

## Integrating an External Device with the GO Device

### Introduction

External devices can communicate with the Geotab GO device through the revised Third Party Data protocol. An initial Handshake will need to be successfully negotiated in order for the GO device to accept third party data. Data can only be sent in a predefined format and will be saved and sent to Checkmate as Status Data.

### Integration Process

The following process should be followed when attempting to integrate a third-party device with the GO unit using our Third Party Data Protocol:

#### 1. Request External Device ID

Contact Geotab support and obtain an External Device ID for your third-party device. This will allow us to uniquely identify the device when it connects to a GO unit. The external device detection will be saved as Status Data in Checkmate.

#### 2. Request Status Data IDs

Request Status Data ID's for every separate piece of information you wish to send to the GO device. Each piece of information must be sent separately and have its own Status Data ID assigned. For each piece of information the conversion method is also required. Conversions must be in the form as per the examples below:

Status Data ID	Description	Multiplier	Offset	Symbol
5 ( <i>assigned by Geotab</i> )	Odometer	0.1	0	Km
53 ( <i>assigned by Geotab</i> )	Outside Temperature	1	-40	Degrees C

#### 3. Implement the Third Party Protocol

Implement the Third Party Protocol in the external device as detailed below.

## Handshake

The GO device will listen for the Handshake Sync from the external device at the time it starts to detect whether a device is present and to auto-detect the external device BAUD rate. This sync character needs to be sent once per second by the external device. The GO device will then reply with a Handshake Request message to which the external device must reply with a Handshake Confirmation message. From this time the external device no longer needs to send the Handshake Initialisation message and can start sending third party data when required. The external device must still respond to the Handshake Request message with a Handshake Confirmation message at any time. If the external device does not receive an acknowledge message from the GO device when sending third party data it must start streaming the Handshake Sync again.

1. The External device sends through the Handshake Sync once per second until the Handshake Request is received from the GO Device.
2. The External Device responds to the Handshake Request with the Handshake Confirmation containing the Geotab assigned External Device ID.
3. The External Device sends a Third Party Data message whenever it wants the GO Device to save data. It will receive an Acknowledge to confirm the message was received.
4. The GO Device can send through a Handshake Request message at any time and the External Device must respond with the Handshake Confirmation.
5. If the External Device receives no response to the Third Party Data message it must restart the handshake process – return to point 1 above.

*See Appendix A for example raw message data, see Appendix B for a graphical representation of the message flow.*

## Data

All values must be sent with the least significant byte first.

## Checksum

Each message will contain a 2 byte Fletchers Checksum calculated across all the bytes of the message up to, but not including, the checksum. The checksum values are bytes, so they overflow from 255 (0xFF) to 0 (0x00).

Checksum calculation pseudo code:

```
byte ChkA = 0;
byte ChkB = 0;

// n is the number of bytes in the message
// up to, but not including, the checksum
for (i = 0; i < n; i++)
{
    ChkA = ChkA + MsgBuffer[i];
    ChkB = ChkB + ChkA;
}
// ChkA precedes ChkB in the message.
```

## Messages from GO Device

### Msg Type 1: Handshake Request

Issued by GO Device on receipt of the Handshake Sync, and periodically resent to confirm the External Device is still attached.

	Bytes	Position
STX (0x02)	1	0
Message Type = 1	1	1
Message Body Length = 0	1	2
Checksum	2	3
ETX (0x03)	1	5
<i>Reply: Handshake Confirmation</i>		

### Msg Type 2: Third Party Data Acknowledge

Issued by GO Device on receipt of Third Party Data from the External Device.

	Bytes	Position
STX (0x02)	1	0
Message Type = 2	1	1
Message Body Length = 0	1	2
Checksum	2	3
ETX (0x03)	1	5

## Messages from External Device

### Handshake Sync (Auto-BAUD detect)

Issued by External Device every second until the Handshake Request is received.

	Bytes	Position
Sync Char (0x55)	1	0
<i>Reply: Handshake Request</i>		

### Msg Type 0x81: Handshake Confirmation

Issued by the External Device when it receives the Handshake Request.

	Bytes	Position
STX (0x02)	1	0
Message Type = 0x81	1	1
Message Body Length = 4	1	2
External Device ID (assigned by Geotab)	4	3
Checksum	2	7
ETX (0x03)	1	9

### Msg Type 0x80: Third Party Data as Status Data

Issued by the External Device whenever it wants Third Party Data saved in the GO Device as if it was Status Data.

	Bytes	Position
STX (0x02)	1	0
Message Type = 0x80	1	1
Message Body Length = 6	1	2
Data ID (assigned by Geotab)	2	3
Data	4	5
Checksum	2	9
ETX (0x03)	1	11
<i>Reply: Third Party Data Ack</i>		

### Msg Type 0x82: Free Format Third Party Data

Issued by the External Device whenever it wants Third Party Data saved in the GO Device in a free format (1 to 27 bytes) that will be saved into Checkmate as Custom Data.

	Bytes	Position
STX (0x02)	1	0
Message Type = 0x82	1	1
Message Body Length = x (1 to 27)	1	2
Data	x	3
Checksum	2	3 + x
ETX (0x03)	1	5 + x
<i>Reply: Third Party Data Ack</i>		

## Appendix A: Raw Message Data Example

Handshake Sync from External Device

0x55... 0x55... 0x55

Handshake Request from GO Device

0x02, 0x01, 0x00, 0x03, 0x08, 0x03

Handshake Confirmation from External Device  
(Device ID: 4108 = 0x0000100C)

0x02, 0x81, 0x04, 0x0C, 0x10, 0x00, 0x00, 0xA3, 0x88, 0x03

...

Third Party Data from External Device

(Status Data ID: 9999 = 0x270F, Data Value: 230 = 0x000000E6)

0x02, 0x80, 0x06, 0x0F, 0x27, 0xE6, 0x00, 0x00, 0x00, 0xA4, 0xF1, 0x03

Third Party Data Acknowledge from GO Device

0x02, 0x02, 0x00, 0x04, 0x0A, 0x03

## Appendix B: Sample Message Flow

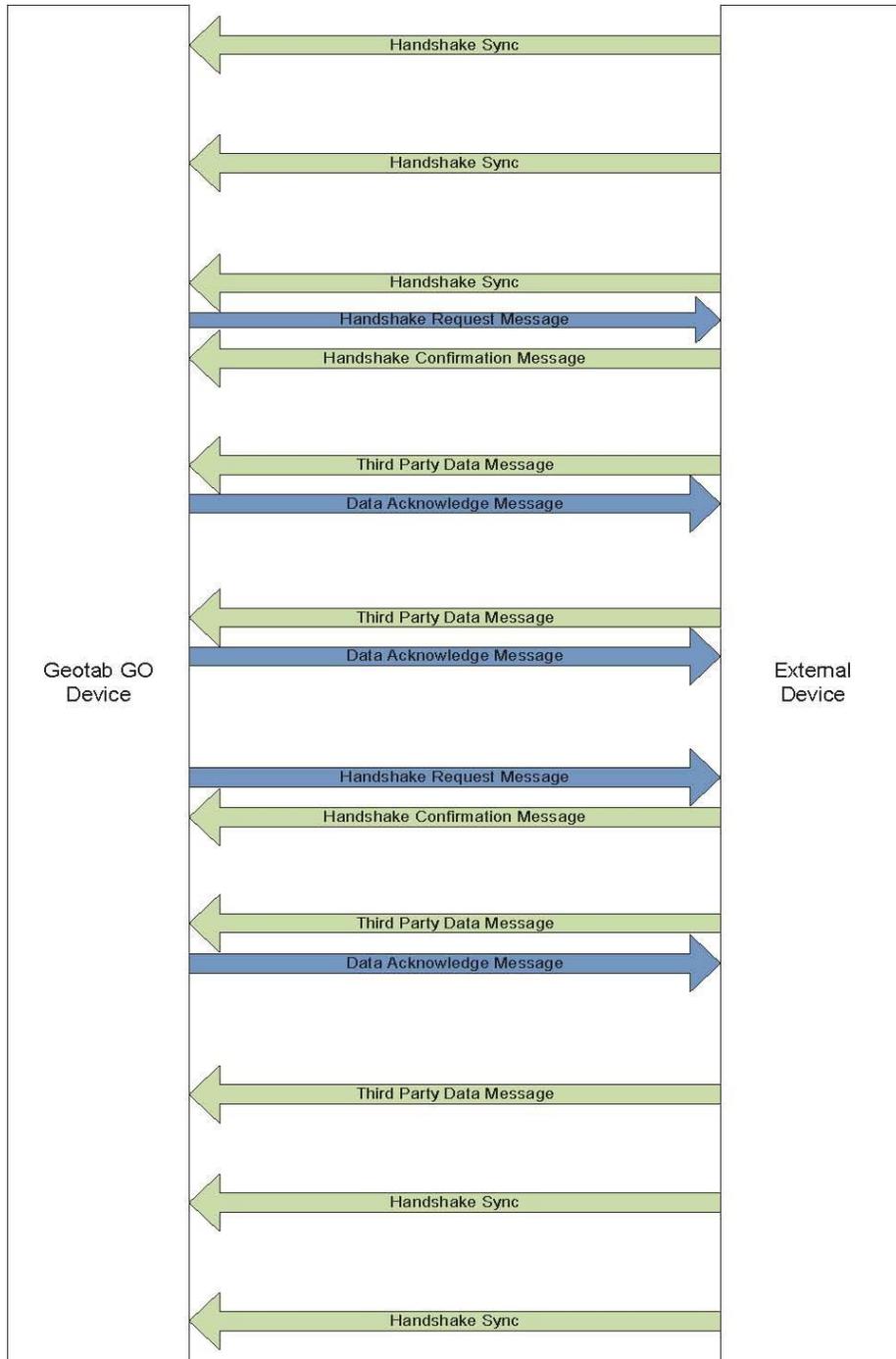


Fig 1: Example of External Device Message Flow