BTM44X – Health Device Profile

Application Note **v1.0**

Introduction

Health Device Profile (HDP) is available on the BTM44x module in both Agent and Manager roles as defined by the Continua Alliance (see www.continuaalliance.org for detailed descriptions).

There are two aspects to HDP, one is the transport layer, for which only Bluetooth is catered for by this module (the Continua Alliance has also ratified others, for example USB), and the other aspect is IEEE data encapsulation.

At the time of writing the Continua Alliance has also ratified the following transports: USB, Zigbee and Bluetooth Low Energy.

The Laird BTM44x module provides a tightly coupled integrated solution for a Weigh Scale Specialization Agent. More specializations will be provided in the future as and when there is demand via a firmware update.

It is assumed that the reader is familiar with all the HDP and IEEE documentation and relevant guidelines published by the Continua Alliance. For HDP it is assumed that the reader has access to the specification from the Bluetooth SIG. For IEEE it is assumed that the reader has access to the IEEE11073-20601 Optimised Exchange Protocol specification and the device specialization specifications 11073-10401 through to 10499.

For the Weigher Scale specialization embedded in the module the specification is 11073-10415. The IEEE standards can be obtained from their website (http://standards.ieee.org) and the Bluetooth HDP specification can be obtained from www.bluetooth.org.

The IEEE data specialization along with the Bluetooth physical transport as defined in the appropriate specifications is very dry and difficult to understand; it would be pointless to reproduce that information here verbatim. However, an attempt is made to describe it from the module's usage point of view where the module and the functionality it provides is treated in a black box manner.

Getting Started: Sample Message Sequence in AT Mode

In a typical weigh scale HDP usage scenario, the sequence of commands (red) and responses (blue) from the module are as follows, where it assumed that the both ends are assumed to be in AT mode.

To reproduce this sequence all that is required are two BTM44x modules configured in AT protocol mode and the Laird utility UwTerminal.exe, which is available on request.

If the module is in MP mode, set the value of S Reg 255 to 2 using the utility MpBtHost.exe provided by Laird.

```
=====================================    =====================================
AGENT                                    MANAGER
=====================================    =====================================

                                         ats9003=5
                                         OK
                                         ats9070=1
                                         OK
                                         at&w
                                         OK
ats9003=5
OK
ats9070=0
OK
at&w
OK
```

```
                         ============================
                                   PAIRING
                         ============================
AT+BTW0016a4fef005
OK
PAIR 0 0016A4FEF005                     PAIR 0 0016A4FEF004
                         ============================
                             REGISTER SDP RECORDS
                         ============================
                                        AT+HME4111,"scale"
                                        OK
                                        AT+HML
                                        OK
AT+HAE4111,"scale"
OK
AT+HAB0016a4fef005,4111
46492
OK
AT+HAL
OK
                         ============================
                                  ASSOCIATE
                         ============================
AT+HAA46392
HDA:ASSOCIATED
46392,4111,1500,4C414952444D4752
                                        HDM:ASSOCIATED
                                        2936,4111,1500,0016A4FEF004B538
OK
                         ============================
                           READ ATTIBUTES @ MANAGER
                         ============================
                                        AT+HMG29364,0,2628
                                        05DC
                                        OK
                         ============================
                            READ ATTIBUTES @ AGENT
                         ============================
AT+HAG46392,2646,0
0000004B
OK
                         ============================
                           WRITE ATTIBUTES @ AGENT
                         ============================
AT+HAS46392,2646,0,00000123
OK
AT+HAG46392,2646,0
00000123
OK
                         ============================
                                  SEND TIME
                         ============================
                                        AT+HMT29364,140B020C102D214E
                                        OK
HDA:TIME 46392,140B020C102D214E
```

```
                         ============================
                            SEND FIXED SCAN REPORT
                         ============================
AT+HAR46392,1234
                                        HDM:SCANREPORT 29364:1234
                                        O:1
                                        A:2646,00000123
                                        A:2448,2011021216453378
OK
                         ============================
                          SEND VARIABLE SCAN REPORT
                         ============================
AT+HAR46392,1234,2454,2448,2646,2375
                                        HDM:SCANREPORT 29364:1234
                                        O:1
                                        A:2454,06C3
                                        A:2448,2011021216453378
                                        A:2646,0000004B
                                        A:2375,8000
OK
                         ============================
                                 DISASSOCIATE
                         ============================
AT+HAD46392
OK
HDA:DISASSOCIATED 46392
                                        HDM:DISASSOCIATED 29364
```

GETTING STARTED: SAMPLE MESSAGE SEQUENCE IN MP MODE

Given two modules in MP protocol mode, the following sections illustrate a typical usage session which consists of a pairing, an association, scan report, time update from manager and disassociation.

The Laird utility MpBtHost.exe, available on request, was used to control both modules.

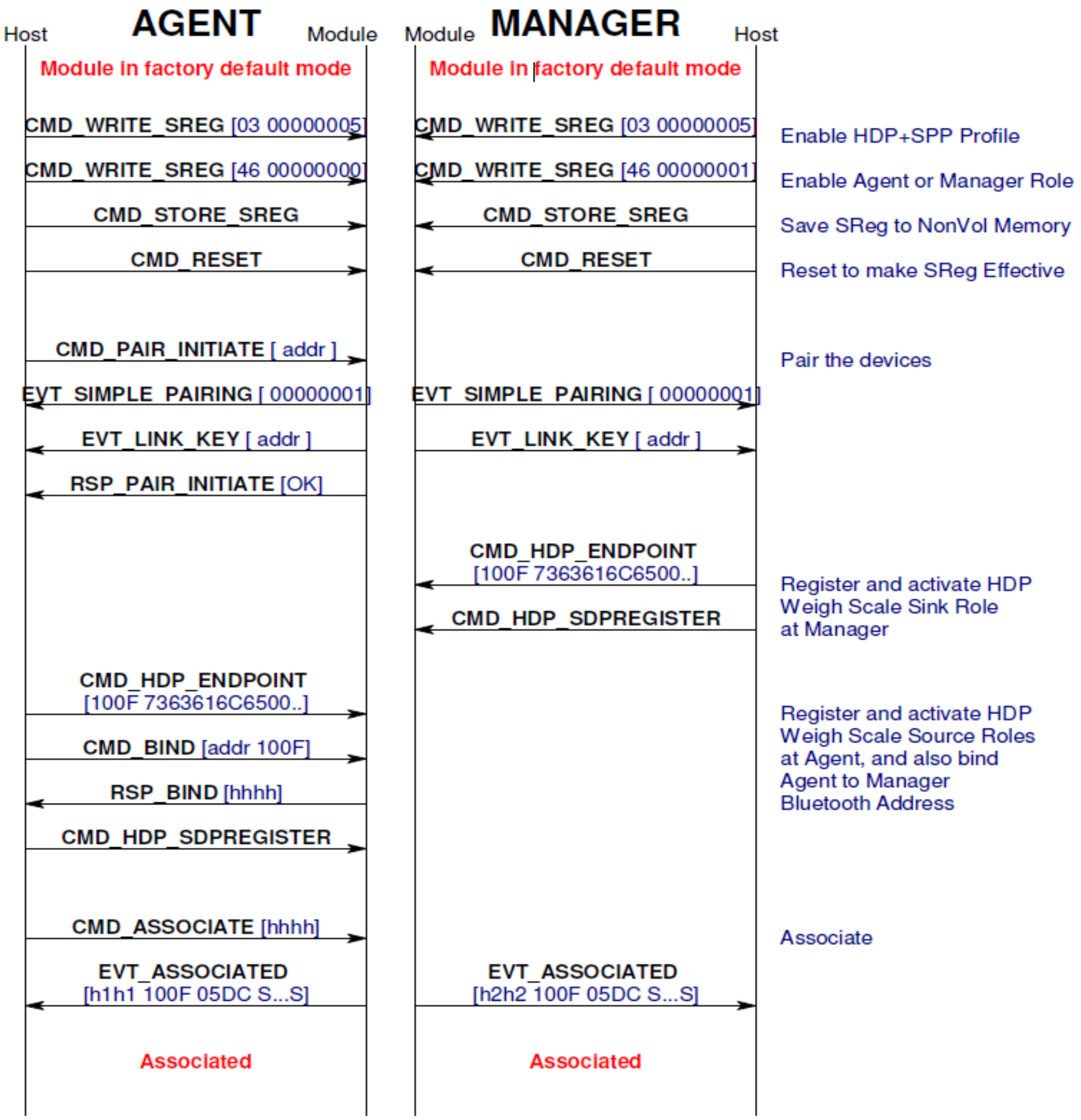

Host
AGENT
Module
Module
MANAGER
Host
Module in factory default mode
Module in factory default mode
CMD_WRITE_SREG [03 00000005]
CMD_WRITE_SREG [03 00000005]
Enable HDP+SPP Profile
CMD_WRITE_SREG [46 00000000]
CMD_WRITE_SREG [46 00000001]
Enable Agent or Manager Role
CMD_STORE_SREG
CMD_STORE_SREG
Save SReg to NonVol Memory
CMD_RESET
CMD_RESET
Reset to make SReg Effective
CMD_PAIR_INITIATE [addr]
Pair the devices
EVT_SIMPLE_PAIRING [00000001]
EVT_SIMPLE_PAIRING [00000001]
EVT_LINK_KEY [addr]
EVT_LINK_KEY [addr]
RSP_PAIR_INITIATE [OK]
CMD_HDP_ENDPOINT [100F 7363616C6500..]
CMD_HDP_SDPREGISTER
Register and activate HDP Weigh Scale Sink Role at Manager
CMD_HDP_ENDPOINT [100F 7363616C6500..]
CMD_BIND [addr 100F]
RSP_BIND [hhhh]
CMD_HDP_SDPREGISTER
Register and activate HDP Weigh Scale Source Roles at Agent, and also bind Agent to Manager Bluetooth Address
CMD_ASSOCIATE [hhhh]
Associate
EVT_ASSOCIATED [h1h1 100F 05DC S...S]
EVT_ASSOCIATED [h2h2 100F 05DC S...S]
Associated
Associated

Figure 1: Multipoint Mode message sequence

(continued on next page)

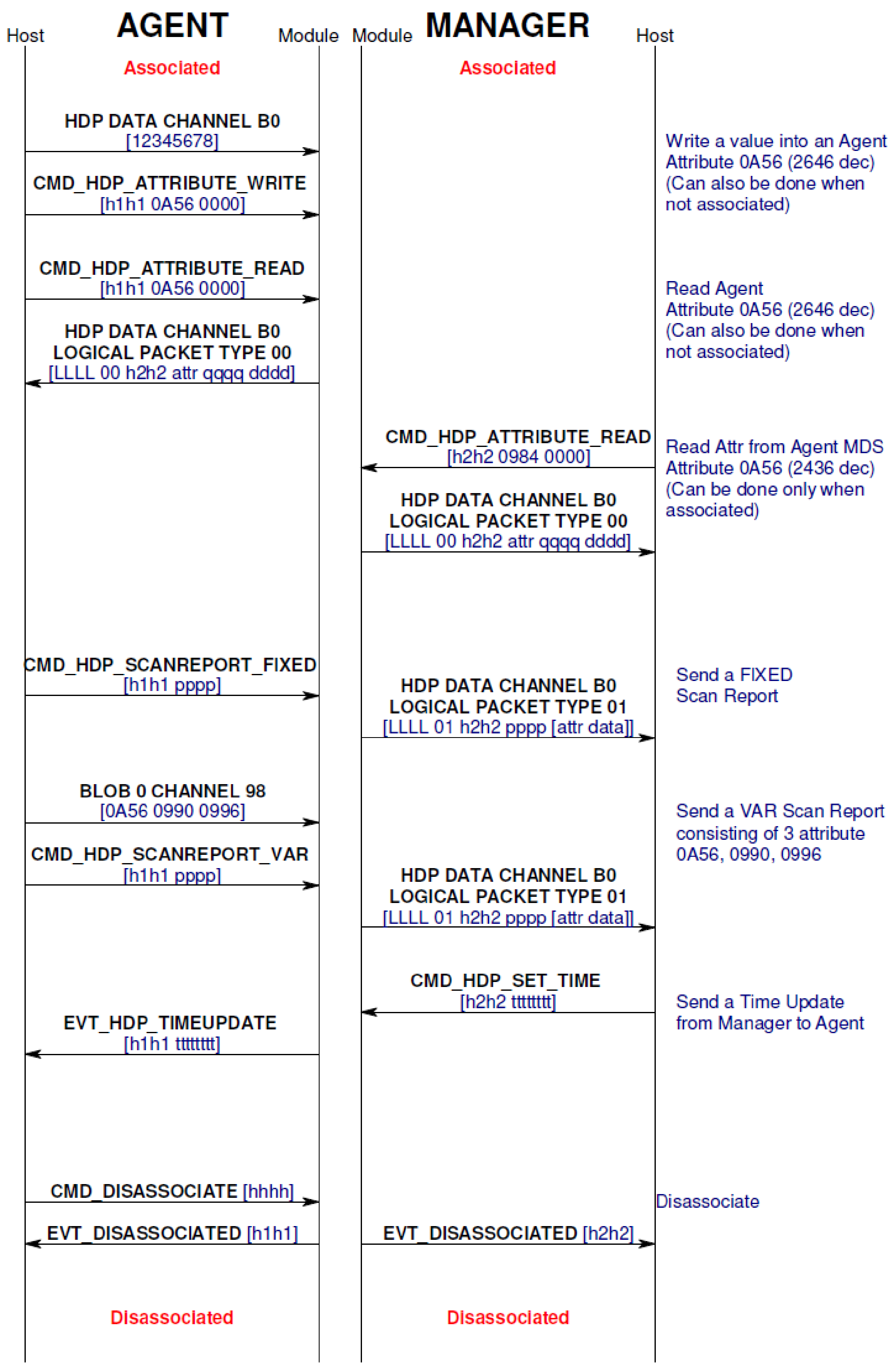


Figure 2: Multipoint Mode message sequence, continued

Sample Agent UART Traffic

This section shows UART traffic for a module operating as a HDP Weigh Scale agent communicating with a manager. It is NOT a log of the UART traffic for the message sequence chart (shown in the previous section).

```
**********************************************
**********************************************
<70 006.896 08 00 81 7F 00 00 00 0C
            EVT_STATUS
//State: RESET_GETADDR
>70 000.000 04 00 02 7F
            CMD_READ_BDADDR
<70 000.078 0B 00 02 7F 00 0016A4FEF000
            RSP_READ_BDADDR (MPSTATUS_OK)
>70 000.015 05 00 17 7F 00
            CMD_INFORMATION
//State: RESET_GETVER
<70 000.078 0E 00 17 7F 00 00 8100001300790673
            RSP_INFORMATION (MPSTATUS_OK)
<70 001.810 08 00 81 7F 00 01 01 0C
            EVT_STATUS
>70 057.018 09 00 04 7F 03 00000005
            CMD_WRITE_SREG
<70 000.109 0A 00 04 7F 00 03 00000005
            RSP_WRITE_SREG (MPSTATUS_OK)
>70 046.192 09 00 04 7F 46 00000000
            CMD_WRITE_SREG
<70 000.109 0A 00 04 7F 00 46 00000000
            RSP_WRITE_SREG (MPSTATUS_OK)
>70 004.790 04 00 05 7F
            CMD_STORE_SREG
<70 000.109 05 00 05 7F 00
            RSP_STORE_SREG (MPSTATUS_OK)
>70 024.523 07 00 29 7F 000000
            CMD_RESET
<70 001.918 08 00 81 7F 00 01 01 0C
            EVT_STATUS
>70 775.201 1C 00 10 7F 10 0016A4FEF001 3132333400000000000000000000000000
            CMD_PAIR_INITIATE
<70 007.300 0F 00 95 7F 0016A4FEF001 00 00000001
            EVT_SIMPLE_PAIRING
<70 000.156 0B 00 89 7F 0016A4FEF001 00
            EVT_LINK_KEY
<70 001.014 05 00 10 7F 00
            RSP_PAIR_INITIATE (MPSTATUS_OK)
>70 032.246 16 00 2E 7F 100F 7363616C650000000000000000000000
            CMD_HDP_ENDPOINT
<70 000.109 05 00 2E 7F 00
            RSP_HDP_ENDPOINT (MPSTATUS_OK)
>70 009.298 0E 00 32 7F 0016A4FEF001 100F 0000
            CMD_HDP_BIND
<70 000.125 07 00 32 7F 00 B538
            RSP_HDP_BIND (MPSTATUS_OK)
>70 026.037 04 00 2F 7F
            CMD_HDP_SDPREGISTER
<70 000.125 05 00 2F 7F 00
            RSP_HDP_SDPREGISTER (MPSTATUS_OK)
>70 011.918 06 00 31 7F B538
            CMD_HDP_ASSOCIATE
<70 000.109 05 00 31 7F 00
            RSP_HDP_ASSOCIATE (MPSTATUS_OK)
<70 002.013 08 00 81 7F 00 00 01 0C
            EVT_STATUS
<70 003.775 13 00 97 7F 00 B538 100F 05DC 4C414952444D4752
            EVT_HDP_ASSOCIATED
>70 029.235 06 B0 12345678
>70 018.127 0B 00 36 7F B538 0A56 0000 CD
            CMD_HDP_ATTRIBUTE_WRITE
<70 000.109 08 00 36 7F 00 CD 0004
            RSP_HDP_ATTRIBUTE_WRITE (MPSTATUS_OK)
>70 013.697 0B 00 35 7F B538 0A56 0000 AB
```

```
            CMD_HDP_ATTRIBUTE_READ
<70 000.109 0F B0 000D00B5380A56000012345678
//HDP Channel : ATTRIBUTE for handle=46392 Attr=2646 QualifierId=0} VALUE = 12345678
<70 000.016 08 00 35 7F 00 AB 0004
            RSP_HDP_ATTRIBUTE_READ (MPSTATUS_OK)
>70 042.900 09 00 33 7F B538 1234 AB
            CMD_HDP_SCANREPORT_FIXED
<70 000.421 08 00 33 7F 00 B538 AB
            RSP_HDP_SCANREPORT_FIXED (MPSTATUS_OK)
>70 021.185 08 B0 0A5609900996
>70 014.571 0A 00 34 7F B538 1234 CD 00
            CMD_HDP_SCANREPORT_VAR
<70 000.109 08 00 34 7F 8F B538 CD
            RSP_HDP_SCANREPORT_VAR (MPSTATUS_ATTRLIST_INVALID)
>70 035.365 0A 98 34 7F 0A56 0990 09 96
            CMD_HDP_SCANREPORT_VAR
>70 008.752 0A 00 34 7F 0A56 0990 09 96
            CMD_HDP_SCANREPORT_VAR
<70 000.125 08 00 34 7F 8B 0A56 09
            RSP_HDP_SCANREPORT_VAR (MPSTATUS_INVALID_BLOBID)
>70 019.906 08 98 0A5609900996
>70 012.823 0A 00 34 7F B538 1234 CD 00
            CMD_HDP_SCANREPORT_VAR
<70 000.296 08 00 34 7F 00 B538 CD
            RSP_HDP_SCANREPORT_VAR (MPSTATUS_OK)
<70 043.166 0E 00 98 7F B538 140B020C102D214E
            EVT_HDP_TIMEUPDATE
>70 008.486 06 00 30 7F B538
            CMD_HDP_DISASSOCIATE
<70 000.125 05 00 30 7F 00
            RSP_HDP_DISASSOCIATE (MPSTATUS_OK)
<70 000.125 07 00 96 7F 00 B538
            EVT_HDP_DISASSOCIATED
```

Sample Manager UART Traffic

This section shows UART traffic for a module operating as a HDP manager communicating with a Weigh Scale agent.

```
**********************************************
**********************************************
<71 003.900 08 00 81 7F 00 00 00 0C
            EVT_STATUS
//State: RESET_GETADDR
>71 000.000 04 00 02 7F
            CMD_READ_BDADDR
<71 000.094 0B 00 02 7F 00 0016A4FEF001
            RSP_READ_BDADDR (MPSTATUS_OK)
>71 000.015 05 00 17 7F 00
            CMD_INFORMATION
//State: RESET_GETVER
<71 000.078 0E 00 17 7F 00 00 8100001300790673
            RSP_INFORMATION (MPSTATUS_OK)
<71 001.810 08 00 81 7F 00 01 01 0C
            EVT_STATUS
>71 062.806 09 00 04 7F 03 00000005
            CMD_WRITE_SREG
<71 000.109 0A 00 04 7F 00 03 00000005
            RSP_WRITE_SREG (MPSTATUS_OK)
>71 022.449 09 00 04 7F 46 00000001
            CMD_WRITE_SREG
<71 000.109 0A 00 04 7F 00 46 00000001
            RSP_WRITE_SREG (MPSTATUS_OK)
>71 016.209 04 00 05 7F
            CMD_STORE_SREG
<71 000.109 05 00 05 7F 00
            RSP_STORE_SREG (MPSTATUS_OK)
```

```
>71 011.310 07 00 29 7F 000000
            CMD_RESET
<71 000.717 08 00 81 7F 00 00 00 0C
            EVT_STATUS
<71 001.950 08 00 81 7F 00 01 01 0C
            EVT_STATUS
<71 787.103 0F 00 95 7F 0016A4FEF000 00 00000001
            EVT_SIMPLE_PAIRING
<71 000.188 0B 00 89 7F 0016A4FEF000 00
            EVT_LINK_KEY
>71 018.564 16 00 2E 7F 100F 7363616C6500000000000000000000
            CMD_HDP_ENDPOINT
<71 000.125 05 00 2E 7F 00
            RSP_HDP_ENDPOINT (MPSTATUS_OK)
>71 005.413 04 00 2F 7F
            CMD_HDP_SDPREGISTER
<71 000.125 05 00 2F 7F 00
            RSP_HDP_SDPREGISTER (MPSTATUS_OK)
<71 067.751 08 00 81 7F 00 00 01 0C
            EVT_STATUS
<71 002.028 13 00 97 7F 01 72B4 100F 05DC 0016A4FEF000B539
            EVT_HDP_ASSOCIATED
>71 093.210 0B 00 35 7F 72B4 0984 0000 AB
            CMD_HDP_ATTRIBUTE_READ
<71 000.110 15 B0 00130072B40984000000080016A4FEF000B539
//HDP Channel : ATTRIBUTE for handle=29364 Attr=2436 QualifierId=0} VALUE =
00080016A4FEF000B539
<71 000.000 08 00 35 7F 00 AB 000A
            RSP_HDP_ATTRIBUTE_READ (MPSTATUS_OK)
<71 011.419 23 B0 00210172B4123401000001010A5600047856341201099000080000000000000000
//HDP Channel : SCAN REPORT handle=29364 personId=4660, reports=1
//                O:1 (0001)
//                A:2646 (0A56),<len=4> 78563412
//                A:2448 (0990),<len=8> 0000000000000000
<71 113.210 47 B0
00450172B4123401000001010A5600047856341201099000080000000000000000010996000206C3010A56
0004785634120109900008000000000000000001099600020 6C3
//HDP Channel : SCAN REPORT handle=29364 personId=4660, reports=1
//                O:1 (0001)
//                A:2646 (0A56),<len=4> 78563412
//                A:2448 (0990),<len=8> 0000000000000000
//                A:2454 (0996),<len=2> 06C3
//                A:2646 (0A56),<len=4> 78563412
//                A:2448 (0990),<len=8> 0000000000000000
//                A:2454 (0996),<len=2> 06C3
>71 043.056 0E 00 37 7F 72B4 140B020C102D214E
            CMD_HDP_SET_TIME
<71 000.109 05 00 37 7F 00
            RSP_HDP_SET_TIME (MPSTATUS_OK)
<71 008.721 07 00 96 7F 01 72B4
            EVT_HDP_DISASSOCIATED
<71 002.059 08 00 81 7F 00 01 01 0C
            EVT_STATUS
```

IEEE 'BLACK BOX' MODEL

A traditional health care environment consists of patients, health professionals, measuring instruments and medical records. Where records update from measuring instruments is either done manually or automatic.

Over the years, many suppliers of measuring instruments have provided proprietary methods for getting the data stored in records with the result that there is little standardization of data formats.

It has always been the role of the health professionals to 'transcribe' data from the various instruments into the archive records. This manual process presents a risk of transcription errors, leading manufacturers to provide even more proprietary automation solutions.

One of the aims of the Continua Alliance is to simplify this arrangement via guidelines and a certification process to ensure consistency and interoperability of measuring instruments and data recording procedures.

The Continua Alliance is uninterested in dictating how instruments (or Agents) are physically designed, as that is best left to the engineers who know how to design them.

Instead they have specified abstract data models for the various types of instruments, which they refer to as Data Specializations, and how they shall convey the data to an entity called a 'Manager' that can be used to consistently store the data in an archive.

The logic is that a measuring instrument abstractly decomposes into a list of data objects, and any procedure defined to act on the measuring instrument consists of reading and writing to that list of data objects. To manage that model, the list of data objects are containerized so sets of objects offer a better working model.

In the Continua Alliance terminology each data object is called an Attribute and the containers are given appropriate labels and handles for reference via the interrogation protocol.

Examples of data specialization abstract models exist for Weigh Scales, Thermometers, Glucose meters, Blood Pressure meters, and ECGs. More will become available as they progress through various stages in appropriate working groups within the alliance. Any Continua Alliance member is free to recommend creation of data specializations as needed. Once ratified, the end result is always an abstract data model which defines what data is pertinent for that instrument and how it shall be presented to the real world.

Abstract Data Model

From a developer's perspective, an abstract data model for an IEEE data specialization can best be described as a collection of arrays of different types of data (used interchangeably in this document with "attributes").

Each attribute is unambiguously defined to consist of a unique 16 bit tag, a type and the actual value. There is no reliance on any programming language in the definition. It is purely a data model.

As a minimum there shall be one array of attributes called the Medical Device System (MDS) representing the properties and services of the device, independent of its health data capabilities and status, and one array of attributes called the Numeric (NU) which contains actual episodic measurements. There is also a collection called RT-SA which represents continuous samples or waveforms, referenced in the ECG data specialization.

The reader is advised to refer to the IEEE11073-20601 standard for a definitive list of other collections, described under the general heading of "Domain Information Model" (DIM).

You should visualise an HDP agent as a collection of data records that can be read from and written to locally under program control. Each data point is identified by its 16 bit tag and the published data type.

This is analogous to a standard database table with 4 fields in each record. Three fields called 'Tag', 'Type' and 'Value' and the fourth field being called 'Collection Name' such as MDS or NU or RT-SA.

You should further visualise this 'database' as being accessible from a Manager over a physical transport media such as Bluetooth or USB using a Continua Alliance defined protocol and for this module locally via a UART using a Laird defined protocol so that the variable attributes can be populated appropriately.

The procedure a Manager shall use to gain access to that database of attributes is rigidly defined and standardised via a 'Service Model' using an association state machine defined in the IEEE11073-20601 standard. This Service Model is transparently encapsulated in the Laird Module and the user is encouraged to think of it in terms of a black box whose internal details are not relevant.

The picture that emerges for the Laird module user who requires a data specialization is that of a black box consisting of that conceptual database with a 4 field table and an 'engine' that implements the association service model over Bluetooth. It facilitates mirroring that database at the HDP manager end.

This vastly simplifies the design and development of a health instrument that has a requirement to be Continua Alliance certified. The hope is that, the user is only required to have a general idea about the content of the IEEE 11073-20601 and the data specialization IEEE11073-104xx standards.

The rest of this document details how the black box is controlled and manipulated. Note that initially only a Weigh Scale data specialization as defined in 11073-10415 is embedded in the black box.

"Embedded" implies that the MDS and NU collections are pre-defined as per the standard and the attributes with changeable values are exposed via the module's UART interface through a Laird defined protocol.

In future, it is hoped that a generic API will be exposed which allows any data specialization to be downloaded and tested. Until that generic API is made available, if you require a specific data specialization, contact Laird with that request.

Note that once you have a working instrument using this module, it is up to you to obtain Bluetooth Listing (quoting the QDID of the Laird module) prior to Continua Alliance testing and certification.

HDP Agent Model

From a firmware perspective the HDP Agent implementation in the Laird modules is as shown in Figure 3.

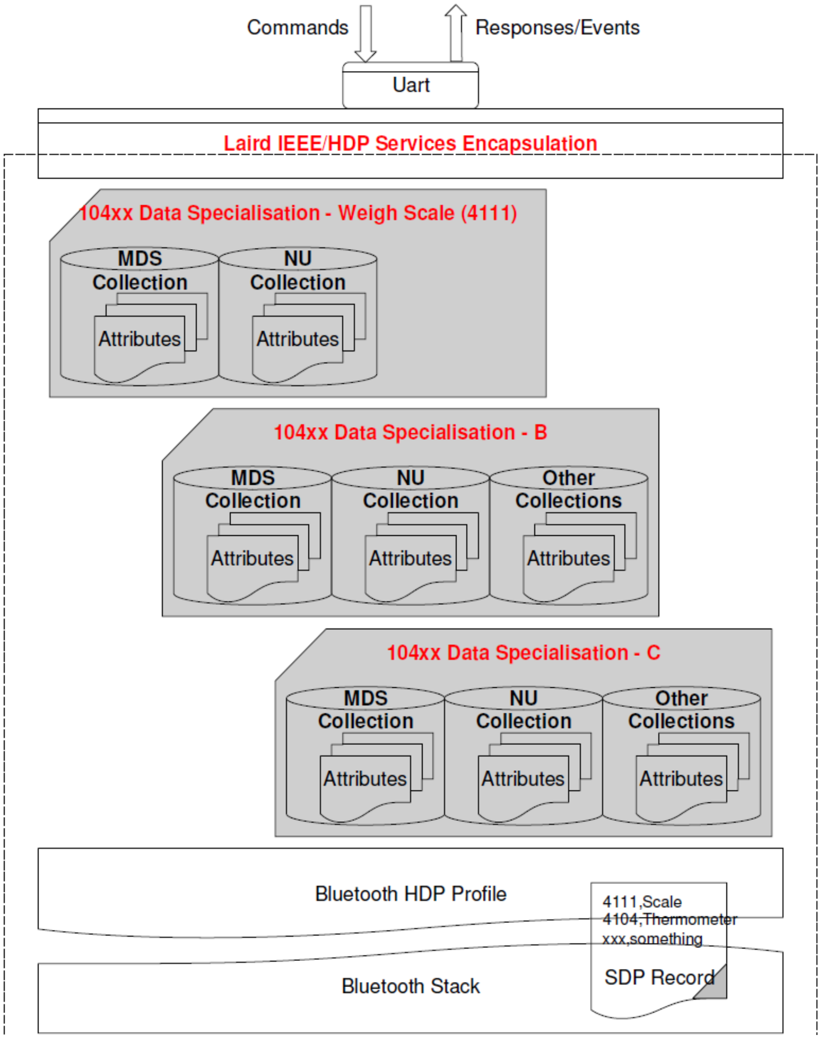


Figure 3: Black box agent model

The diagram shows that the Agent model consists of the Bluetooth communications stack which consists of an SDP record that exposes to the outside world the data specializations it is capable of, a 'Laird IEEE/HDP Service Encapsulation' layer which is used to relay commands and responses to the host and 0 or more instances of Data Specializations. At the time of the first release of the firmware only a Weigh Scale specialization is offered.

All UART commands, AT or MP mode, are provided so that the various entities in the black box can be controlled or interrogated. Given there can be many agent specializations embedded in the firmware, they are identified in various commands using a 16bit handle which is dynamically allocated.

Weigh Scale Data Specialization

The Weigh Scale Specialization (nominal code 4111 decimal) is embedded in the firmware is shown as below and it contains a MDS and an NU object.

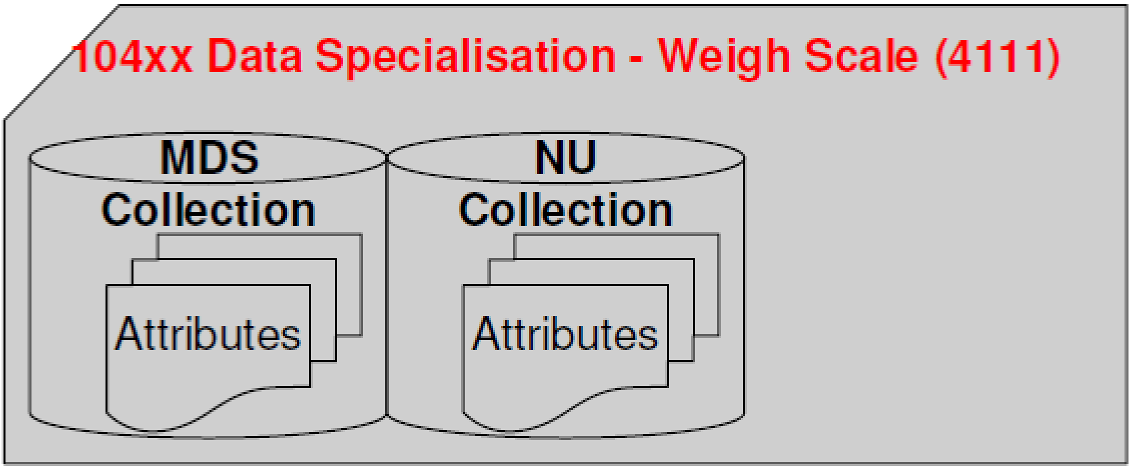


Figure 4: Weight scale specialization

The MDS object is defined in the firmware with the following attributes, where attributes in red can be modified over the UART.

Attribute Tag		Data Type	Comments
MDC_ATTR_ID_HANDLE	2337	HANDLE	Always: 0
MDC_ATTR_SYS_TYPE_SPEC_LIST	2650	TYPE_SPEC_LIST	Const
MDC_ATTR_ID_MODEL	2344	SYSTEM_MODEL	Var:SystemModel
MDC_ATTR_SYS_ID	2436	OCTET_STRING	Var:SystemId
MDC_ATTR_DEV_CONFIG_ID	2628	CONFIG_ID	1500 (0x05DC)
MDC_ATTR_ATTRIBUTE_VAL_MAP	2645	ATTR_VAL_MAP	Const
MDC_ATTR_ID_PROD_SPECN	2349	PROD_SPEC	Const
MDC_ATTR_TIME_ABS	2439	ABSOLUTE_TIME	Var:Time
MDC_ATTR_MDS_TIME_INFO	2629	MDS_TIME_INFO	Const
MDC_ATTR_POWER_STAT	2389	POWER_STATUS	Var:PowerStatus

The NU object is defined with the following attributes:

Attribute Tag		Data Type	Comments
MDC_ATTR_ID_HANDLE	2337	HANDLE	Always: 1
MDC_ATTR_ID_TYPE	2351	TYPE	Const

MDC_ATTR_METRIC_SPEC_SMALL	2630	METRIC_SPEC_SMALL	Const
MDC_ATTR_UNIT_CODE	2454	OID_TYPE	Var:Weight Units
MDC_ATTR_ATTRIBUTE_VAL_MAP	2645	ATTR_VAL_MAP	2646,2448
MDC_ATTR_TIME_STAMP_ABS	2448	ABSOLUTE_TIME	Var:Time
MDC_ATTR_NU_VAL_OBS_SIMP	2646	SIMPLE_NU_OBS_VAL	Var:Weight
MDC_ATTR_NU_ACCUR_MSMT	2378	FLOAT_TYPE	Always: 1
MDC_ATTR_MSMT_STAT	2375	MEASUREMENT_STATUS	Var:Measurement Status

The attributes commented as 'variables' are exposed to the host for reading and writing via the UART interface using AT+HAG and AT+HAS commands respectively in AT mode. In MP Mode the read/write commands are CMD_HDP_ATTRIBUTE_READ and CMD_HDP_ATTRIBUTE_WRITE respectively.

The variable attributes highlighted in red above are identified by the host on the UART interface using an attribute ID and an additional sub ID. The concept of 'sub ID' is a Laird artefact and is not part of any IEEE standard, but the attribute ID will in most cases be the same as those defined in the IEEE standard.

The complete list of variable attributes for the Weigh Scale specialization is as per the table below and should be used with the agent attribute read/write commands AT+HAG and AT+HAS (or CMD_HDP_ATTRIBUTE_READ and CMD_HDP_ATTRIBUTE_WRITE) .

PLEASE NOTE: the attribute values passed back and forth from the host are NOT validated in any way by the firmware in the Laird module. It is up to the host to ensure that the correct data is written into an attribute. Illegal values will be picked up at time of Continua Alliance certification testing and will prevent certification.

Table 1: Variable Attributes in Weigh Scale Specialization

Attribute Name (See IEEE spec for format)	ATR ID	Sub ID	Size in Bytes
Weight	2646	0	4
Weight Units	2454	0	2
Time	2448	0	8
Power Status	2389	0	2
Measurement Status	2375	0	2
System ID	2436	0	8
System Model – Product name	2344	0	12
System Model – Model name	2344	1	16
Serial Number	2349	0	8

All the attributes in the table above are multibyte and the reader should defer to the IEEE specifications with regards the format of the data. In terms of endienness, the reader should try determine that via trial and error by seeing the values appear at the manager.

Agent Related HDP Commands

This section describes all the commands used to manage the Agent role for HDP (both AT and MP mode commands are described) after the following table summarizing those commands.

AT Mode Commands	MP Mode Commands	Comments
AT+HAEiiii,"name"	CMD_HDP_ENDPOINT	Affects an entry in the SDP record
AT+HAL	CMD_HDP_SDPREGISTER	Register / activate HDP service record
AT+HABeeeee,iiii	CMD_HDP_BIND	Create agent data specialization of

		type iiii which connects to manager with address 'eeeeee'
AT+HAAhhhh	CMD_HDP_ASSOCIATE	Yields a connection to the manager
AT+HARhhhh,pppp[…]	CMD_HDP_SCANREPORT_FIXED CMD_HDP_SCANREPORT_VAR	In MP, the attribute list for var report is supplied via a blob channel.
AT+HAGhhhh,aaaa,ssss	CMD_HDP_ATTRIBUTE_READ	Read an attribute value. In MP mode the data value is sent in data channel B0
AT+HAShhhh,aaaa,ssss,dddddd	CMD_HDP_ATTRIBUTE_WRITE	Write an attribute value. In MP mode the data value is taken from data channel B0
AT+HADhhhh	CMD_HDP_DISASSOCIATE	The connection gets dropped

Associate with HDP Manager

AT Command:	AT+HAAhhhh
AT Response:	<cr,lf> OK<cr,lf> Or <cr,lf>ERROR nn<cr,lf> Or <cr,lf>HAD:ASSOCIATE xxxx<cr,lf>OK<cr,lf> Or <cr,lf> HAD:DISASSOCIATE xxxx<cr,lf>ERROR nn<cr,lf>
Description:	This command establishes a connection to an HDP manager and associates the agent with it to enable attribute data exchange. The Bluetooth address of the HDP manager and the agent specialization that needs to be associated is defined by the handle 'hhhh' which is pre-obtained using command AT+HAB. This command waits for the procedure to complete successfully or otherwise before responding with OK or ERROR. If the agent is already associated then the immediate response is OK. If the agent is not already associated then a Bluetooth connection is initiated and as soon as a connection is established the association state machine procedes to negotiate a configuration and ultimately confirms association. If association is successful, a "HDA:ASSOCIATE ….." asynchronous response is sent to the host and if association fails (because BT connection failed or configuration could not be negotiated) then the async response "HDA:DISASOCIATE …" is sent to the host. An OK or ERROR response terminates the procedure. Note if S Reg 9071 is non-zero, then there is an automatic disassociation after a time specified by that register. The timer is restarted every time a scan report is sent to the manager. SReg Required Settings: Bit 2 set in S9003 and S9070=0.
MP Command:	CMD_HDP_ASSOCIATE If association is successful prior to receiving the response, the host is sent EVT_ASSOCIATED, which will contain the config id of the MDS configuration that was negotiated and the system id of the peer. See EVT_ASSOCIATED for further details. If association was not successful, because for example, the manager is not in range or the Bluetooth device specified in CMD_HDP_BIND does not offer HDP services, then the EVT_DISASSOCIATED is sent to the host prior to the response message. This is shown in the message sequence diagram below.

Disassociate an agent

AT Command:	AT+HADhhhh
AT Response:	<cr,lf>OK<cr,lf> Or <cr,lf>> HAD:DISASSOCIATE xxxx<cr,lf>OK<cr,lf> Or <cr,lf>ERROR nn<cr,lf>
Description:	This command disassociates an agent identified by the handle hhhh, from a manager. It waits for the procedure to complete, successfully or not, before responding with OK or ERROR. If the agent is already disassociated, the response is an immediate OK. SReg Required Settings: Bit 2 set in S9003 and S9070=0
MP Command:	CMD_HDP_DISASSOCIATE The command provides a handle parameter obtained via the associate command.

Create an Endpoint

AT Command:	AT+HAEiiii,"name"
AT Response:	<cr,lf>OK<cr,lf> Or <cr,lf>ERROR nn<cr,lf>
Description:	This command must be issued prior to sending the AT+HAL commands and is used to include the data specialization endpoint 'iiii' with the string "name" as a HDP source (agent) in the SDP record that will be exposed to potential peers. This in effect tells potential peers that the device offers a 'iiii' data specialization source.
MP Command:	Without this entry in the SDP record, a manager is unable to make a connection to the HDP agent, but it is rare for a manager to initiate connections in typical use cases. SReg Required Settings: Bit 2 set in S9003 and S9070=0

Read an Attribute

AT Command:	AT+HAGhhhh,aaaa,ssss
AT Response:	<cr,lf>hhhhhhhhhhhhh<cr,lf>OK<cr,lf> Or <cr,lf>ERROR nn<cr,lf>
Description:	This command reads (gets) the value of one of the variable attributes identified by 'aaaa' and sub id 'ssss' in the attribute collections for that agent identified by the handle 'hhhh'. For the embedded weigh scale data specialization this command reads the value of an attribute listed in Table 1. The value is always returned as a string of hexadecimal digits representing the binary value, and the size of that string will be even. Different attributes have different sizes. SReg Required Settings: Bit 2 set in S9003 and S9070=0
MP Command:	CMD_HDP_ATTRIBUTE_READ If the attribute exists, the data will be sent in a Logical HDP data packet transported in one or more physical data packets over channel 0xB0. Physical packets of incoming data in channel 0xB0 are viewed as a stream of data making up logical packets.

	See User's Manual for format of the data in the data channel.

Register and Activate and SDP Record

AT Command:	AT+HAL
AT Response:	<cr,lf>OK<cr,lf> Or <cr,lf>ERROR nn<cr,lf>
Description:	This command must be issued after sending at least one AT+HAE command. It is used to register and activate the SDP record with data supplied in AT+HAE commands. Without the SDP record, a manager is not able to make a connection to the HDP agent, although it is rare for a manager to initiate connections in typical usage scenarios. SReg Required Settings: Bit 2 set in S9003 and S9070=0
MP Command:	CMD_HDP_SDPREGISTER Internally, in the module, the SpecType and Name information previously cached in heap memory from CMD_HDPENDPOINT commands is transferred into a SDP record and submitted to the underlying stack. After this a peer will be able to see this device offering HDP services

Send a Scan Report

AT Command:	AT+HARhhhh,pppp[,aaaa[,aaaa[…]]]
AT Response:	<cr,lf> OK<cr,lf> Or <cr,lf>ERROR nn<cr,lf>
Description:	This command is used to trigger a scan report for person ID 'pppp' (16 bit decimal number) from the agent identified by handle 'hhhh'. If the agent is not associated with the bound manager (see AT+HAB) then it first triggers the start of an association. Once association exists, a scan report is sent. If [,aaaa[,aaaa[…]]] is absent from the command, the standard scan report as identified by the attribute `MDC_ATTR_ATTRIBUTE_VAL_MAP` in the NU collection is sent to the manager. Otherwise, the [,aaaa[,aaaa[…]]] can be a list of any attribute mentioned in the NU collection. For example, to send a scan report with just the measurement status, just specify the single value 2375. Note that only attributes mentioned in the NU collection are allowed. Any 'aaaa' value not found in the NU collection is silently ignored. SReg Required Settings: Bit 2 set in S9003 and S9070=0
MP Command:	CMD_HDP_SCANREPORT_FIXED or CMD_HDP_SCANREPORT_VAR A VAR report sends the values for a list of attributes in the NU Collection object to a manager where the list is pre-supplied by the host. If an attribute does not exist in the NU collection then it is silently ignored. The attribute list for VAR reports is provided by the host via the blob data channel for the BLOBID specified in the command (that is, channel number 0x98 plus BLOBID).

Write to an Attribute

AT Command:	AT+HAShhhh,aaaa,ssss,hhhhhhhhhhh
AT Response:	<cr,lf> OK<cr,lf> Or <cr,lf>ERROR nn<cr,lf>
Description:	This command is used to write (set)a new value 'hhhhhhhhhhh' to one of the variable attributes (identified by 'aaaa' and sub id 'ssss') in the attribute collections for the agent identified by the handle 'hhhh'. For the embedded weigh scale data specialization this command is used to write to an attribute listed in Table 1. The value 'hhhhhhhhh' is in hexadecimal and is NOT validated in any way apart from the requirement that it shall be twice the size in bytes as specified for that attribute. SReg Required Settings: Bit 2 set in S9003 and S9070=0
MP Command:	CMD_HDP_ATTRIBUTE_WRITE If the attribute exists, then the data is moved from channel 0xB0 to the attribute container variable. If the attribute does not exist, then the data is discarded. All the data in the channel is treated as the data for the attribure, that is, there is no qualifying information affixed to the data. In addition, if the attribute exists, but the data length does not match that of the attrivute, then the write fails (with an appropriate error code) and the data in the channel is discarded. Since all data is transparently treated in the module, the endienness of DATA[N] should be determined by trial and error with the aid of a HDP manager and finalized to be correct by the time the implementation is submitted for certification by the Continua Alliance. However, if the attribute data type is a 16 or 32 bit integer/float, then it is little endian.

Agent Related Asynchronous Responses and Events

This section describes all the asynchronous responses sent to the host by the HDP Agent.

Command:	No Command. This is a status message.
AT Async Response:	HDA:DISASSOCIATED hhhh
Description:	This response is sent to the host when an association attempt fails (as a result of AT+HAA or AT+HAR commands) or when an association is terminated by either AT+HAD or due to loss of Bluetooth connectivity. The parameter 'hhhh' which is a 16 bit decimal number identifies the agent.
MP Event:	EVT_HDP_DISASSOCIATE

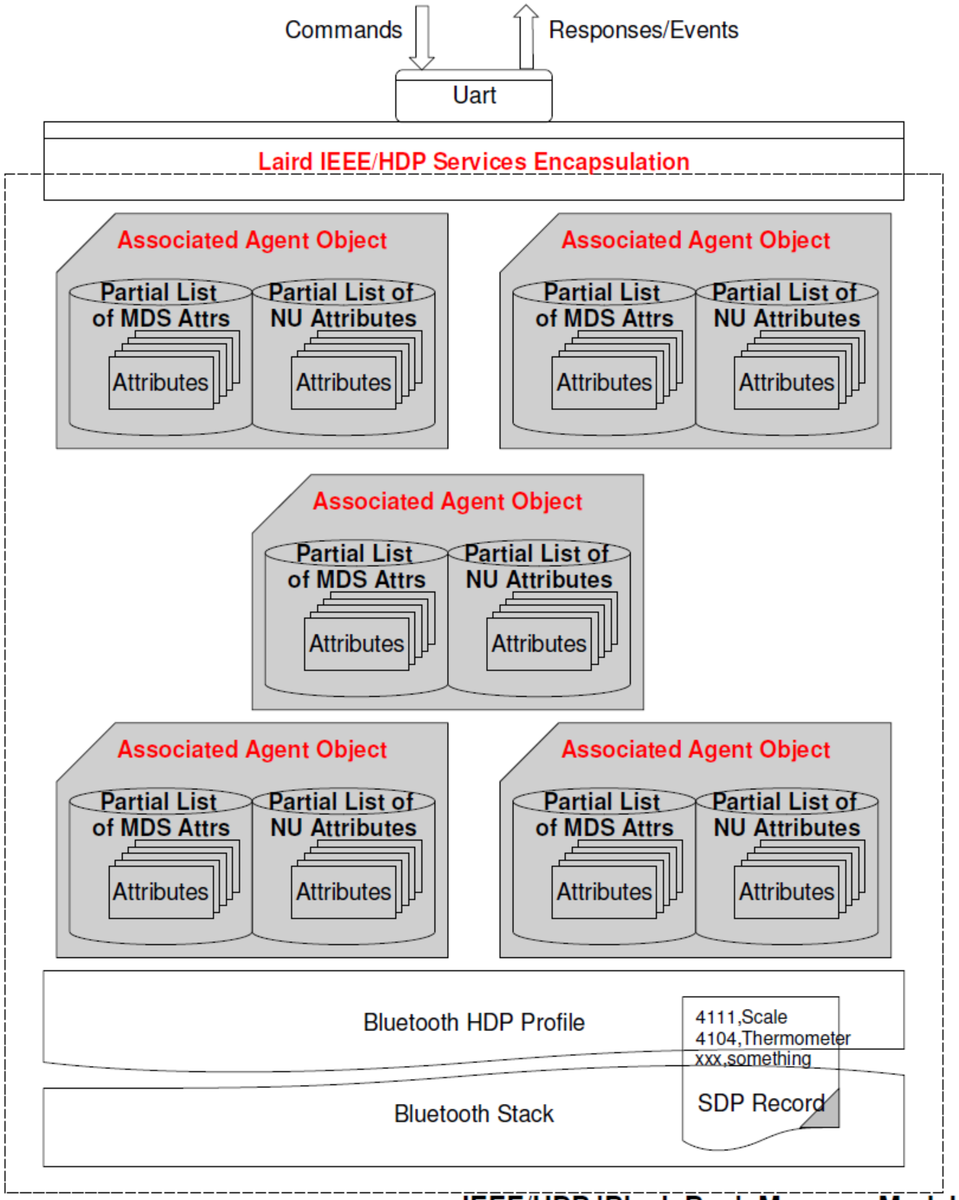
Command:	No Command. This is a status message.
Response:	HDA:ASSOCIATED hhhh,iiii,cccc,ssssssssssss
Description:	This response is sent to the host when a successful association happens for the agent identified by 'hhhh' (16 bit decimal number). For completeness, the data specialization nominal code 'iiii' (16 bit decimal) and the device configuration ID 'cccc' (16 bit decimal), that got negotiated with the manager, is also provided. The 16 character parameter 'ssssssss' specifies the system ID of the manager. Note: An agent data specialization can be basic as specified in the relevant spec, or one of the more enhanced ones which have more capabilities. For example, with the weigh scale, Laird have provided for the basic MDS collection, but it is possible to specify multiple MDS collection which expose more functionality (e.g. body mass

	index attributes). These extended collections are identified by configuration IDs and are offered to a manager during the association phase in descending order of complexity until the manager accepts one that it can work with.
MP Event:	EVT_HDP_DISASSOCIATE

Command:	No Command. This is a status message.
Response:	HDA:TIME hhhh,ccyymmddhhmmssaa
Description:	This response is sent to the host when a manager sends new time information by writing to the MDC_ATTR_TIME_ABS attribute in the MDS collection of the agent identified by 'hhhh'. The time ccyymmddhhmmssaa is a 16 character hexadecimal value which is encoded as follows: CC Century (e.g. 14==20) YY Year (e.g. 0B==11) MM Month (e.g. 0C==12) DD Day (e.g 1F==31) HH Hour (e.g. 17==23) MM Minutes (e.g 32==50) SS Seconds (e.g. 2F==47) AA Hundreths of seconds (e.g 63==99)
MP Event:	E.g. the data and time 12 Feb 2011, 16:45:33.78 is sent as 140B020C102D214E

HDP Manager Model

From a firmware perspective the HDP Manager implementation is as shown in the diagram below. The functionality is provided mainly to enable prototyping and regression testing of agent specializations. There are many far more capable HDP Managers available which are hosted on a PC. For example, the latest Toshiba Bluetooth Stack is HDP capable and there are imminent BlueZ releases for Linux PCs.



IEEE/HDP 'Black Box' Manager Model

The diagram above shows that the Manager model consists of the Bluetooth communications stack which consists of an SDP record that exposes to the outside world the data specializations it is capable of accepting as sinks, a 'Laird IEEE/HDP Service Encapsulation' layer which is used to relay commands and responses to the host and zero or more instances of Specialization Associations. These 'Associated Agent Objects' are transient and come into existence only when an agent successfully associates. The associated process results in the top few attributes in the MDS and NU collections being cached in the Manager which can be read by the host using the AT+HMG command after an association in AT mode and CMD_HDP_ATTRIBUTE_READ in MP mode.

Just like the Agent end, the Manager indicates via its SDP record which Data Specializations it is capable of sinking, and commands analogous to the ones provided for the Agent model have been provided to manipulate that SDP record content.

Manager Related AT Commands

This section describes all the commands used to manage the Agent role for HDP.

AT Mode Commands	MP Mode Commands	Comments
AT+HMEiiii,"name"	CMD_HDP_ENDPOINT	Affects an entry in the SDP record
AT+HML	CMD_HDP_SDPREGISTER	Register and activate HDP service record
AT+HMGhhhh,oooo,aaaa	CMD_HDP_ATTRIBUTE_READ	Read an attribute value. In MP mode the data value is sent in data channel B0
AT+HMThhhh,ttttt	CMD_HDP_SETTIME	Used at manager end to send new date/time to the agent

Create an Agent Endpoint

Command:	AT+HMEiiii,"name"
Response:	<cr,lf>OK<cr,lf> Or <cr,lf>ERROR nn<cr,lf>
Description:	This command must be issued prior to sending the AT+HML commands and is used to include the data specialization endpoint 'iiii' with the string "name" as a HDP sink in the SDP record that is exposed to potential agents. This in effect tells potential peers that the device offers a 'iiii' data specialization sink. Without this entry in the SDP record, an appropriate agent is not able to make a connection to the HDP Manager. SReg Required Settings: Bit 2 set in S9003 and S9070=1
MP Command:	CMD_HDP_ENDPOINT

Read an Attribute

Command:	AT+HMGhhhh,oooo,aaaa
Response:	<cr,lf>hhhhhhhhhhhhh<cr,lf>OK<cr,lf> Or <cr,lf>ERROR nn<cr,lf>
Description:	This command is used to read the value of one of the cached attributes for an agent identified by 'hhhh'. The parameter 'oooo' shall be 0 for attributes in the MDS collection cache and '1' for the NU collection cache and 'aaaa' is the id of the attribute to read. The value is always returned as a string of hexadecimal digits representing the binary value, and the size of that string is even. Different attributes have different sizes. SReg Required Settings: Bit 2 set in S9003 and S9070=1
MP Command:	CMD_HDP_ATTRIBUTE_READ

Register and Activate an SDP Record

Command:	AT+HML
Response:	<cr,lf>OK<cr,lf> Or <cr,lf>ERROR nn<cr,lf>
Description:	This command must be issued after sending at least one AT+HME command and is used to register and activate the SDP record with information supplied in the AT+HME commands. Without the SDP record, an agent is not able to make a connection to the HDP manager.

	SReg Required Settings: Bit 2 set in S9003 and S9070=1
MP Command:	CMD_HDP_SDPRECORD

Send a Time Update to Agent

Command:	AT+HMThhhh, ccyymmddhhmmssaa
Response:	<cr,lf> OK<cr,lf> Or <cr,lf>ERROR nn<cr,lf>
Description:	This command is used to send the current data and time to the associated agent identified by handle 'hhhh'. The time ccyymmddhhmmssaa is a 16 character hexadecimal value which is encoded as follows: CC Century (e.g. 14==20) YY Year (e.g. 0B==11) MM Month (e.g. 0C==12) DD Day (e.g 1F==31) HH Hour (e.g. 17==23) MM Minutes (e.g 32==50) SS Seconds (e.g. 2F==47) AA Hundreths of seconds (e.g 63==99) E.g. the data and time 12 Feb 2011, 16:45:33.78 is sent as 140B020C102D214E. SReg Required Settings: Bit 2 set in S9003 and S9070=1
MP Command:	CMD_HDP_SETTIME

Manager Related Asynchronous Responses

This section describes all the asynchronous responses sent to the host by the HDP Manager. Each response is framed by a <cr,lf> at the start and end.

AT Command:	No Command. This is a status message.
AT Response:	HDM:DISASSOCIATED hhhh
Description:	This response is sent to the host when an association is terminated by an agent or due to loss of Bluetooth connectivity. The parameter 'hhhh' which is a 16 bit decimal number identifies the agent.
MP Event:	EVT_HDP_DISASSOCIATE

AT Command:	No Command. This is a status message.
AT Response:	HDM:ASSOCIATED hhhh,iiii,cccc,ssssssssssss
Description:	This response is sent to the host when a successful association happens for the agent identified by 'hhhh' (16 bit decimal number). For completeness, the data specialization nominal code 'iiii' (16 bit decimal) and the device configuration ID 'cccc' (16 bit decimal), that got negotiated with the manager, is also provided. The 16 character parameter 'ssssssss' specifies the system ID of the agent. Note: An agent data specialization can be basic as specified in the relevant spec, or one of the more enhanced ones which have more capabilities. For example, with the weigh scale, Laird have provided for the basic MDS collection, but it is possible to specify multiple MDS collection which expose more functionality (e.g. body mass index attributes). These extended collections are identified by configuration IDs and are offered to a manager during the association phase in descending order of complexity until the manager accepts one that is acceptable.
MP Event:	EVT_HDP_ASSOCIATE

AT Command:	No Command. This is a status message.
AT Response:	HDM:SCANPERIOD hhhh<more>

Description:	This response is sent to the host when a scan report is received from agent identified by handle 'hhhh'. Since a scan report can consist of data values for many attributes, this report is formatted with embedded <lf> characters as follows:
	`<cr><lf>HDM:SCANREPORT hhhh:pppp`
	`<lf>O:oooo`
	`<lf>A:aaaa,ddd..ddd`
	`<lf>A:aaaa,ddd..ddd`
	…
	`<lf>O:oooo`
	`<lf>A:aaaa,ddd..ddd`
	`<lf>A:aaaa,ddd..ddd`
	…
	…
	`<cr><lf>`
	Where <lf>O: identifies the collection object (oooo == 1 for NU etc) and then subsequence <lf>A: lines consist of value pairs aaaa,ddd.ddd where 'aaaa' is the attribute nominal code and ddd...ddd is its value in hexadecimal. The size of the value for each attribute is specified in the IEEE standards.

MP Command:	This data is sent in data channel 0xB0 Data is sent in a data channel instead of an event packet because the amount of data is variable.

REVISION HISTORY

Revision	Date	Description	Approved By
1.0	19 May 2012	Initial Release	Jonathan Kaye