ANT Auto Shared Channel

ABSTRACT

The ANT auto shared channel is an optional extension to an ANT shared channel. It offers a simple method to dynamically and automatically add/remove slave devices to an ANT shared channel. The ANT auto shared channel can be implemented at the application level, or by using a SensRcore™ device. SensRcore ™ devices have auto shared channel slave functionality built in. An example application level implementation of an auto shared channel master is provided with this application note.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 Introduction

ANT shared channels are used to establish reliable bi-directional communication between a master device and multiple slave devices using a single ANT channel. This configuration is especially suited for applications in which a central node collects measurement data from a large number of sensors. ANT shared channels avoid the problem of contention among slave devices that can occur in standard ANT channels by using slave addressing. The master may give each slave device a unique address. This enables the polling of individual slaves to reliably retrieve measurement data.

ANT shared channels require the master to have knowledge of each slave device's address. In a system that has a fixed number of known sensors, this is relatively simple to implement. If the system involved an arbitrary number of sensors with unknown addresses, or required that slaves be added or removed dynamically from the network, the ANT shared channel will not be sufficient on its own. The ANT auto shared channel extension was created to control and manage this additional functionality. This application note provides implementation details of an auto shared channel master device using the ANT Development Kit.

2 Relevant Documents

It is strongly recommended that the following documents be reviewed prior to using this application note, and viewing the source code:

- ANT Message Protocol and Usage
- Interfacing with ANT General Purpose Chipsets and Modules
- ANT AT3 RF Transceiver Module Datasheet
- AN06 ANT SensRcore™ Mode
- ANT SensRcore Messaging and Usage

3 Bill of Materials

- 1 x ANTDKT3 or ANTAP2DK1 Development Kit

4 ANT Auto Shared Channel

ANT shared channels are relatively simple to implement in systems that have a set number of slaves that are assigned unique addresses. However, many use cases exist where:

- The master device has no knowledge of the slaves' shared addresses
- The slave devices cannot be provided with unique shared addresses.

Configuring and maintaining a shared channel in these situations are not straight forward. The ANT auto shared channel functionality has been created to allow a master device to dynamically register a slave device. The master device can also assign the slave a unique address that can be used for participation in the shared channel.

The ANT auto shared channel functionality has been encapsulated into simple application level state machines in both master and slave devices. The goal was to have the least amount of impact on higher level application code, and make integration of the new functionality as easy as possible.

5 Handshaking

"Handshaking" allows a slave device to register with a shared channel master device and obtain an address for participation in the shared channel. The shared address field may be 1 or 2 bytes, depending on the capabilities of the ANT parts used (Refer to datasheets for more information). The example presented in this application note (and accompanying source code) is based on a 2 byte shared address.

The handshaking process consists of a series of command and response messages passed between the master and slave devices. The message sequence used during the registration process is described in Section 5.2.

5.1 Detailed Message Description

The auto shared channel handshaking message details are provided in the following sections.

5.1.1 Address Available (0xFF)

The Address Available message is sent by the master to announce that a shared address is available on the shared channel. The shared address of this message is set to 0xFFFF. Shared address 0xFFFF is reserved for registration messages, and is not assigned to any slave device. The remaining fields are set according to Table 1.

Table 1. Address Available Message Content

Byte	Description	Length	Value	Units	Max Value
0	Shared Address LSB	2 bytes	Holds the value of the shared address. For the Address Available message, the shared address is 0xFFFF. 0xFFFF shall not be assigned permanently to any slave device.	N/A	65533
1	Shared Address MSB				
2	Message ID	1 byte	Indicates the ID of the message. For Address Available, the message ID is 0xFF.	N/A	N/A
3	Channel Period Decimation LSB	2 bytes	The master may specify the recommended channel period decimation value if desired.	s*32768	65535
4	Channel Period Decimation MSB				
5	Data Timeout	1 byte	The master may specify the recommended data timeout value if desired.	2 s	255 max (infinite)
6	Next Address LSB	2 bytes	The next available address on the auto shared channel.	N/A	65535
7	Next Address MSB				

5.1.1.1 Channel Period Decimation

This is a two byte value that specifies the recommended channel period decimation value, in seconds * 32768. This value indicates how often a particular slave can expect to be polled by the master during the normal operation of the shared channel, and is usually related to the data timeout. The master may suggest this value to the slave so that the device may decimate its channel period to an acceptable level that conserves power, while catching all requests for data from the master. It is not recommended that the slave decimate the channel period, because it is difficult to decide which message the slave should synchronize to, unless the application has been designed specifically to poll the slave with that in mind.

5.1.1.2 Data Timeout

The data timeout field allows the master to specify the data timeout value that a slave should wait before giving up its shared address. The data timeout value is presented in 2 second increments. Different timeouts may be chosen for different slaves.

5.1.1.3 Next Address

This field specifies the next shared address that is available on the shared channel.

5.1.2 Request Address (0xFD)

The Request Address message is sent by the slave in response to an Address Available (0xFF) message. This message informs the master that the slave wants to initiate the registration process. The shared address of this message is set to 0xFFFF. The remaining fields are set as shown in Table 2.

The Request Address message is sent as an acknowledged message. A successful acknowledgement allows the slave to determine that this message was received by the master.

If the slave does not receive a successful acknowledgement, it will wait for a variable timeout period and then retry the request. If the slave continues to receive no acknowledgement, it will return to the beginning of the handshake sequence, and wait for the Address Available message.

Table 2. Request Address Message Content

Byte	Description	Length	Value	Units	Max Value
0	Shared Address LSB	2 bytes	Holds the value of the shared address. For the Request Address message, the shared address is 0xFFFF. 0xFFFF shall not be assigned permanently to any slave device.	N/A	65533
1	Shared Address MSB				
2	Message ID	1 byte	Indicates the ID of the message. For Request Address, the message ID is 0xFD.	N/A	N/A
3	Reserved	1 byte	Value = 0x00	N/A	N/A
4	Unique ID 0	4 bytes	The Unique ID of the slave device requesting an address on the auto shared channel.	N/A	4294967295
5	Unique ID 1				
6	Unique ID 2				
7	Unique ID 3				

5.1.2.1 Unique ID

This four byte field allows the slave to uniquely identify itself during the registration process. This value can be the unique serial number assigned to the slave by the manufacturer, or it can be a random number generated by the slave at the time of registration.

5.1.3 Busy Acquiring (0xFE)

The Busy Acquiring message is sent by the master once it has received a Request Address (0xFD) message from a slave. This message is used to indicate that the master is now actively engaged in the registration process with a slave device. The shared address of this message is set to 0xFFFE. All fields shall beset as described in Table 3.

Table 3. Busy Acquiring Message Content

Byte	Description	Length	Value	Units	Max Value
0	Shared Address LSB	2 bytes	Holds the value of the shared address. For the Busy Acquiring message, the shared address is 0xFFFE. 0xFFFE shall not be assigned permanently to any slave device.	N/A	65533
1	Shared Address MSB				
2	Message ID	1 byte	Indicates the ID of the message. For Busy Acquiring, the message ID is 0xFE.	N/A	N/A
3	Reserved	1 byte	Value = 0x00	N/A	N/A
4	Unique ID 0	4 bytes	The Unique ID of the slave device requesting an address on the auto shared channel. Same ID as found in the Request Address message.	N/A	4294967295
5	Unique ID 1				
6	Unique ID 2				
7	Unique ID 3				

5.1.3.1 Unique ID

This field shall contain the unique identifier of the slave device to which the master is assigning the shared address. This shall contain the same four byte value that was received by the master in the Request Address (0xFD) message.

5.1.4 Confirm Acquire (0xFC)

This message is sent by the slave to confirm with the master that it has received the Busy Acquiring (0xFE) message. The shared address on this message is set to 0xFFFE. All fields shall be set according to Table 4.

Confirm Acquire is sent as an acknowledge message. If successfully acknowledged, the registration process is complete. The slave may begin normal operation of the shared channel with the address assigned to it by the master. The master device shall include the newly assigned shared address during typical shared channel operation.

If the slave does not receive a successful acknowledgement it will return to the beginning of the handshake sequence, and try to acquire a new address. The master will attempt to poll the slave device. After a certain time period of inactivity from that device, the master will reclaim the address.

Table 4. Confirm Acquire Message Content

Byte	Description	Length	Value	Units	Max Value
0	Shared Address LSB	2 bytes	Holds the value of the shared address. For the Confirm Acquire message, the shared address is 0xFFFE. 0xFFFE shall not be assigned permanently to any slave device.	N/A	65533
1	Shared Address MSB				
2	Message ID	1 byte	Indicates the ID of the message. For Confirm Acquire, the message ID is 0xFC.	N/A	N/A
3	Reserved	1 byte	Value = 0x00	N/A	N/A
4	Unique ID 0	4 bytes	The Unique ID of the slave device requesting an address on the auto shared channel. Same ID as found in the Request Address Message.	N/A	4294967295
5	Unique ID 1				
6	Unique ID 2				
7	Unique ID 3				

5.1.4.1 Unique ID

This field shall contain the unique identifier of the slave device to which the master is assigning the shared address. This shall contain the same four byte value that was received by the master in the Request Address (0xFD) message.

5.1.5 Shared Channel Information (0xFA)

The master sends the Shared Channel Information message to provide a slave device information it may need to participate more effectively in the shared channel. This message contains information that may be useful to a slave device, but it is not necessary for the operation of the shared channel. Therefore, the Shared Channel Information message is not included in the registration process.

This message may be sent by the master whenever application data for a specific slave is unavailable or not required. That is, if there is no application data for "Slave A" in its designated timeslot, then the shared channel information message may be sent instead. It is possible for a slave to request this information using the Request (0xF0) message.

This message is addressed to a specific slave and shall contain the shared address corresponding to that slave device. All fields in this message shall be set as described in Table 5.

Table 5. Shared Channel Information Message Content

Byte	Description	Length	Value	Units	Max Value
0	Shared Address LSB	2 bytes	Holds the value of the shared address. For the Channel Information message, this will be the shared address assigned to the slave after handshaking has completed. This value shall not be 0xFFFF or 0xFFFE	N/A	65533
1	Shared Address MSB				
2	Message ID	1 byte	Indicates the ID of the message. For Channel Information, the message ID is 0xFA.	N/A	N/A
3	Reserved	2 bytes	Value = 0x0000	N/A	N/A
4	Reserved				
5	Data Timeout	1 byte	The master may specify the recommended data timeout value.	2 s	255 max (infinite)
6	Channel Period Decimation LSB	2 bytes	The master may specify the recommended channel period decimation value.	seconds * 32768	65535
7	Channel Period Decimation MSB				

5.1.5.1 Data Timeout

This is a single byte value used by the master to specify the recommended data timeout value, in 2 second increments, that a slave device should use before relinquishing its shared address. The master can choose to recommend different values for different slave devices

5.1.5.2 Channel Period Decimation Value

This is a two byte value that specifies the recommended channel period decimation value, in seconds * 32768. This value indicates how often a particular slave can expect to be polled by the master during the normal operation of the shared channel, and is usually related to the data timeout. The master may suggest this value to the slave so that the device may decimate its channel period to an acceptable level that conserves power, while catching all requests for data from the master. It is not recommended that the slave decimate the channel period, because it is difficult to decide which message the slave should synchronize to, unless the application has been designed specifically to poll the slave with that in mind.

5.1.6 Request (0xF0)

The Request message is used by the slave to request channel information from the master. This message shall contain the shared address that was assigned to the slave. All fields in this message shall be set as described in Table 6.

Table 6. Request Message Content

Byte	Description	Length	Value	Units	Max Value
0	Shared Address LSB	2 bytes	Holds the value of the shared address. For the Request message, this will be the shared address assigned to the slave after handshaking has completed. This value shall not be 0xFFFF or 0xFFFE	N/A	65533
1	Shared Address MSB				
2	Message ID	1 byte	Indicates the ID of the message. For Request, the message ID is 0xF0.	N/A	255
3	Reserved	4 bytes	Value = 0x00000000	N/A	N/A
4	Reserved				
5	Reserved				
6	Reserved				
7	Data/Command Wanted	1 byte	Indicates the Data or Command that the slave device is requesting. For shared channel information, this field would be 0xFA.	N/A	N/A

5.1.6.1 Data/Command Wanted

This is a single byte value used to specify the specific data type or command message being requested.

5.1.7 No Address Available (0xFB)

This message is sent by the master in place of the Address Available (0xFF) message when further slave connections are unavailable or not desired. The shared address on this message is set to 0xFFFF. All fields in this message shall be set as described in Table 7.

Table 7. No Address Available Message Content

Byte	Description	Length	Value	Units	Max Value
0	Shared Address LSB	2 bytes	Holds the value of the shared address. For the No Address Available message, the shared address is 0xFFFF. 0xFFFF shall not be assigned permanently to any slave device	N/A	65533
1	Shared Address MSB				
2	Message ID	1 byte	Indicates the ID of the message. For No Address Available, the message ID is 0xFB.	N/A	N/A
3	Reserved	2 bytes	Value = 0x00	N/A	N/A
4	Reserved				
5	Data Timeout	1 byte	The master may specify the recommended data timeout value.		
6	No Free Address LSB	2 bytes	This is used to indicate that there is no free address on the auto shared channel. Value = 0x0000	N/A	N/A
7	No Free Address MSB				

5.2 Message Sequence

The master device begins by sending the Address Available (0xFF) message to inform any searching slave that there is availability on the shared channel.

The master shall send the Address Available (0xFF) message on any free channel period. This occurs whenever the master is not directly communicating with a specific slave. The master shall also only send the Address Available (0xFF) message if it has not reached its maximum capability of slave connections. If it has reached the maximum capacity, it shall send the No Address Available (0xFB) message instead.

Any slave that is listening for the Address Available message shall then respond with the Request Address (0xFD) message. The slave device shall include a unique ID with the request. This may be the slave device's serial number or a random number generated for the purposes of registration

On receiving the Request Address (0xFD) message, the master shall transmit the Busy Acquiring (0xFE) message on any free channel periods. The unique ID received from the Request Address message is used by the master to identify the particular slave to which it is assigning the shared address. The use of the slave's unique ID mitigates the possibility of contention between multiple slave devices when requesting an address.

A slave device shall only continue with the registration process if its unique ID matches the one contained in the Busy Acquiring (0xFE) message. The slave shall then send the Confirm Acquire (0xFC) message, also containing the unique ID of the slave. A successful acknowledgement of this message indicates the end of the handshaking process. Section 5.1 addresses the possibility of an unacknowledged message during the registration process.

During typical operation, the master shall send the Shared Channel Information (0xFA) message when no application data is available for a specific slave in its timeslot. Refer to Section 5.1 for more details.

Alternatively, if the master application does not allow for providing the Shared Channel Information (0xFA) message by default, the slave can request the information using the Request (0xF0) message.

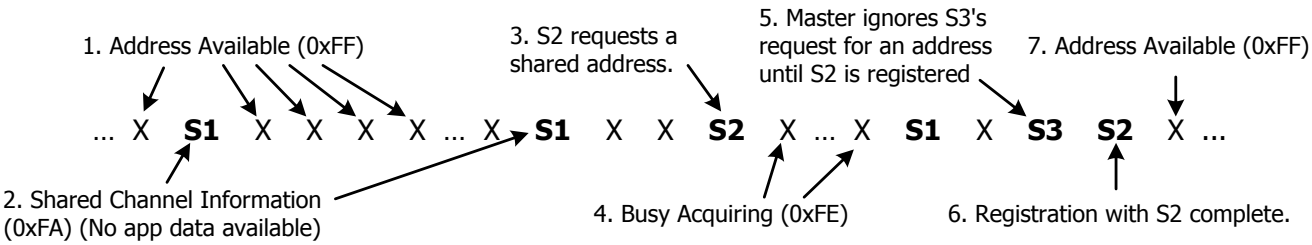

Figure 1. Message Rotation During Regulation Auto Shared Channel Communication

5.3 Example Message Sequence

A detailed example of an initial "Handshake" is shown in Figure 2.

In the example in Figure 2, the unique ID is 0x12345678.

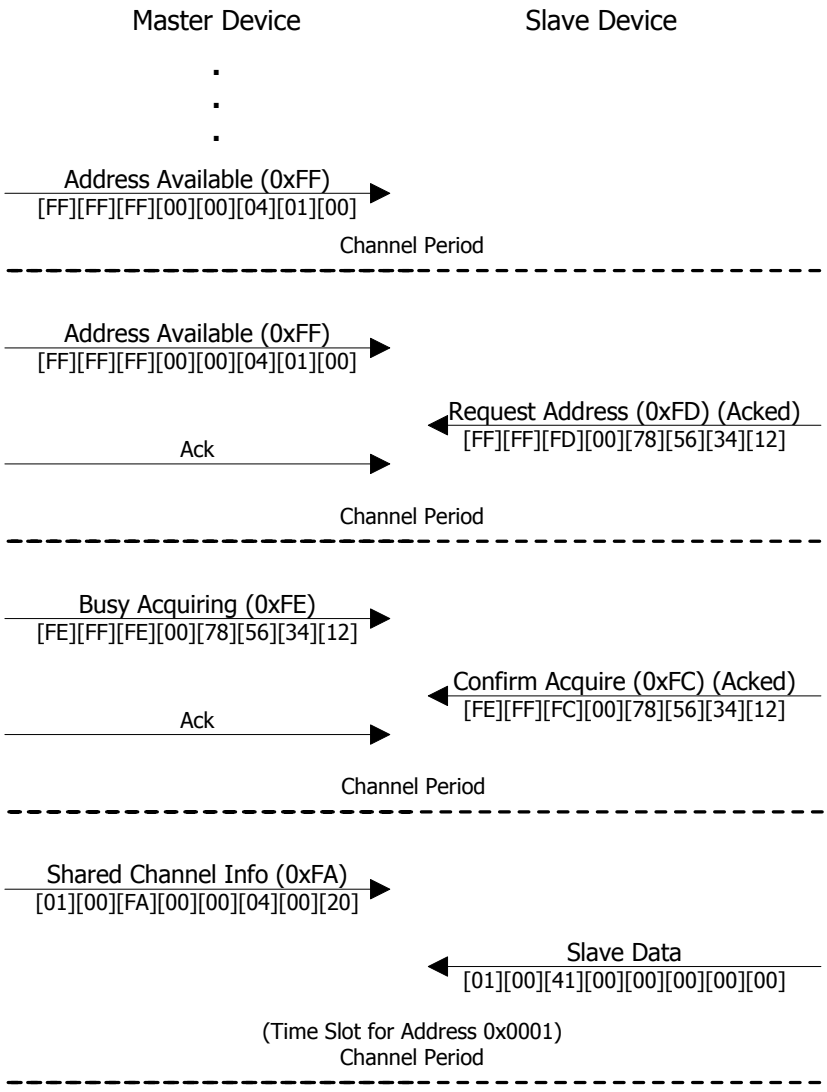

Figure 2. Message Sequence for Device Registration

Assigned shared addresses can be reclaimed by the master once it has determined that a slave device is no longer present or required. The master can reclaim the address by either using a timeout mechanism or an application level command message. In order to un-assign a shared address the master may simply stop using it and allow the slave device to timeout (i.e. exceed the suggested Data Timeout); however this does have to be implemented at the application level.

5.4 Master State Machine

Figure 3 shows the auto shared channel registration state diagram of the master device.

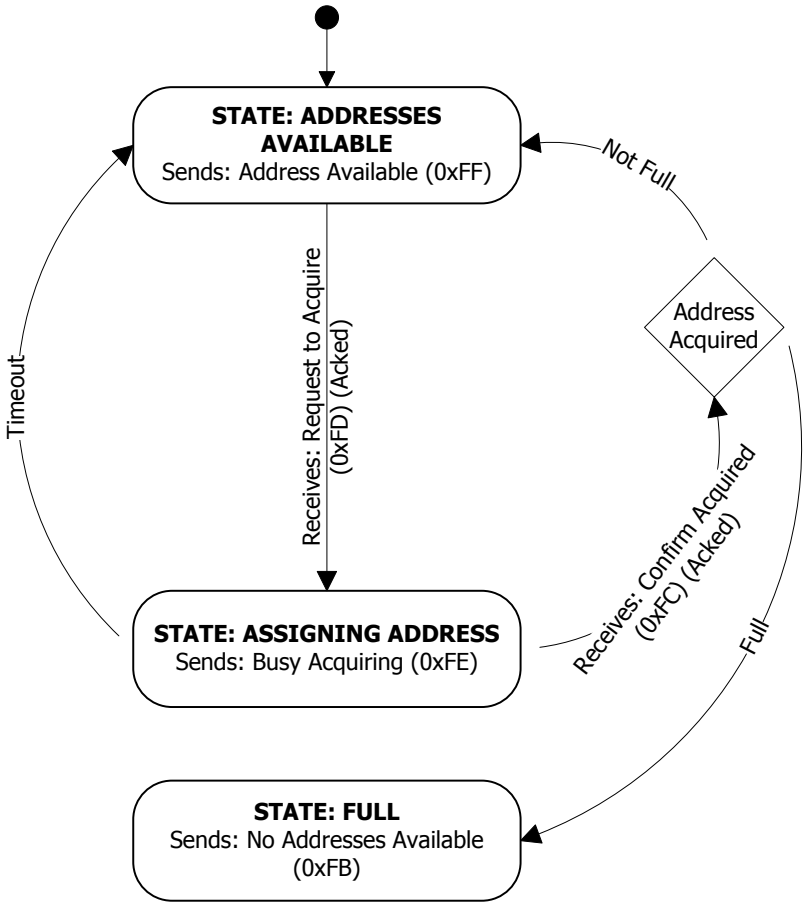

Figure 3. Auto Shared Master State Machine

5.5 Slave State Machine

Figure 4encapsulates the different stages the slave device will go through during the shared channel registration process.

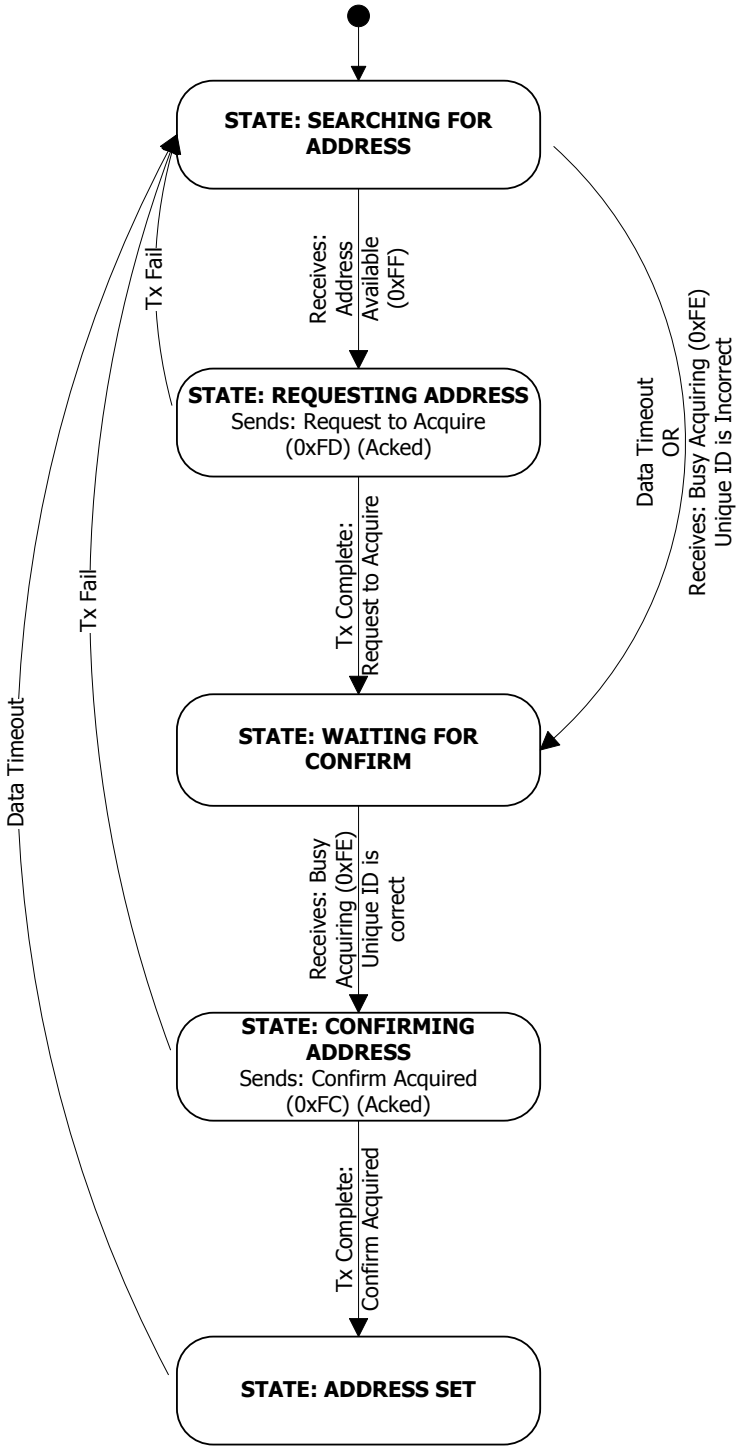

Figure 4. Auto Shared Slave State Machine

6 Implementation Example

The attached source files provide a PC based data collection application configured to be an auto shared channel master. Note that the example application and source code provided does not implement 'reclamation' of shared addresses on the master side once a slave disappears from the message rotation. It simply provides a starting point for exploring the auto shared channel functionality.

6.1 SensRcore Device

The SensRcore module has preloaded scripts that will allow it to run in an auto shared channel slave configuration. The setup of the module is detailed in the following sections.

6.1.1 Hardware Setup

The hardware required for the auto shared channel example is provided in the ANT Development Kit. The battery boards are used to provide power and configure the ANT11TS33M5IB modules to run in the correct mode. To configure the modules for this example set switches 1 through 4 to GND position and switch 5 to the VCC position. The configuration dip switches are located on the battery board next to the battery holder.

Table 8. Auto Shared Slave temperature Sensor Pin Configuration

	Vcc	Gnd
1 (IOSEL)		X
2 (MEMSEL)		X
3 (DEVSEL)		X
4 (DEVSEL)		X
5 (DEVSEL)	X	

6.2 PC Node

The Auto Shared Channel Demo application uses an ANT USB board with an ANT module to form the master side of the Auto Shared Channel example. The application's default parameters have been selected to match the channel parameters set on the ANT SensRcore™ modules.

6.2.1 Software Requirements

In order to use to Auto Shared Channel Demo, it is necessary to download and install Microsoft .NET Framework 4.0. In addition to the Auto Shared Channel Demo application, the source code for Visual C# 2010 is available with the application note package.

6.3 Test Procedure

Follow the procedure listed below to run and test the ANT auto shared channel example:

1. Connect the ANT USB board containing the Master module to the PC.
2. Run the Auto Shared Channel Demo application.
3. Select the correct USB Device Number (0 if only one ANT USB board is connected).
4. Click the "Connect Device" button.
5. Select the desired Channel Settings. For this example, leave all settings in their default configuration. Click the "Open Channel" button.
6. Configure the dip switches on the battery board that will hold the Slave Module
7. Connect the ANT11TS33M5IB as a slave module (up to two slave modules may be configured using the development kit).

8. Power cycle or reset the Slave module using the reset button on the battery board.

The demo application will show the progress of the slave device as it registers with the master. After registration is complete, temperature data from the slave device will be displayed in the demo application window. If an ANT IO Board is connected to the slave module, the LED checkbox on the master application can be used to control the state of LED A.

If the channel connection between the master and slave device is lost for a prolonged period of time, or if the slave is reset, the slave device will re-register itself as soon as it is able to find the master channel again. Likewise, if the master application was reset, the slave device will re-register itself after it has determined that it has lost its original connection to the master.

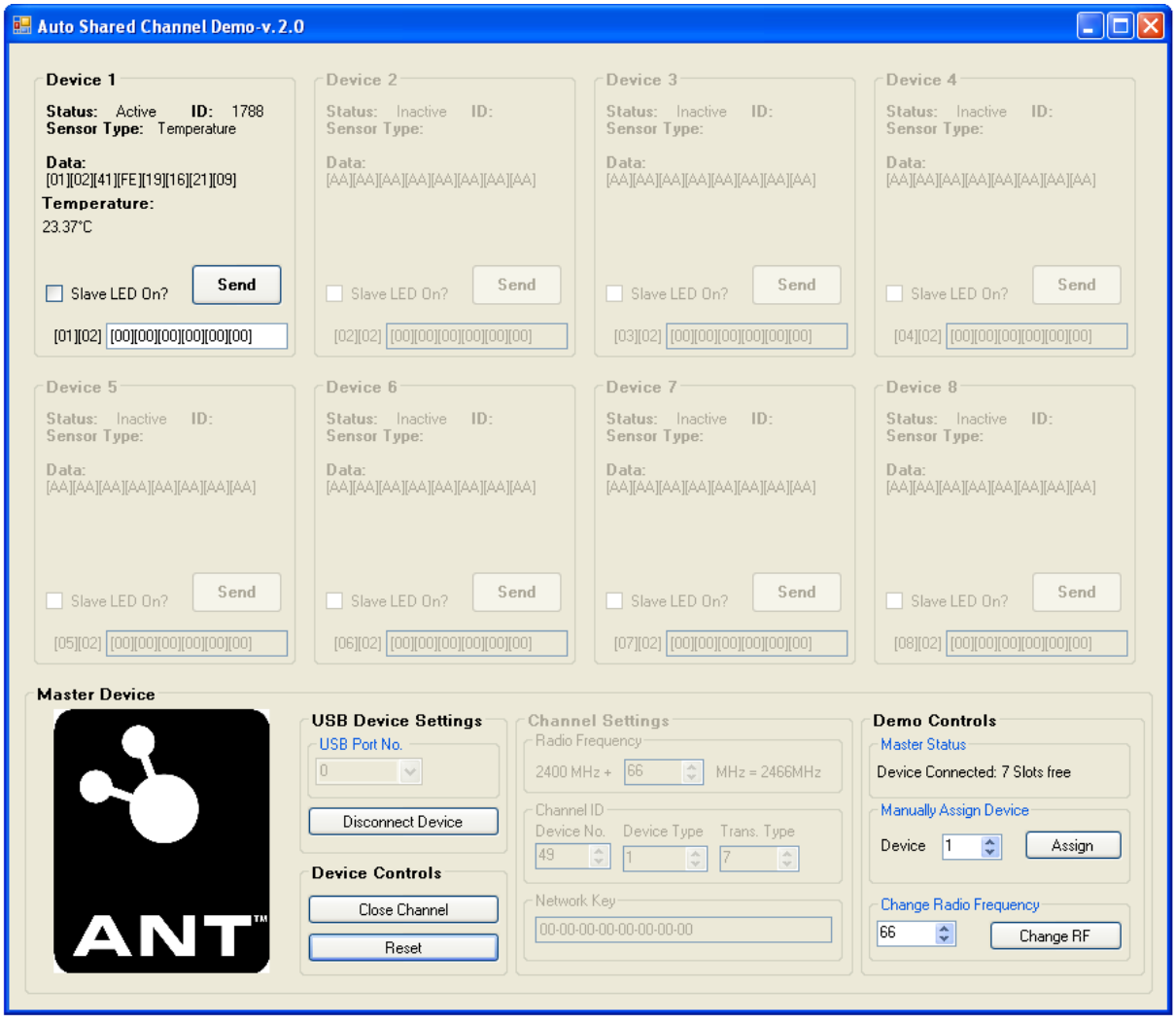

Figure 5. Screen Capture of Auto Shared Channel Demo Application with Slave Connected

If it appears that the slave device is not displayed on the master application, it is likely that the slave device has not been configured correctly. Check the configuration switches on the battery board attached to the slave module and reset both the master application and slave module.

7 Closing Remarks

This application note is aimed at providing a detailed description of the ANT auto shared channel application level extension, along with its features, advantages and proper usage. The example source files show how to implement an ANT auto shared channel PC based master device.

Please direct any technical inquires and/or general comments regarding this document or associated files to thisisant.com.