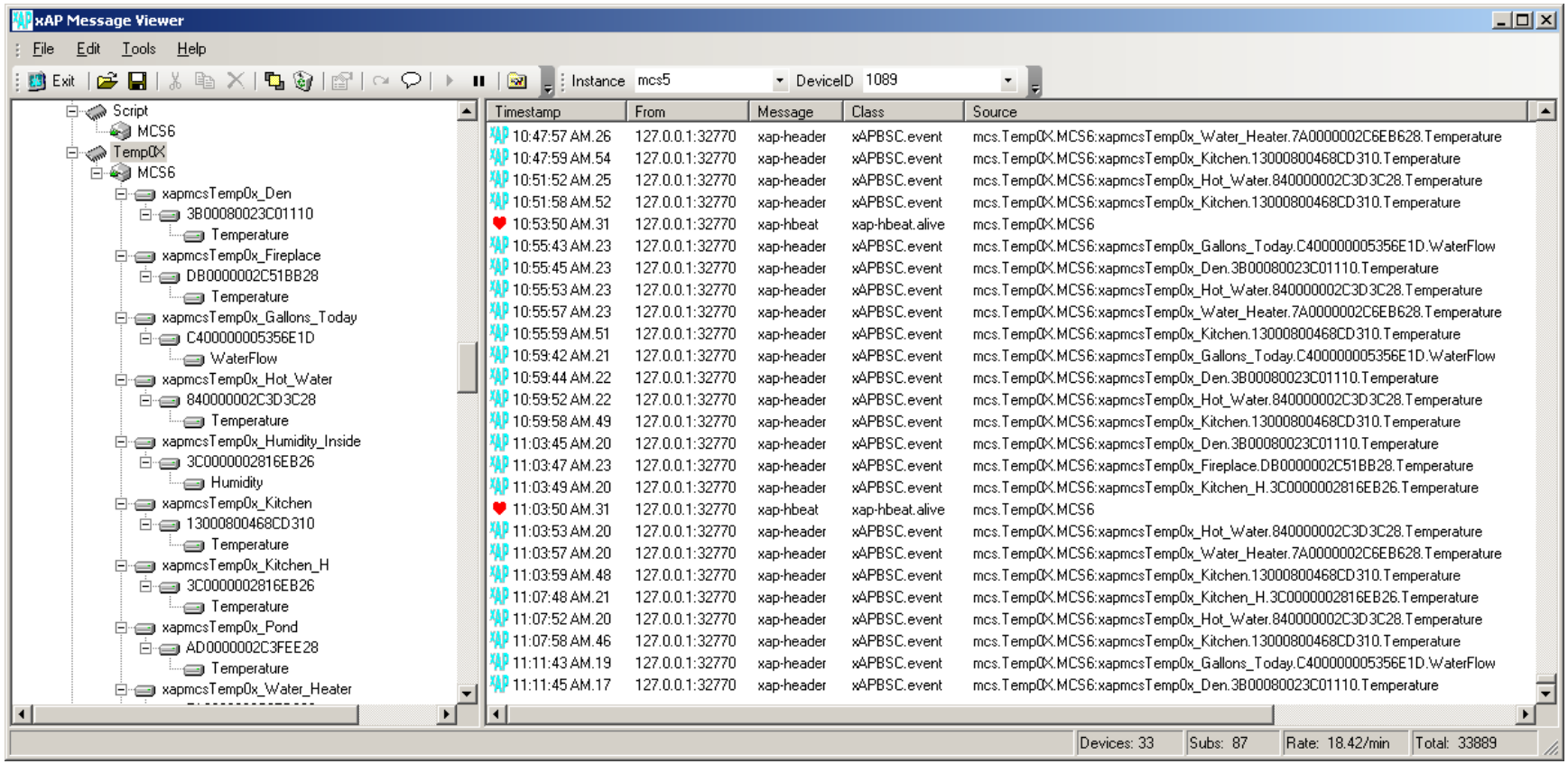


Figure 9 xAP Viewer

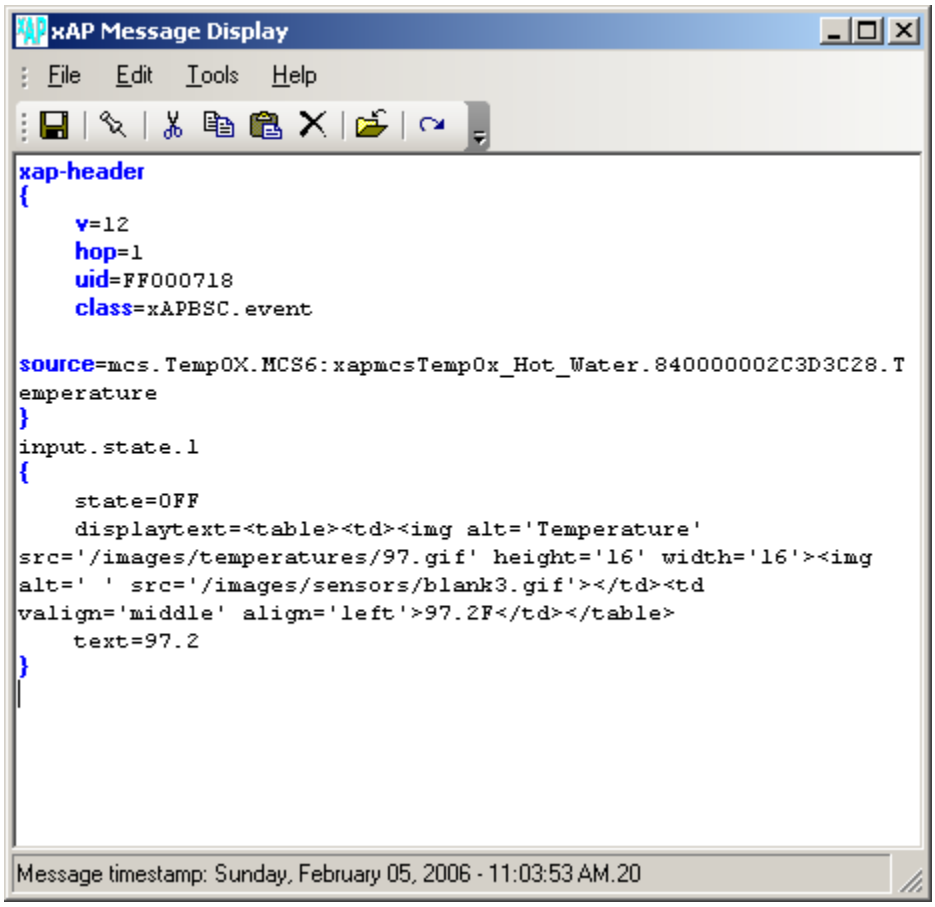


Figure 10 xAP Viewer Message Detail

6 xAP Schema

The xAP schema described in this section is used by xapmcsMultimeter.

6.1 xapmcsMultimeter

6.1.1 Receive xAP Schema

xAPBSC
    xapbsc.cmd – used to reset the min/max reporting vaue

6.1.2 Transmit xAP Schema

xAPBSC
    xapbsc.event – used to report changes in multimeter readings
    xapbsc.info – used to report last received multimeter readings

WDT.Echo
    Echo.Response
        Response=<value received in wdt.echo/echo.query>

Homeseer.Role
    Role.Notification
        Role=Computer Management
        Path=<folder path to xapmcsWebControl.exe>

Homeseer.Event
    Event.Log
        Time=<time>
        Type=<text>
        Data=<text>