Table of Contents

DigiSnap Pro Modules

Within the DigiSnap Pro product line, there are multiple independent Modules, each with it's own internal functionality and means to communicate. Some Modules are hard-wired, and others are hot-plug-able.

The DigiSnap Pro Controller (DSPC) board is the main circuit board to which all other hardware devices attach. The basis for this board is a 9S08PT60 microcontroller, with code written in a mix of assembly and C

The DSPC handles most of the power and camera operations, and also implements the Module Hub function, managing the registry of all Modules, and acting as the communication pathway among them. The DSPC will internally register as one or more peer Modules, allowing control and communication from/to other Modules.

The Network board is connected to the DSPC as a fixed assembly, not hot-plug-able hardware. The Network board includes an embedded computer operating a Linux operating system. The embedded computer was selected for it's low power, wealth of resources, and form factor. The addition of the Network board enhances the DS Pro tremendously, adding USB host ports, a dedicated GPS port, Ethernet, WiFi and classic Bluetooth communications. The Network board allows for transfer of images, notification messaging, and remote access. In addition, the Network board provides a platform from which to operate a great range of advanced time-lapse operations, via direct USB control over the camera.

The Network board operates substantially independently of the DSPC, and only needs to exchange a few operations with the rest of the Modules of the DigiSnap Pro. The Network board will register as several Modules, expressing different functional groups. Each Module is intended to embody a distinct set of features.

Primary Modules

Some Primary Modules are hard-wired into the Module Hub, or are implemented in the same firmware as the Hub. They are not hot-pluggable hardware elements. Other Primary Modules may be embedded in the Network board, and implicitly connected when that board is connected.

Control Module

The Control Module (CTRLM) is the mechanism used to communicate and control the DSPC power and external I/O. The CTRLM is automatically registered by the Hub at power up.

Camera Module

The Camera Module (CAM) allows control over the power and monitoring functions of an SLR camera attached to the DSPC. Time-Lapse and Real-Time clock functions are also included. The CAM is automatically registered by the Hub at power up.

Network Control Module

The Network board implements a Network Control Module interface with connection to the Module Hub, allowing local operator control via a User Interface module over the operating parameters of the Network board.

In addition, if the Network board is connected to a Network, the remote operator will be able to communicate with all of the other Modules through the Network Control Module.

Network Access Module

The Network board implements a Network Access Module as well, allowing protected configuration of network access logins and passwords.

User Modules

During development of the DigiSnap Pro, there were a series of failed attempts at creating a user interface. An on-board display in conjunction with multiple buttons were used to create a graphic menuing interface, somewhat similar to what might be found on the back of a digital camera. In the end, we discovered that the DigiSnap Pro was less useful as a hand held device, given the many cabled connections, so the on-board user interface was abandoned.

There are three user interface paths intended for the DigiSnap Pro.

- Mobile (local wireless connection via Bluetooth Low Energy)
- Hardwired connection to a computer serial port
- Remote access via network

Bluetooth Low Energy Module

The DigiSnap Pro is designed to leverage the most user friendly and ubiquitous electronic interfaces ever developed... smartphones. smartphone accessories have traditionally been limited to those approved by the cell phone manufacturer, leading to creative solutions such as audio interfaces for exchanging data between a phone and accessory. This was seriously considered, but it is of course, wired, and there is a strong desire to implement a wireless interface, so the operator does not need to climb a pole, etc., to access the camera system. Modern phones support Bluetooth Smart/Low Energy (BLE), which removes traditional iOS restrictions on accessories. BLE provides a means to discover nearby devices and exchange arbitrary data, wirelessly, while being very power-efficient for the peripheral. For these reasons, BLE was chosen as the primary user interface for the DigiSnap Pro.

A BLE module can be connected to the DSPC, providing a means for a wireless user interface to the DigiSnap Pro. Apps will be developed for the Android and iOS operating systems to configure and control the DigiSnap Pro. The BLEM hardware connects to a DSPC Module Port like any other ModuleAccessory Module.

Debug Module

A menu driven 'DOS-like' program for windows PCs provides a way to view and modify the parameters of the other modules in the system. This program, interfaces via serial port on the PC, connected to one of the DSPC Accessory ModuleAccessory Module ports.

Accessory Modules (AMs)

There are a variety of other Accessory Modules (AMs) that will be developed for the DigiSnap Pro system, as time allows and interest dictates.

1) GPS – the Network board includes a connector for use with a GPS receiver. The Network board will register a new GPS Module when the GPS receiver is connected. A dedicated GPS Module could also be created to connect directly to the DSPC.

2) Motion control is a popular technique to incorporate in time-lapse projects. Harbortronics had a very early integrated time-lapse motion control rig (Snap360), and we plan to re-enter this area of the field.

3) One of the 'Holy Grail' problems in time-lapse is controlling a camera for a smooth exposure transition from day to night. We have an idea for an integrated light sensor which may help.

4) Many researchers and just-plain-folk have used motion sensing cameras, and we plan to offer motion sensing modules to our system, to implement a very high quality and featured 'trail cam'.

Other specialized applications require other sensor technologies for triggering cameras, and our ability to quickly develop custom sensing modules based on a common structure may be quite useful.

Each AM will contain a small microcontroller to communicate with the Module Hub and onto other Modules. When initially connected to the DigiSnap Pro, it will be registered for general access.

Module Registry

The Module Hub maintains an internal registry of the connected Modules, and their function. There is a maximum of 16 possible connected Modules, and 255 potential unique Module Type IDs. If the Module Type field associated with an address is set to 0, there is no Module associated/connected with address. A list of functional types, their data structures, etc., will be published when that Module is developed.

The registry consists of 16 items, each 16 bits wide (32 bytes total). The Module address is assigned to given Module Type upon registration. The Port # indicates the hardware port to which a Module is connected/associated. The Module Type ID indicates what kind of Module it is, as reported by the device at registry.

The top two bits of the Port number indicate which hardware socket on the DigiSnap Pro Controller board the Module is connected through, and the lower 2 bits are for subdivision to distinguish primary modules, or within a Port Expander.

Port (binary)	Connection

0000	Primary Module/HUB (Socket 0)
00xx	Primary Module/HUB
0000	HUB
0001	DSPC Embedded Module
0010	Network Board Embedded Module
0011	reserved
01xx	AM Port #1 (Socket 1)
10xx	AM Port #2 (Socket 2)
11xx	AM Port #3 (Socket 3)
xx00	Expander Sub-Port 0
xx01	Expander Sub-Port 1
xx10	Expander Sub-Port 2
xx11	Expander Sub-Port 3

Defined Module List

Module Type ID (1..255)	Name (Abbreviation)
255 (FF)	Control Module (CTRLM)
254 (FE)	Camera Module (CAM)
253 (FD)	Bluetooth LE Module (BLEM)
252 (FC)	Network Control Module (NCM)
251 (FB)	Network Access Module (NAM)
250 (FA)	GPS Module (GPSM)
249 (F9)	Servo360 Module (S360M)
248 (F8)	Light Sensor Module (LSM)
247 (F7)	Wifi Module (WIFIM)
246 (F6)	Email Module (EMAILM)
-	
1	Debug Module (DBUGM)
0	Hub

The Primary Modules are connected through dedicated ports, and the Port number is set to 0.

Example Registry:

Module Addr (4 bits)	**Port #** (4 bits)	**Module Type ID** (1..255) (8 bits)
15	0000	Control Module = 255
14	0000	Camera Module = 254
13	0000	Bluetooth Low Energy Module = 253
12	0000	Network Control Module = 252
11	0000	Network Access Module = 251
10	0000	0 (available)
9	0100	Debug Module = 1
8	0000	0 (available)

7	1100	Servo360 Module = 249
6	0000	0 (available)
5	1101	Light Sensor Module = 248
4	1000	GPS Module = 250
3	1110	Servo360 Module
2	1111	Debug Module=1
1	0000	0 (available)
0	0000	Registry = 0

It is certainly possible for multiple implementations of a Module to register. There may be multiple motion axis Accessory Modules, and it is even possible for multiple User Modules to register, for example a BLEM to a smartphone, a Debug Module, and another via Network Board and HTTP.

Module Communications

A simple data packet protocol is used for exchange among the various Modules. All Modules will have a unique registered address, and packets may be routed from one Module to another. The connection to the Hub may be via several methods.

1) Direct.
A connection is made to the Module Hub via a hardware UART, connected to one of the three DSPC connectors.

2) Embedded.
The DSPC Modules are emulated in the DSPC micro controller, as if they were connected directly.

3) Indirect.
A device connected to the Hub may emulate more than one Module. Each of these Modules will have the same Port number, and the emulator would handle the routing. The Network Board Modules use this method.

4) Extended.
A Port Expander with a direct connection to the Hub may add up 4 UART connections for additional Module connections. Each endpoint would be assigned a sub-port identification, allowing the Port Expander to be relatively 'dumb'.

Note that not all Modules will be 'smart' enough to communicate with, and make use of information from, or control other Modules. The DigiSnap Pro Network board software will be periodically updated with the Module definitions, and hooks included in the scripting menus to make use of those Modules. Likewise, the smartphone app may be periodically updated to allow control and configuration of new Modules. The DSPC firmware may only be updated occasionally, and likely will not make direct use of any AMs. Most AMs also will not have knowledge of other AMs.

There is no communication prioritization among different Modules. A combined interrupt/polling process will be used, allowing communication when time allows. A single character is send via UART to signal that the DPSC or Module is ready to send a message, but no messages will be transferred until an acknowledgment signal is received.

The messaging process is deliberately designed to not use a Command/Response protocol. Messages are sent with no expectation of reply, all message retries and other handshaking are implemented at higher levels.

An SPI interface is implemented betwixt the DSPC and the Network board. The Network Board is always the SPI Master, and the DSPC is the SPI Slave. A dedicated signal line (RTS = ready to send) is used from the DSPC to the Network Board to indicate that the DSPC is ready to send a byte to the NB.

The UART baud rate will be fixed at 19.2KBaud. Some Accessory Modules may be connected via somewhat long cables, and 3V UART signals may be susceptible to noise and signal degradations at some distance/speed.

Security

Given the ease of connection via BLE, some mechanism for preventing malicious activity must be implemented in the connection mechanism. We do not want someone to develop an App that turns the DSPC off or reprograms the firmware of other Modules when driving by. Each DSPC will have a unique serial number assigned in production. This serial number will be required by the App during initial setup, which is then stored by the App along with the configuration settings for that DSP. Multiple serial numbers / configuration sets can be registered, corresponding with multiple DigiSnap Pro systems. When walking up to the system, the App

will only show availability of DSPs whose serial numbers match one that has been entered into the App. The user friendly Name entered into the DSP will also be displayed.

A password is implemented for additional security by the owner of the system. The password is never displayed to the user. Any User Module that is not hardwired to the DSP will be required to demonstrate that it knows the password before allowing any other exchange with the DSP. Hardwired connections (Debug Module, CIM) and remote Modules that have authenticated themselves are permitted to change the password, but none may see the password. They may re-enter the password and submit for authentication to verify that the DSPC has received and set it.

Some commands, such as firmware transfers, could be performed with enough skill to disable every other Module attached to the Hub. Firmware transfer commands are therefore only accepted if received by User Modules.

Beyond the above considerations, the Module communication is a trusted system, where malicious behavior is possible, and not prevented. For instance, any Module may register and send commands to other Modules, reconfiguring parameters at will. If abuses are eventually found, additional security measures could be incorporated into updates.

Write to any Module

Any Module may write a message to any Module, including itself.

1. Module Control - Message is sent to another addressed Module, intended for registration, configuration, requests, and control of that Module.
2. Response – Note that the protocol is not in a command/response format. Messages are written, and if the message requests local information, that information may be is messaged back, but if the message requests something from another Module, that information is later sent back as a normal Module message, with no immediate acknowledgment required.

Read from any Module

1) Public Read – Any Module may request a copy of the information stored in the public shared access memory of any Module, including it's own.
2) Read Registry – By reading from Module address 0, the complete registry will be returned.

Each Module has a set of status information, and operating parameters which may be viewed, and in some cases modified by other Modules. The data format for each Module is pre-defined by this document.

Any Module may request the public data from any other Module. This request is a standalone message, and no immediate response should be expected. The Module-being-requested may require some time to wake, gather it's internal data, and send a public data content message to the requesting Module. In some cases, resources may not be available so the content message never arrives. The requesting module should be designed to operate normally without a response.

Module Message Format

Source Module Addr (4 bits)	Destination Module Addr (4 bits)	Message Type (4 bits)	Reserved (4 bits always High)	Data Length in bytes (8 bits)	Data Up to 250 bytes	Checksum (16 bits)

The source Module is the registered address of the Module originating the message.

The destination Module is the registered address of the intended receiving Module.

The Message Type is a field that indicates the message context.

The Data Length is the number of bytes in the Data field of the message, which may be zero. The 8 bit field can indicate up to 255 bytes, while the maximum length is 250 bytes..

The Data is a context dependent set of bytes.

The checksum uses the Fletcher 16 algorithm, having dual summations of the entire message, and yielding a two byte field, pre-calculated to zero the received checksum. When the entire message including the checksum is received, and run through the checksum algorithm, the resulting sums should be zero. The Fletcher algorithm is simpler than a CRC, but more powerful than a simple additive process.

Message Types

Message Type Value (4 bits)	**Message Context**
0h	Reserved – should never occur.
1h	Register (Module asks to register, Hub provides Registry info to Module)
2h	-removed / not used-
3h	Module to Module
4h	Broadcast – announcement to all Modules
5h	Authenticate
6h	Log
7h	ReRegister Request (DSPC may have reset)
8h	- not defined-
9h	- not defined-
Ah	- not defined-
Bh	- not defined-
Ch	- not defined-
Dh	- not defined-
Eh	- not defined-
Fh	Firmware Transfer

Module Messaging Protocol

Message from Module to Hub : Common to all exchanges

Module >> ESC	Module sends <ESC> char to Hub, as a wake-up trigger.
Hub == IRQ	Hub receives character, which triggers interrupt, waking it.
Hub >> ACK	Hub send <ACK> character, letting Module know it's listening, and ready to receive message
Module >> Message	Module sends message content and checksum

The Hub may take some time to respond to the wake-up trigger, as it may be asleep and slow to awake, busy with other tasks, or waiting for room in the message queue to accept new message transfers. The Module should wait at least 1 second before retrying the wake-up trigger.

Message from Hub to Module: Common to all exchanges

Hub >> Address	Hub sends 8 bit Address to connection port, as a wake-up trigger to the Module. The Hub Expander will strip the SubPort value from the address before forwarding.
Module == IRQ	Module receives character, which triggers interrupt, waking it.
Module >> ACK	Module responds, signaling that it is awake
Hub >> Message	Hub sends message content and checksum

The Module may take some time to respond to the wake-up trigger, as it may be asleep and slow to awake, busy with other tasks, or waiting for room in its message queue to accept new message transfers. The Hub should wait at least 300mS second for an ACK, and then abandon the attempt to transfer that message, deleting from the queue. If thre attempt to transfer messages to a Accessory Module fails, the Accessory Module may be removed from the registry, and the Hub may cycle the power to that port, if there is no Port Expander. If there is a Port Expander present (known by the use of SubPort numbers), the Port Expander will cycle the power to that Accessory Module.

Module Registration

The Hub registers the Module, given the Type ID, providing it's registered Address

From Module to Hub (Request Registration)
Source: 0
Destination: 0 (Registry)
Msg Type: 1 (Register)
DataLen: 2
Data:

Byte 1: Module Type ID
Byte 2: Address- Module Sends 0 since it doesn't know yet it's address, Port Expander may modify the 2 bit SubPort #

From Hub to Module (Assign Registration)
Source: 0 (Registry)
Destination: Address (0 plus Sub-Port)
Msg Type: 1 (Register)
DataLen: 1
Data: 8 bit Assigned Address & Port

Port Expander Specific
The Port Expander remembers the SubPort number for each registered Module Address, for internal routing.

As there may be more than one module connected, the Port Expander will only allow registration of one module at a time.

If a register request is ignored for more than 1 second, the attempt is discarded, and that SubPort power is removed. Other pending Module messages may then be passed through (other register requests, for instance). Once all other pending messages have been handled, the SubPort power is re-established (minimum of 5 sec off time), to allow it to re-attempt to register.
-

Module to Module Write

From Source Module to Hub (Write the message to the Hub, for future transfer)
Source: Assigned Address
Destination: Address of Module to send message
Msg Type: 3 (Module to Module Write)
DataLen: 0..250

Data Bytes

The Source Module sends the message to the Hub, which receives it and stores in a message queue. The Hub will transfer this message unaltered to the Destination Module, when available.

If the Destination Address is the same as the Source, the Message content is placed in the Public Storage for that Module.

Writes to Destination Address 0 (Registry) are accepted, but ignored once the module has been registered.**From Hub to Destination (Transfer from Message queue)**
The message content is identical to the written message.

Module-Module Commands
Commands may be sent from one Module to another., using a specific message format. The command type is the first byte in the data field of the message. The number of bytes to follow depend on the command type and context. The message length field is used to determine the command length.

There are some predefined universal command types, but Modules may have additional types of commands specific to themselves. Not all Modules will accept all of the universal commands.

Command Type	Function	Description
01h	Set Parameter	Parameter Reference # (1 byte), followed by new value (1 or more bytes)
02h	Public Data Request	Source is requesting public data from destination
04h	Public Data Content	Source is sending public data to destination
08h	Execute Process	Process Reference # (1 byte), followed by modifier (1 or more bytes)
20h	Firmware	Data Transfer of arbitrary # bytes, in multiple messages, as firmware update.

Broadcast

From Source Module to Hub (Write the message to the Hub, for transfer to all other Modules)
Source: Assigned Address
Destination: 0 (Hub)

Msg Type: 4 (Broadcast)
DataLen: 1
Data: Broadcast type

The Hub re-transmits one Module's broadcast to Hub to all Modules in the Registry, changing the destination address appropriately for each Module receiving the broadcast message. This only requires one queued message, but the Hub must keep track of delivery to each Destinations .

Authenticate

From Source Module to Hub
Source: Assigned Address
Destination: 0 (Hub)
Msg Type: 5 (Authenticate)
DataLen: 16
Data: Password Attempt

The Source Module sends what it thinks to be the correct 16 byte password to the Hub.
From Hub to Destination (Transfer from message queue)
Source: 0 (Hub)
Destination: Source Address
Msg Type: 5 (Authenticate)
DataLen: 1
Data: 0F = Authentic, F0 = Wrong Password

The Hub will return the Message to the Source Module, with a Pass/Fail field.

Log

From Source Module to Hub (Write the message to the Hub, for transfer to selected Modules)
Source: Assigned Address
Destination: 0 (Hub)
Message Type: 6 (Log)
Data Len: 2..33
Data: Severity Level byte, ASCII text characters

The Log message is used when an internal event has occurred in a module which may be of note, or concern. The Log message may be archived by the Network Control Module (NCM), and the Debug Module. The message data includes a 1 byte 'Severity Level' indication, followed by 1 to 32ASCII characters. The Hub will pass these messages to the NCM, DBUGM, and BLEM, if they are connected.

per https://en.wikipedia.org/wiki/Syslog#Severity_level

Severity level

Value	Severity	Keyword	Description	Examples
0	Emergency	emerg	System is unusable	This level should not be used by applications.
1	Alert	alert	Should be corrected immediately	Loss of the primary ISP connection.
2	Critical	crit	Critical conditions	A failure in the system's primary application.
3	Error	err	Error conditions	An application has exceeded its file storage limit and attempts to write are failing.
4	Warning	warning	May indicate that an error will occur if action is not taken.	A non-root file system has only 2GB remaining.
5	Notice	notice	Events that are unusual, but not error conditions.	
6	Informational	info	Normal operational messages that require no action.	An application has started, paused or ended successfully.
7	Debug	debug	Information useful to developers for debugging the application.	

The Network Board will be powered off except when in use to transfer images or send notification emails. In order to then save Log messages while the NB is off, the DSPC must use internal storage, then transfer the Log to the NB when available, as individual messages.

Log Messages can be stored in RAM. There are two header bytes for each Log message, containing the Source module address, and the severity. These two bytes are both sparse, so they will be combined for best efficiency. The Source address will be in the MS nibble, and the severity in the LS nibble. The Log Message will still contain from 1 to 32 ASCII characters. The end of the Log message will be a flag byte of FF.

This means that each Log message may be stored with between 3 and 36 bytes of RAM. Assuming that an average Log message length is 16 bytes, a 256 byte block of RAM could handle about 16 Log messages.

Log messages should not be generated frequently, but I'd guess that they will cluster a bit. To prevent overflow of the Log block, the NB should be awaken if more than 75% of the allocated RAM space is used. If the RAM is filled, no more Log messages will be stored in RAM until all of the messages in the block are transmitted to the NB, as individual messages.

Firmware Transfer

From Source Module to Hub (Write the message to the Hub, for future transfer to all other Modules)
Source: Assigned Address
Destination: Address of Module to receive new Firmware
Msg Type: F (Firmware Transfer)
DataLen: 2..36

Data format for updating the controller firmware:
2 bytes indicating message type:
“SS” - just 2 bytes requesting scratchpad flash erase
“S1” - followed by two bytes of address and up to 32 bytes of code data (from a Motorola S1 record)
“Si” - followed by total byte count (2 bytes) and total S1 record count (2 bytes).

The first 2 data bytes in the first transfer message sent to the controller must be 'SS' which the controller uses to erase scratchpad flash memory. Note that this command is completely destructive, as this causes the microcontroller to erase the entire flash space. After a 2 second delay the S1 messages are sent in which the first two data bytes are 'S1', followed by 2 bytes of address and up to 32 bytes of data. The s-record checksum is not sent since the messaging protocol has its own checksum. The microcontroller will respond at the end of the message with ACK (0x06) if the message was received with valid checksum, or NAK (0x15) to indicate some error. If nothing is received within 100uS, any character should be sent until a NAK is received, to resynchronize. If a NAK is received, the record should be retransmitted.

After all S1 records have been sent, a final install message is sent in which the first two data bytes are 'Si', followed by 2 bytes indicating the total number of data bytes and 2 bytes indicating the total number of S1 records. Upon receipt of this message the controller completes the installation of the firmware, and resets the processor. After the “Si” message is received by the controller, it shall be offline (while installing the new program into flash) not more than 5 seconds.

Hub (Registry)

Function Type ID : 00h = 0
Port Number : 0

The Hub is the central connection point for all Modules in the DigiSnap Pro system. It is implemented as firmware on the DSPC microcontroller.

Public Access Data

Item	Description	Bytes	Format

Supported Incoming Message Types

The Hub will accept several types of input commands.

Command Type #	Command Type	Content
01h	Request Registration	Module that has just powered up, is asking to be registered
04h	Broadcast	Broadcast messages are forwarded to all other connected modules.
05h	Authenticate	Module is asking if a 16 byte password value matches this device.
06h	Log	Module has experienced an event it would like to report. Log reports are forwarded to all connected NCM, BLEM, and DBUGM modules.

Supported Module-Module Commands

Command Type #	Command Type	Content
02h	Public Data Request	Source is requesting registry from Hub
04h	Public Data Content	Hub is sending registry to destination

Outgoing Announcement / Broadcast Messages

Announcement Reference #	Announcement Name	Description
01h	Registry Update	Registry has been updated (Module added or removed)

Control Module (CTRLM)

Function Type ID : FFh = 255

The DSPC has control over the power to all other Modules, cameras, external equipment and LED indicators. The Control Module provides control over some of these functions via parameter change messages from other Modules.

The BLEM reads the Format. Firmware Rev, Serial Number, and Name. These fields must not be changed in format, or the BLEM code would need to be updated.

Public Access Data

Item	Description	Bytes	Format	Position
Format	Repository Format Version (This Table=5)	1	Numeric	0
Firmware	Firmware Version (DigiSnap Pro Controller)	1	Numeric	1
Serial Number	Factory programmed identification	4	SerNum	2
Name	User friendly name for this particular unit	24	Byte	6
SSLR	Seconds Since Last Reset	4	Secs32	30
State	DSPC State	1	DSPC State	34
History	DSPC History	1	DSPC Hist	35
Temp Amb	Last sampled Ambient Temp, in the air between the boards.	1	Temperature	36
Time Amb	Time since Ambient Temp Sample	2	Secs16	37
Temp Board	Current temperature on the circuit board, near the charger.	1	Temperature	39
Humidity	Current Relative humidity, between the circuit boards.	1	Humidity	40
Batt Stat	Battery Status	1	DSPC BaS	41
BattV1	Battery 1 Voltage	1	Volt25.6	42
BattV2	Battery 2 Voltage	1	Volt25.6	43
NB STS	How long until Network Board Sleeps (Secs Till Sleep)	2	Secs16	44
IO State	State of IO signals	1	DSPC IO	46
IO In Mode	Input Trigger Function	1	DSPC In Mode	47
IO Out Mode	Signal Out output mode	1	DSPC Out Mode	48
Charger State	Charger status	1	DSPC ChS	49
Ch Curr Out	Charge output current	1	Curr2.56	50
Ch Curr In	Charger input current	1	Curr2.56	51
Ch Volt In	Charger input voltage	1	Volt38.4	52
SW Batt Out	Switched battery output mode	1	DSPC Out Mode	53
Sw 5V Out	Switched 5v output mode	1	DSPC Out Mode	54
RemoteMode	Networking Remote Command Check-in timing	1	Remote Mode	55
Calibration	Various calibration constants for factory use	4	Calib	56
L_ISM	Local image storage mode	1	ImgStoreMode	60
R_ISM	Remote image storage mode	1	ImgStoreMode	61
R_Decimate	Image transfer Decimation, to save bandwidth to Remote storage	1	Dec_Index	62
R_Size	Image size to sent to Remote storage	1	Size_Index	63

Email Mode	When to send email message to operator	1	EmailMode	64
Battery Type	Select a battery chemistry, for charger	1	BatteryType	65
Errors	Collection of Error Flags (documented elsewhere)	4	Binary	66
	*** Fixed Length Data Total ***	70		

Secs32: Number of seconds (0.. 2**31)
Secs16: Number of seconds (0..65,535) >> 18 hrs. Does not overflow.
Secs 8: Number of seconds (0..255)
Volt12.8: $FF=12.7V. Resolution=50mV/LSB
Volt25.6: $FF=25.5V. Resolution=100mV/LSB
Volt38.4: $FF=38.25V. Resolution=150mV/LSB
Curr2.56: $FF=2.55A. Resolution=10mA/LSB
TimeExp: Index value (0..255) corresponding to a 1/12 stop increment time value, from 1mS..2494 sec
Temperature: Signed Byte, +/-127 deg C. Valid range = -40/+125
Humidity: $FF=100%RH, $00=0%
String: ASCII text. If the length of the string is less than the max, the string shall have a null termination.

Dec_Index {0=1 (no decimation), 1=5, 2=10, 3=25, 4=50, 5=100, 6=250, 7=500, 8=Noon}
Size_Index {0=Full size, 1=Large Thumbnail, 2=Medium Thumbnail, 3=Small Thumbnail}

ImgStoreMode: Byte
0: Never transfer images to storage location
1: Transfer images whenever convenient (when network hardware is already on)
2: Transfer images after each new picture
3: Transfer images after every 30 pictures are taken
4: Transfer images at Midnight each night
5: Transfer images at Midnight each Sunday night

Battery Type: Byte
0: Primary battery / power adapter, disable charging
1: 4 Cell Lithium Ion (14.8V nominal)
2: 3 Cell Lithium Ion (11.1V nominal)
3: 4 Cell Lithium Iron Phosphate (12.8V nominal)
4: 12V Lead Acid

Email_Mode:

7		
6		
5	Trigger	Send email if input Trigger is activated
4	FriMorning	Send email at Friday Monring (6AM)
3	Morning	Send email each morning (6AM)
2	Picture	Send email for each picture taken
1	ProbLimit	Send Email when a problem is detected, but limit to 6AM, Noon, 5PM
0	Problems	Send Email when a problem is detected. Sent hourly as long as problem exists.

DSPC In Mode: Byte
0: Do Nothing [Broadcast and Email may still be stimulated]
1: Take Pic
2: Take Burst of Pics (Delay, NBurst, IntervalBurst)
3: Enable all Time-Lapse Sequences, if disabled

DSPC Out Mode: Byte
0: Output is always off (inactive)
1: Active before picture taken (Out Before TimeExp time)
2: Active after picture taken (Out After TimeExp time)
3: Active both before and after picture taken
4: Active always

5. Heater control.
6. Active when Network Board is powered on. May supply local storage or networking devices.

SerNum: Factory Programmed 4 Byte Number
The serial number is a composite of board revision level, production date, and production index number.
Upper 4 Bits: Controller Board Rev, offset from Rev E. i.e. Rev F would have a value of 1.
Next 4 Bits: Network Board Rev, offset from Rev C. i.e. Rev F would have a value of 3.
Next 4 Bits: Year, offset from 2016. i.e. 2016 would have a value of 0.
Next 4 Bits: Month, 1..12
Next 5 Bits: Day, 1..31
Next 11 Bits: Index. incrementally assigned number, 0..2047

Remote Mode: Byte
0: Off [Network board is never waken for the purpose of checking for remote commands]
1: Check for queued commands each day at Noon.
2: Check for queued commands each Friday at Noon.
3: Wait for remote interaction each day at Noon (Available for 15 minutes).
4: Wait for remote interaction each Friday at Noon (Available for 15 minutes).

DSPC State:

7	CamPwrOn	Camera Power is currently enabled – voltage is applied to connector
6	NMPwrOn	Network Board Power is On
5	ChargeOn	Charger is currently enabled
4	CamPConn	Current State of DSPC camera power connector (Plugged in=1)
3	ChPConn	State of DSPC Charger jack. (1=Plugged In) – If 0, then charge power may be POE
2	NetDet	Network Board Detected (0 = detected, 1= not detected)
1	5V_Pwr	Internal 5V Supply Power is On (regardless of 5V outputs)
0	RTC_OK	1 if RTC OK, 0 if Clock not set

DSPC Hist:

7	R Image Transfer	Image transfer to Remote image storage is requested.
6	Email	Status Email is requested to be sent.
5	Remote Check	Check for remote commands are requested.
4	Module	A Module has requested Network Board be turned on, for unspecified reason. This could be due to Bluetooth App connection. Keep awake until App disconnects.
3	L Image Transfer	Image transfer to local storage is requested.
2	LowBatt	A low battery event has occurred, which may have prevented networking in the past.
1		
0		

DSPC BaS:

7	B1 Installed	There appears to be a battery installed, at Batt1.
6	B1 Clipped	The voltage measurement for Batt1 has clipped (voltage may be higher)
5	B2 Installed	There appears to be a battery installed, at Batt2.
4	B2 Clipped	The voltage measurement for Batt2 has clipped (voltage may be higher)
3	AllCharged	All batteries are near fully charged.

2	BattLow	Both Battery voltages are running low. Less than about 25% full charge.
1	BattCrit	Both Battery voltages are running critically low. Less than about 5% full charge.
0	Loaded	Battery connected to load: 0=Batt1, 1=Batt2

DSPC IO:

7	Sig_In	State of Input Signal (0=Inactive/High/Open, 1=Active/Low/Shorted)
6	Sig_Out	State of Output Signal (0=Inactive/High/Open, 1=Active/Low/Shorted). Note that this is the DSPC enable, the Network Board may also activate, but this is not monitored
5	5V_IO	State of Switched 5V Power at IO connector. (0=Off, 1=Enabled)
4	Batt_IO	State of Switched Battery Power at IO connector. (0=Off, 1=Enabled)
3	Pwr_AM1	State of Accessory Module Port 1 Power (1=On)
2	Pwr_AM2	State of Accessory Module Port 2 Power (1=On)
1	Pwr_AM3	State of Accessory Module Port 3 Power (1=On)
0		

DSPC ChS:

7	Charging	1 =charging
6	Fault	Inverted pin 5 from the charger chip: 1=Fault, 0=OK
5	Charging2	Battery 2 is connected to Charger
4	Charging1	Battery 1 is connected to Charger
3	Dis_OverCh	Charger has been disabled, as all connected batteries appear to be overcharged
2		
1	Dis-Temp	Charger has been disabled as the temperature rise is excessive, the ambient temperature is too high or low, or the board temperature is too high.
0		

Supported Module-Module Commands

Command Type #	Command Type	Content
01h	Set Parameter	Parameter Reference # (1 byte), followed by new value (1 or more bytes)
02h	Public Data Request	Source is requesting public data from destination
04h	Public Data Content	Source is sending public data to destination
08h	Execute Process	Process Reference # (1 byte), followed by modifier (1 or more bytes)
20h	Firmware	Data Transfer of arbitrary # bytes, in multiple messages, as firmware update.

Set Parameter

Note that locating the multi-byte parameters at the end and using an even number of bytes makes the design cleaner.

Parameter Reference #	Parameter Name	Description	Value Bytes
01h	NB Minutes Till Sleep	Network Board keep-alive time Value in minutes	1
02h	NB Logic Power	Network Board Logic Power switch. 0>>Turn off, any other value, wake the board.	1

		Note: Waking the board involve the 3.6V power, PSwitch, and RST signals.	
03h	NB 5V Power	Network Board 5V Power source 0>>Not Required, any other value, turn on	1
04h	Signal Out Mode	Signal Output mode (DSPC Out Mode) on IO Port	1
05h	5V Out Mode	5V Power Output mode (DSPC Out Mode) on IO Port	1
06h	Batt Out Mode	Battery Power Output mode (DSPC Out Mode) on IO Port	1
07h	AM1 Power	Battery Output on Accessory Module Port #1 0>>Disable, any other value, Enable DSPC will normally manage this signal.	1
08h	AM2 Power	Battery Output on Accessory Module Port #2 0>>Disable, any other value, Enable DSPC will normally manage this signal.	1
09h	AM3 Power	Battery Output on Accessory Module Port #3 0>>Disable, any other value, Enable DSPC will normally manage this signal.	1
0Ah	DSPC In Mode	Action to take when input signal goes active	1
0Dh	Serial Number	Serial number (only accepted by debug module)	4
0Eh	Name	User friendly name for the unit	24 max
0Fh	Password	Password (user enters 3-16 characters)The Password is never displayed, or transmitted, only set.	16 max
10h	Remote Mode	Networking Remote Command Check-in timing	1
11h	History	Write 0 value to clear the History, and Errors	1
12h	Calibration	Calibration constant, accepted from Debug Module only	4
13h	L_ISM	Local image storage mode	1
14h	R_ISM	Remote image storage mode	1
15h	R_Decimate	Image transfer Decimation, to save bandwidth to Remote storage	1
16h	R_Size	Image size to sent to Remote storage	1
17h	Email Mode	When to send email message to operator	1
18h	Battery Type	Configure the battery pack type	1

Execute Process

Process Reference #	Process Name	Description
01h	Enable Charger	Permit the Battery Charger to operate
02h	Disable Charger	Prevent the Battery Charger from running
03h	Cycle power on Connections	Remove power from the network board and the 3 Accessory Module connectors AM1,2,3 for at least 15 sec and then restore power.
04h	Reset Controller	Reset the controller board
05h	Test A	Perform production test sequence A . Report “PASS” or “FAIL” in Log Message
06h	Test B	Perform production test sequence B . Report “PASS” or “FAIL” in Log Message

07h	Identify	Flash LEDs in a sequence to identify the unit. Useful with multiple devices connected to App.
08h	Email Now	Send a status Email to the operator
09h	Factory Default	Reset Parameters to factory defaults

Outgoing Announcement / Broadcast Messages

Announcement Reference #	Announcement Name	Description
1	Trigger	Input trigger activation has occurred
2	BatteryLow	If an event was suppressed by the DSPC due to low battery power.

Camera Module (CAM)

Function Type ID : FEh = 254

The Camera Module (CAM) is an additional Module implemented by the DSPC, dealing specifically with camera control and monitoring, time keeping, and time-lapse functions.

Public Access Data

Item	Description	Bytes	Format	Position
Format	Repository Format Version (This Table=4)	1	Numeric	0
Time and Date	Current RTC time and date	6	RTC format	1
Camera Status	Camera status	1	DSPC CaS	7
C Curr Sleep	Last Pic camera current, when inactive	1	Curr2.56	8
C Pwr Mode	Camera Power Mode	1	C Pwr Mode	9
Before Time	Time for actions before picture	1	Secs8	10
After Time	Time for actions after picture	1	Secs8	11
C Curr Ready	Last Pic camera ready current (Half-Press)	1	Curr2.56	12
C Curr Peak	Last Pic camera peak current (Full Press)	1	Curr2.56	13
Cam Volt Set	Camera voltage set point	1	Volt12.8	14
Camera Volt	Camera Power Supply Voltage, measured	1	Volt12.8	15
Camera Curr	Camera Power Supply Current Draw, now	1	Curr2.56	16
Cam HP Time	Camera Half-Press duration setting	1	Time16	17
Cam FP Time	Camera Full-Press duration setting	1	TimeExp	18
TL1 Start	Time of day to start time lapse sequence	2	Hr, Min	19
TL1 End	Time of day to end time lapse sequence	2	Hr, Min	21
TL1 Interval	Time between pics	3	Hr, Min, Sec	23
TL1 Enable	TL1 sequence Overall and Days of Week	1	Byte	26
TL2 Start	Time of day to start time lapse sequence	2	Hr, Min	27
TL2 End	Time of day to end time lapse sequence	2	Hr, Min	29
TL2 Interval	Time between pics	3	Hr, Min, Sec	31
TL2 Enable	TL2 sequence Overall and Days of Week	1	Byte	34
TL3 Start	Time of day to start time lapse sequence	2	Hr, Min	35
TL3 End	Time of day to end time lapse sequence	2	Hr, Min	37
TL3 Interval	Time between pics	3	Hr, Min, Sec	39
TL3 Enable	TL3 sequence Overall and Days of Week	1	Byte	42
TL4 Start	Time of day to start time lapse sequence	2	Hr, Min	43
TL4 End	Time of day to end time lapse sequence	2	Hr, Min	45
TL4 Interval	Time between pics	3	Hr, Min, Sec	47
TL4 Enable	TL4 sequence Overall and Days of Week	1	Byte	50
N Burst	Input Trigger Burst: Number of Pics	1	0 = infinite	51
Interval Burst	Input Trigger Burst: Interval (Mins, Secs)	2	0..59, 0..59	52
Interval Delay	Input Trigger Burst Delay	2	Secs16	54
PeakC Ref	Camera peak current reference	1	Curr2.56	56

Last Pic Date	Date and time of last successful picture	6	RTC format	57
Last Pic Attempt	Date and time of last attempted picture	6	RTC format	63
Inhibited by	Module # which sent the inhibit cmd if the last attempted picture was inhibited	1	0 if not inhibited	69
NextSchPic	Time of Next Scheduled Picture	5	Source, DOW, Hr, Min, Sec	70
PicsAttempted	# of pictures attempted this day	2	numeric	75
PicsFailed	# of pictures believed to have failed this day	2	numeric	77
	*** Fixed Length Data Total ***	79		

Time16: Number of 1/16th second increments (0..255) >> 62.5mS.. 16Sec (implicitly add 1)

C Pwr Mode: Byte
0: Never provide power to the camera (camera uses another power source)
1: Power Always On
2: Power applied before and after Pic
3: Off momentarily each day (Reset camera)

Enable: Byte - Enable overall, and select days of week
0: Monday
1: Tuesday
2: Wednesday
3: Thursday
4: Friday
5: Saturday
6: Sunday
7: Time Lapse Sequence Enable

RTC format (6 bytes):

RTC Year	Real Time Clock Year (000..255) 2xxx	1	Numeric
RTC Month	Real Time Clock Month (1..12)	1	Numeric
RTC Day	Real Time Clock Day of Month(1..31)	1	Numeric
RTC DOW	Real Time Clock Day of Week 1..7 (Mon=1)	1	Numeric
RTC Hr	Real Time Clock Hour (0..23)	1	Numeric
RTC Min	Real Time Clock Minute (0..59)	1	Numeric

NextSchPic format (5 bytes):. Values set to 255 if no picture is scheduled.

Source	0:Burst, 1..4=TimeLapse Sequence	1	Numeric
DOW	Day of Week	1	Numeric
Hour	Hour of Day(0..24)	1	Numeric
Minute	Minute (0..59)	1	Numeric
Sec	Second (0..59)	1	Numeric

CaS:

7	CamPwrDis	Camera Power has been disabled locally... voltage or current error
6	HalfPress	Current State of Half-Press line to camera (Open/High=1)
5	FullPress	Current State of Full-Press line to camera (Open/High=1)

4	CV Clipped	The voltage measurement for Camera Volt has clipped (voltage may be higher)
3	CI Clipped	The camera current measurement has clipped (may be higher)
2	Failed	Last picture may have failed.(camera currents compared with calibration values unacceptable). If camera is self or externally powered, no check is possible, and value is cleared.
1		
0		

Supported Incoming Module-Module Commands

Command Type #	Command Type	Content
01h	Set Parameter	Parameter Reference # (1 byte), followed by new value (1 or more bytes)
02h	Public Data Request	Source is requesting public data from destination
04h	Public Data Content	Source is sending public data to destination
08h	Execute Process	Process Reference # (1 byte), followed by modifier (1 or more bytes)

Set Parameter

Parameter Reference #	Parameter Name	Description	Value Bytes
01h	Cam Volt	Camera Power Supply Voltage Value in Volt12.8 format. 0>>Turn Off	1
02h	Cam HP Time	Camera Half-Press Duration Value in ExpTime	1
03h	Cam FP Time	Camera Full-Press Duration Value in ExpTime	1
04h	RTC	Real-Time clock on the DSPC Year, Month, Day, DOW, Hr, Min	6
05h	TL1 Start	Time Lapse 1, Start Time. Hr, Min	2
06h	TL1 End	Time Lapse 1, End Time. Hr, Min	2
07h	TL1 Interval	Time Lapse 1 Interval, Hr, Min, Sec	3
08h	TL1 Enable	Time Lapse 1, Days of Week to enable	1
09h	TL2 Start	Time Lapse 2, Start Time. Hr, Min	2
0Ah	TL2 End	Time Lapse 2, End Time. Hr, Min	2
0Bh	TL2 Interval	Time Lapse 2 Interval, Hr, Min, Sec	3
0Ch	TL2 Enable	Time Lapse 2, Days of Week to enable	1
0Dh	TL3 Start	Time Lapse 3, Start Time. Hr, Min	2
0Eh	TL3 End	Time Lapse 3, End Time. Hr, Min	2
0Fh	TL3 Interval	Time Lapse 3 Interval, Hr, Min, Sec	3
10h	TL3 Enable	Time Lapse 3, Days of Week to enable	1
11h	TL4 Start	Time Lapse 4, Start Time. Hr, Min	2
12h	TL4 End	Time Lapse 4, End Time. Hr, Min	2
13h	TL4 Interval	Time Lapse 4 Interval, Hr, Min, Sec	3
14h	TL4 Enable	Time Lapse 4, Days of Week to enable	1
15h	Camera Power	C Pwr Mode (0..3)	1

	Mode		
16h	N Burst	Number of pics to take when IO In Mode = 2	1
17h	Burst Interval	Time between pics when IO In Mode = 2. Min, Sec	2
18h	Burst Delay	Delay (in seconds) from input signal active to action	2
19h	IO Out Before	Seconds to output signal before picture	1
1Ah	IO Out After	Seconds to output signal after picture	1

Execute Process

Process Reference #	Process Name	Description
01h	Take Pic	Take picture, using pre-defined parameters for timing
02h	Inhibit	If a Pending Picture broadcast message has just been sent, skip this next picture, otherwise ignore.
04h	Start Burst	Initiate the Burst picture process
05h	Factory Default	Reset all parameters to factory defaults

Outgoing Announcement / Broadcast Messages

Announcement Reference #	Announcement Name	Description
01h	PicPend	The Camera is about to take a pic. Modules have an opportunity to inhibit
02h	PicDone	The Camera has taken a pic. New camera current data is available.
03h	PicAborted	The last picture attempted was aborted

Bluetooth Smart/Low Energy Module (BLEM)

Function Type ID : FDh = 253
Port Number : 00h = 0

The Bluetooth Low Energy Module (BLEM) is an option to the DSPC, and as such is removable. The BLEM operates substantially independently of the DSPC, with all interaction via a UART interface, as do all other Modules. The BLEM registers with the Hub upon powerup.

The BLEM contains a microcontroller which will handle the BLE processes, and reformat the data to align with the DigiSnap Pro Module protocol, allowing the smartphone operator to manipulate the settings of the DSPC and other Modules.

For security, each DigiSnap Pro Controller board will manage a 16 byte unique password. This password is printed on a label that ships with each unit. Passwords assigned at the factory will be limited to 8 ASCII printable characters. The operator will enter the password into the smartphone App ahead of time for use when connecting to the system. When connected, the only message the BLEM will allow to pass through the from the App to the Hub is to check that the password matches the unit. If there is a match, the session is continued, and if not, the BLEM will indicate failure to the App, where a message is displayed to the operator that the password entered is not correct. The operator may choose to re-enter and retry up to 30 times. At that point, the BLEM will disconnect the session, and set a 'Locked' flag in the advertisements. The BLEM will not allow any reconnect attempts for 15 minutes after receiving 30 incorrect attempts in a row. Once authenticated, the controller will accept a Set Password message, allowing the operator to change a password. The operator should have the opportunity to enter a new password from 3 to 16 characters in length. Unused characters shall be padded with 0xFF so that the controller always receives a 16 byte password.

When the BLEM is connected to a user, it will continue to advertise with a 'Connected' flag set, to indicate to a second operator that the system is working, but someone else is currently connected.

Public Access Data

Item	Description	Bytes	Format	Position
Format	Repository Format Version (This Table = 2)	1	Uint8	0
Version	BLEM Firmware Version	1	Version	1
Refresh Info Interval	Time between reads of system info (in 10 second increments); power-up default = 90 (15 min), 0 means never auto-refresh.	1	Period (Uint8)	2
BLEM State	State of BLEM Module – bit fields indicating state	1	BLEM State	3
BT MAC	MAC address of BT module; power-up default is the factory-set value. If changed by a Set Parameter msg, that value will be used, unless set to 0, in which case the factory-set value is restored.	6	MAC (Uint8[6])	4
Temp Amb	Last sampled Ambient Temp	1	Temp C (Int8)	10
Advertise Interval	Advertisement beacon period (in 0.1 seconds); power-up default = 10 (1 second)	1	Period (Uint8)	11
BT Session Idle Timeout	Idle connection duration timeout, at which point smartphone is disconnected (in 10 second increments); power-up default = 30 (5 min), 0 means never timeout.	1	Duration (Uint8)	12
Tx Power	Transmit power (4 dBm down to -30 dBm, in 4 dBm steps); power-up default is 4 dBm	1	Power (Int8)	13
MinCI_mS	Minimum Connection Interval (8...255)	1	mS	14
MaxCI_Muli	Maximum Connection Interval Multipler MaxCI = MinCI * Mult / 10	1	numeric	15
SlaveLat	Slave Latency (0...255)	1	numeric	16
	*** Fixed Length Data Total ***	17		

BLE State:

Bit	Name	Description
0	*Standby*	BLE Module is disconnected and has been instructed to stop advertising. If this bit is set, the only other valid bit that can be set is *Locked*.

1	*Advertising*	BLE Module is advertising its availability for connection. If if this bit is set, the only other valid bits that can be set are either *Connected* or *Locked*.
2	*Connected*	BLE Module is connected to a smartphone operator. If this bit is set, the only other valid bits that can be set are either *Advertising* or *Authenticated.*
3	*Locked*	BLE Module is locked due to excessive incorrect authentication requests.
4	*Authenticated*	BLE Module is connected and has received a valid authenticate request.

Supported Incoming Module-Module Commands

Command Type #	Command Type	Content
01h	Set Parameter	Parameter Reference # (1 byte), followed by new value (1 or more bytes)
02h	Public Data Request	Source is requesting public data from destination
04h	Public Data Content	Source is sending public data to destination
08h	Execute Process	Process Reference # (1 byte), followed by modifier (1 or more bytes)

Set Parameter

Parameter Reference #	Parameter Name	Description	Value Bytes
01h	*Refresh Info Interval*	Refresh interval (in 10 second increments)	1
02h	*Advert Interval*	Advertisement beacon period (in 0.1 second increments)	1
03h	*Idle Session Timeout*	Connection duration timeout (in 10 seconds increments)	1
04h	*MAC Override*	New MAC address of BT interface. If set to 00:00:00:00:00:00, then the default MAC is used.	6
06h	*Tx Power*	Transmit power in dBm	1
07h	*MinCI_mS*	Minimum Connection Interval (default to 20mS)	1
08h	*MaxCI_Mult*	Maximum Connection Interval multiplier (default to 25)	1
09h	*SlaveLat*	Slave Latency (default to 2)	1

Execute Process

Process Reference #	Process Name	Description
01h	*Disconnect*	Disconnect from the smartphone operator
02h	*Stop Advertising*	Force module to stop advertising beacon
03h	*Start Advertising*	Force module to start advertising beacon
04h	*Read Temperature*	Sample BLEM temperature sensor and update public memory with new reading
05h	*Factory Default*	Reset all parameters to factory default

Outgoing Announcement / Broadcast Messages

Broadcast Reference #	Broadcast Name	Description
01h	*BLEConn*	The BLE has connected to a smartphone operator

02h	*BLEDisc*	The BLE has disconnected from a smartphone operator

Messages Over Bluetooth

The BLEM supports an abbreviated version of the DigiSnap protocol to simplify message transfers between the BLEM and the smartphone app.

In-bound Messages

In-bound messages (from the smartphone app to the BLEM) are intended to be forwarded from the BLEM to the Hub for further action/routing. In-bound messages can be of type: Public Store Read, Module-Module, or Authenticate. In-bound messages are at least 3 bytes in length, and can be up to 94 bytes, with the following format:

Destination Module Addr (4 lsb of 8 bits)	Message Type (4 lsb of 8 bits)	Data Length in bytes (7 lsb of 8 bits)	Data (up to 91 bytes)

Out-bound Messages

Out-bound messages (from the BLEM to the smartphone app) originate from the Hub and are intended to be sent directly to the smartphone app. Out-bound messages can be of type: Public Store Read, Broadcast, or Authenticate. Out-bound messages are at least 3 bytes in length, and can be up to 94 bytes, with the following format:

Source Module Addr (4 lsb of 8 bits)	Message Type (4 lsb of 8 bits)	Data Length in bytes (7 lsb of 8 bits)	Data (up to 91 bytes)

Advertisement and Scan Data

When the BLEM is advertising its presence, it provides some key information that enables the smartphone app to differentiate its broadcast from that of other BLE devices, and therefore enables the app to present only DigiSnap Pro devices to the user. The advertisement data is always broadcast, at a default interval of 1 second, except while in a connection with a smartphone app (this is due to a limitation in the Laird BL600 firmware rev 1.5.70). Scan data is provided only after the smartphone app issues a scan request.

Advertisement Data

Advertisement data consists of the following fields and data values:

Is Connectable = 1 if not locked, 0 if locked
Local Name = “DigiSnap Pro”
Services UUIDs = 0x180A (“Device Information”)
Manufacturer Specific Data = 0x######## (4-byte serial number),
0x## (status byte, bit 3 is set if locked or bit 2 is set if connected)

Scan Data

Advertisement data consists of the following fields and data values:

Manufacturer Specific Data = “<user-friendly name>” (up to 24 character name)
Transmit Power = ## (transmit power in dBm, from -30 to 4)

Network Control Module (NCM)

Function Type ID : FCh = 252
Port Number : 00h = 0

The Network Hardware Module (NCM) is one of the Modules presented by the Network board. The NCM manages the resources of the Network board.

Public Access Data

Item	Description	Bytes	Format	Position
Format	Repository Format Version (This Version=5)	1	Numeric	0
Version	Software Version	1	Numeric	1
SSLR	Seconds Since Last Reset	3	Secs24	2
SSLCR	Seconds Since Last time Nb Reset the DSPC	3	Secs24	5
NB Year	Year (000..255) 2xxx	1	Numeric	8
NB Month	Month (1..12)	1	Numeric	9
NB Day	Day of Month(1..31)	1	Numeric	10
NB DOW	Day of Week 1..7 (Mon=1)	1	Numeric	11
NB Hr	Hour (0..23)	1	Numeric	12
NB Min	Minute (0..59)	1	Numeric	13
Conn State	State of Network board Connections	1	Conn State	14
Image SS	Image Storage State	1	Byte	15
Cam Total	Camera storage total capacity	2	Storage Size	16
Cam Free	Camera storage free space	2	Storage Size	18
Local Total	Local (USB) storage total capacity	2	Storage Size	20
Local Free	Local (USB) storage free space	2	Storage Size	22
USB0 State	USB Host Port 0 State	1	USB State	24
USB1 State	USB Host Port 1 State	1	USB State	25
NB Mode	Network Configuration	1	Network Mode	26
NB State	Network Status	1	Network State	27
Net IP	Network board IP Fixed Address and prefix length	5	IP Address	28
NDBootTime	Network device boot time	1	NDBootTime	33
Time Options	Time keeping options	1	TimeOptions	34
Timezone	Timezone abbreviation	3	3 char string	35
UTC Offset	Time offset from UTC. 15 min resolution 2s Complement. +24Hr=96 (0x60), 0min=0, +15min=1, -15min=127 (0x7F), -96=160 (0xA0)	1	Numeric	38
Address: eth0	Current IP address of Ethernet port	4	IP4 address	39
Address: wlan0	Current IP address of WiFi port	4	IP4 address	43
Address: cellular	Current IP address of cellular modem (wwan0, ppp0, eth1 – depending on hardware)	4	IP4 address	47
	*** Fixed Length Data Total ***	51		

Storage Size (2 bytes / 16 bits, capable of expressing sizes 0 KB to 16384 TB):

High bits (15..14): unit index
0: Kilobytes
1: Megabytes
2: Gigabytes
3: Terabytes
Low bits (13..0):
unsigned integer value (0..16384)

IP4 Address: xxx.xxx.xxx.xxx

NDBootTime:
0: Network device boots very quickly/instantly when power is applied
1: Network device will require less than 1 minute to boot & connect
2: Network device will require less than 2 minutes to boot & connect
3: Network device may take as long as 5 minutes to boot & connect

Conn State:

7	Ethernet	Ethernet is Active [Read only]
6	Internet	Internet Connection is Active [Read only]
5	Cellular	Cellular Connection is Active [Read only]
4	WiFi	Wifi Connection is Active [Read only]
3	Bluetooth	Bluetooth Connection is Active [Read only]
2		
1		
0	Update	A software update is available [Read only]

Image Storage State:

7	Local Att	Local Memory storage is attached (USB Memory Stick) [Read only]
6	Local Full	Local Memory is full [Read only]
5		
4	Remote Att	Remote Memory transfer is in progress. [Read only]
3	RemoteErr	Remote Memory Server Error
2		
1	Purge	Delete from Camera, if full size image has been successfully transferred, if enabled.
0		

USB State:

7	Pwr	USB Port power is applied [Read only]
6		
5		
4	USBF3	Detected USB Module function (0..15)
3	USBF2	
2	USBF1	
1	USBF0	
0	Active	USB Port connection has been established, and is working [Read only]

USB Module Function Table:

15	Ethernet	Ethernet Adapter		7		

14	Cellular	Cellular Adapter	6		
13	WiFi	WiFi Adapter	5		
12	Bluetooth	Bluetooth Adapter	4	Memory	Mass storage (USB memory drive/stick)
11	BLE	Bluetooth Low Energy Adapter	3	Camera M.S.	Camera, via mass storage interface
10	Satellite	Satellite Adapter	2	Camera PTP	Camera, via Picture Transfer Protocol
9			1	Unknown	Unidentified USB device detected
8			0	None	Not connected

Network Mode:

7	DHCP	Request address via DHCP (1) (else use fixed IP address for Network board)
6		
5		
4		
3		
2		
1		
0		

Time Options

7	DST	Daylight Savings time adjustments enabled if set, suppressed if clear
6	NT Updates	CAM & camera clocks may be synchronized to Network Time, including large changes
5	NT Bumps	CAM & camera clocks may be adjusted to Network Time, if less than 5 minute difference
4	GPS Updates	CAM & camera clocks may be synchronized to GPS Time, including large changes
3	GPS Bumps	CAM & camera clocks may be adjusted to GPS Time, if less than 5 minute difference
2		
1		
0		

Network State:
0: No network available
1: LAN only, static IP
2: LAN only, IP successfully issued via DHCP
3: Internet detected
4: Internet detected, but too slow to complete configured operation

Supported Incoming Module-Module Commands

Command Type #	Command Type	Content
01h	Set Parameter	Parameter Reference # (1 byte), followed by new value (1 or more bytes)
02h	Public Data Request	Source is requesting public data from destination
04h	Public Data Content	Source is sending public data to destination
08h	Execute Process	Process Reference # (1 byte), followed by modifier (1 or more bytes)

Set Parameter

Parameter	Parameter Name	Description	Value

Reference #			Bytes
02h	TimeZone	Preset time zone on the NM (ASCII string)	60
03h	USB0 Pwr	USB Host 0 Power 0>>Disable, any other value, Enable	1
04h	USB1 Pwr	USB Host 1 Power 0>>Disable, any other value, Enable	1
05h	Purge	1=Purge images from camera when transferred	1
0Dh	NDBootTime	Network Device boot time	1
0Eh	Time Options	Time Options	1

Execute Process

Process Reference #	Process Name	Description
01h	Conn WiFi Oper	Connect to the operator smartphone via WiFi
02h	Check Connection	
03h	Test A	Perform production test sequence A. Report “PASS” or “FAIL” in Log Message
04h	Test B	Perform production test sequence B. Report “PASS” or “FAIL” in Log Message
05h	Check for updates	Downloads software version list from update server, compares to current versions, and updates the “update available” bit if something newer is available.
06h	Upgrade software	Downloads and installs any newer software available on the update server.
07h	Check camera storage	Refresh camera total and available storage values.
08h	Check local storage	Refresh local (USB) total and available storage values.
09h	Safely Disc Local	Safely unmount USB storage filesystem (before unplugging/swapping)
0Ah	Pull Time	Preload network time from CAM real-time clock
0Bh	Test Picture Transfer	Take picture (if connected), send picture or file to network server, verbose Log comments
0Ch	Factory Reset	Reset parameters to factory defaults. All network and operator data will be erased, including all history of network connections, Logs, etc.
0Dh	Reboot	Reset the network hardware
0Eh	Shutdown	Perform clean shutdown
0Fh	Push Time	Push network time to CAM and camera clocks

Outgoing Announcement / Broadcast Messages

Announcement Reference #	Announcement Name	Description
01h	NetConn	The Network Board has connected to a network
02h	NetDisc	The Network Board has disconnected from a network
03h	SWUpdateBegin	Starting software update from Harbortronics server
04h	PicTransferred	Picture has been transferred to remote storage
05h	Ready	Network hardware has booted
06h	Unavailable	Network hardware is shutting down or otherwise unavailable

Network Access Module (NAM)

Function Type ID : FBh = 251
Port Number : 00h = 0

The Network Interface Module (NAM) is one of the Modules presented by the Network board.

Public Access Data

Item	Description	Bytes	Format	Position
Format	Repository Format Version (This Table=3)	1	Numeric	0
	*** Fixed Length Data Total ***	1		
Strings:	Each terminated with null character (included in length)			
Net Login	Network Storage Log-in String	20 max	String	
Net URL	Network Storage Location	36 max	String	
Rem Login	Remote Access Login	20 max	String	
APN	Cellular Network APN	36 max	String	

If the Net URL / APN is >36 characters, the Network Control Module will shorten the displayed string to the first 16 characters, a three dot ellipse, and the last 16 characters.

Supported Incoming Module-Module Commands

Command Type #	Command Type	Content
01h	Set Parameter	Parameter Reference # (1 byte), followed by new value (1 or more bytes)
02h	Public Data Request	Source is requesting public data from destination
04h	Public Data Content	Source is sending public data to destination

Set Parameter

Parameter Reference #	Parameter Name	Description	Value Bytes
02h	Net Login	Network Storage Log-in String	20 max
03h	Net Pass	Network Storage Password String	32 max
04h	Net URL	Network Storage Location	90 max
05h	Rem Login	Remote Access Login	20 max
06h	Rem Pass	Remote Access Password	32 max
07h	APN	Cellular Network APN	100 max
08h	SIM PIN	Cellular Network SIM card unlock PIN	8 max

Security: The NAM will only accept Set Parameter commands from user interface modules, such as the BLEM and DBUGM. The NAM will not present password / PIN parameters in Public Access Data.

Execute Process

Process Reference #	Process Name	Description

Outgoing Announcement / Broadcast Messages

Announcement Reference #	Announcement Name	Description

Email Module (EMAILM)

Function Type ID : F6= 246
Port Number : 00h = 0

The EmailModule (EMAILM) is one of the Modules presented by the Network board.

Public Access Data

Item	Description	Bytes	Format	Position
Format	Repository Format Version (This Table=2)	1	Numeric	0
SMTP Port	SMTP port	2	Numeric	1
	*** Fixed Length Data Total ***	3		0
Strings:	Each terminated with null character (included in length)			
Dest Addr	Notification Email	36 max	String	
SMTP Server	SMTP server address	36 max	String	
SMTP Username	SMTP username	36 max	String	

If any string parameter is greater than the field lengths defined above, the Email Module will shorten the displayed string to the first N characters, a three dot ellipse, and the last N characters, to fit the display field.

Supported Incoming Module-Module Commands

Command Type #	Command Type	Content
01h	Set Parameter	Parameter Reference # (1 byte), followed by new value (1 or more bytes)
02h	Public Data Request	Source is requesting public data from destination
04h	Public Data Content	Source is sending public data to destination
08h	Execute Process	

Set Parameter

Parameter Reference #	Parameter Name	Description	Value Bytes
01h	Dest Addr	Notification Email / SMS Address String	90 max
02h	SMTP Server	SMTP server address string	90 max
03h	SMTP Username	SMTP Username String	90 max
04h	SMTP Password	SMTP Password String	90 max
05h	SMTP Port	SMTP Port number	2

Execute Process

Process Reference #	Process Name	Description
01h	*Test*	Send Test Email

Outgoing Announcement / Broadcast Messages

Announcement Reference #	Announcement Name	Description

Wifi Module (WIFIM)

Function Type ID : F7h = 247
Port Number : 00h = 0

The Wifi Module shows saved networks and wifi scan results from the Network Board. It is implemented as a list. Each public data read will contain one list element. To request a different list element, use the Set List Index parameter, then do another Public Data Read. The last list element will have an index equal to Found – 1 (the list is zero-indexed).

Both saved networks and scan results will appear in the same list. The Saved flag indicates whether the network has previously been selected.

Typical usage:
Send Execute Command: Scan
Sleep 5 seconds
Read Public Access Data
For (i=0; i<Found; i++):
    Send Set Parameter Command: Set List Index i
    Read Public Access Data
    Print SSID, RSSI

Public Access Data

Item	Description	Bytes	Format	Position
Format	Repository Format Version (This Table=2)	1	Numeric	0
Found	Number of networks found, and in list	1	Numeric	1
Index	SSID List Index	1	Numeric	2
RSSI	Signal strength of network corresponding to Index	1	RSSI	3
State	WiFi Module Status	1	WiFi State	4
Connected	Index of connected network, 0 if none	1	Numeric	5
	*** Fixed Length Data Total ***	6		
Strings:	Each terminated with null character (included in length)			
SSID	SSID of network corresponding to the Index	32 max	String	

RSSI: Signed Byte, value TBD

WiFi State:

7	Connected	WiFi has connected to a network
6	Saved	1 if selected SSID index has been previously saved 0 if selected SSID index is an unsaved scan result
5		
4		
3		
2		
1		
0		

Supported Incoming Module-Module Commands

Command Type #	Command Type	Content
01h	Set Parameter	Parameter Reference # (1 byte), followed by new value (1 or more bytes)
02h	Public Data Request	Source is requesting public data from destination
04h	Public Data Content	Source is sending public data to destination
08h	Execute Process	Process Reference # (1 byte), followed by modifier (1 or more bytes)

Set Parameter

Parameter Reference #	Parameter Name	Description	Value Bytes
01h	Index	Specify one of the listed WiFi networks for subsequent operations.	1
02h	WPA-PSK Login	Add WPA-PSK Login String “SSID[null]PSK” to the list of WiFi connections.	90 max

Execute Process

Process Reference #	Process Name	Description
01h	Scan	Start scanning for networks.
03h	Discard	Remove a saved network from the list the Network Board is allowed to connect to. Note that the network may still appear in the list as a scan result if it is in range.
04h	Move Up	Increase the connection priority of the saved network at the current list index. It is suggested that the list be iterated after moving an item.
05h	Move Down	Decrease the connection priority of the saved network at the current list index. It is suggested that the list be iterated after moving an item.

Outgoing Announcement / Broadcast Messages

Announcement Reference #	Announcement Name	Description

GPS Module (GPSM)

Function Type ID : FAh = 250
Port Number : 00h = 0

The GPS Module can be implemented in two ways. The implementation should not be relevant to the registration process or use.
1) A bare GPS receiver device is connected to the Network board and is registered via indirect method.
2) A GPS Module could be designed, and connected directly to a DSPC AM Port.

Public Access Data

Item	Description	Bytes	Format	Position
Format	Repository Format Version (This Table=1)	1	Numeric	0
GPS State	GPS Module State	1	GPS State	1
GPS Acq Mode	When are GPS fixes acquired	1	GPS Acq Mode	2
GPS Interval	GPS Acquisition Interval	1	TimeExp	3
LGS Mode	Local GPS Storage Mode (i.e. memory stick)	1	GPS Store Mode	4
LGS Acqs	Store after N acquisitions	2	Numeric	5
RGS Mode	Remote GPS Storage Mode (network)	1	GPS Store Mode	7
RGS Acqs	Store after N acquisitions	2	Numeric	8
Queued	Number of GPS fixes stored in Queue	2	Numeric	10
Index	Queue Index (0=Current fix, 1=most recent queued fix, ...)	2	Numeric	12
	*** Fixed Length Data Total ***	14		
Strings:	Each terminated with null character (included in length)			
GPS Pos	Last acquired / queued GPS Position/Time string	36 max	$GPGLL	

GPS Storage Mode:
0: Never. Network board is never awaken for data transfer.
1: After new GPS fix.
2: After N sets of new data collected
3: Periodically, Every night
4: Periodically, Every Sunday night

GPS Acquisition Mode:
0: Never
1: After every pic is taken
2: After N pics are taken
3: Periodically (interval)

GPS State:

7	Pwr	GPS Module power is applied [Read only]
6	Active	GPS Module connection has been established, and is working [Read only]
5		
4		

3		
2		
1		
0		

Supported Incoming Module-Module Commands

Command Type #	Command Type	Content
01h	Set Parameter	Parameter Reference # (1 byte), followed by new value (1 or more bytes)
02h	Public Data Request	Source is requesting public data from destination
04h	Public Data Content	Source is sending public data to destination
08h	Execute Process	Process Reference # (1 byte), followed by modifier (1 or more bytes)

Set Parameter

Parameter Reference #	Parameter Name	Description	Value Bytes
01h	GPS Pwr	GPS Module Power 0>>Disable, any other value, Enable	1
02h	GPS Acq Mode	Set when GPS fixes are taken	1
03h	GPS Interval	GPS Acq Mode - Set interval for acquisitions	1
04h	LGS Mode	Local GPS Storage Mode	1
05h	LGS Acqs	Local GPS Storage Mode - # Acquisitions	2
06h	RGS Mode	Remote GPS Storage Mode	1
07h	RGS Acqs	Remote GPS Storage Mode - # Acquisitions	2
08h	Index	Set Queue index, to select GPS fix to display	1

Execute Process

Process Reference #	Process Name	Description

Outgoing Announcement / Broadcast Messages

Announcement Reference #	Announcement Name	Description

Servo360 Module (S360M)

Function Type ID : F9h = 249
Port Number : Assigned

The Servo 360 is a single axis motion control Module. Motion may be controlled manually, or via a queue of Way Points. Way Points are a Position/Delta-Time entity, and the Servo360 can move from one to another, in a linked fashion. Shoot-Move-Shoot or continuous movement may be implemented at the next higher level of control.

Public Access Data

Item	Description	Bytes	Format	Position
Format	Repository Format Version (This Table=1)	1	Numeric	0
Firmware	Firmware Version	1	Numeric	1
SSLH	Seconds Since Last Home	3	Secs24	2
State	Current Operational State	1	S360State	5
Position	Current Position (out of date if not stopped)	4	S360Pos	6
Speed	Current Rate of Speed	1	S360Spd	10
ConstantP	Proportional Constant for position control	1	Numeric	11
ConstantI	Integral Constant for position control	1	Numeric	12
ConstantD	Differential Constant for position control	1	Numeric	13
MaxSpeed	Internal Max Speed, per factory, no load	1	S360Spd	14
Goal Position	Position at next way point	4	S360Pos	15
Goal Speed	Speed to next way point	1	S360Spd	19
Goal Time	Time to next way point	3	mSec24	20
Way Points	Number of Way Points left to reach	1	Numeric	23
	*** Fixed Length Data Total ***	24		

Secs24: Number of seconds (0..16,777,215) >> 194 days.
May overflow/underflow. Might want to reset MC if number gets high. Value=0 may indicate Never.
mSecs24: Number of milliseconds seconds (0..16,777,215) >> 4.6 hours.
S360Pos: 32 Bit position signed value, relative to last Home process
S360Spd: 8 Bit logarithmic speed index (TBD), 2s complement for directionality

S360 State:

7	Moving	Moving (1) / Stopped (0)
6	Home	1 if at Home position
5	WayPoint	1 if at current Way Point position
4	Empty	1 if Way Point queue is empty (have reached last Way Point)
3	Full	1 if Way Point queue is full (cannot handle more Way Points)
2		
1		
0	Broken	1 if hardware fault detected

Supported Incoming Module-Module Commands

Command	Command Type	Content

Type #		
01h	Set Parameter	Parameter Reference # (1 byte), followed by new value (1 or more bytes)
02h	Public Data Request	Source is requesting public data from destination
04h	Public Data Content	Source is sending public data to destination
08h	Execute Process	Process Reference # (1 byte), followed by modifier (1 or more bytes)
20h	Firmware	Data Transfer of arbitrary # bytes, in multiple messages, as firmware update.

Set Parameter

Parameter Reference #	Parameter Name	Description	Value Bytes
01h	Add Way Point	Add a Way Point to queue (S360Pos, mSec24)	5
02h	Constants	Set PID constants	3
03h	Speed	Set manual speed (S360Spd)	1

Execute Process

Process Reference #	Process Name	Description
01h	Home	Current position and time are zeroed
02h	Move Next	Start/Continue movement toward next Way Point
03h	Stop	Stop motion
04h	Clear Way Points	Clear all programmed way points (stops if currently in motion to any Way Point)
05h	Delete Way Point	Delete last queued Way Point (stops if currently in motion to that Way Point)
06h	Move at Speed	Move manually, at given speed
07h	Return Home S	Move back to Home position, at given speed
08h	Return Local S	Move back to same rotor orientation as Home, without unwinding, at given speed
09h	Return Home F	Move back to Home position, at maximum speed
0Ah	Return Local F	Move back to same rotor orientation as Home, without unwinding, at maximum speed
10h	Move All	Move through all Way Points in queue

Outgoing Announcement / Broadcast Messages

Announcement Reference #	Announcement Name	Description
01h	AtWayPoint	The Servo360 has reached the goal Way Point.
02h	LastWayPoint	The Servo360 has reached the last queued Way Point
03h	StartingMove	The Servo360 is about to start motion

Light Sensor Module (LSM)

Function Type ID : F8h = 248
Port Number : Assigned

The LightSensor is a high dynamic range module that returns the amount of ambient light detected. The amount of light may used for presetting the camera exposure, or for simple day/night detection. Each request for public data will cause the Light Sensor Module to refresh the reading… public data will always be fresh.

Public Access Data

Item	Description	Bytes	Format	Position
Format	Repository Format Version (This Table=1)	1	Numeric	0
Firmware	Firmware Version	1	Numeric	1
SSLM	Seconds Since Last Measurement	3	Secs24	2
State	Current Operational State	1	LSState	5
Intensity	Last measured light intensity	2	LSIntensity	6
UpdPer	Period of measurement update	2	Secs16	8
LSCalib	Calibration Value(s)	4	LSCal	10
	*** Fixed Length Data Total ***	14		

Secs16: Number of seconds (0..65525) >> 18.2hrs. May overflow/underflow.
LSIntensity: 16 Bit light intensity value...TBD
LSCal: Factory Calibration value … TBD

LSState:

7		
6		
5		
4		
3		
2		
1	Saturated	Set of sensor is at maximumintensity
0	Broken	1 if hardware fault detected

Supported Incoming Module-Module Commands

Command Type #	Command Type	Content
01h	Set Parameter	Parameter Reference # (1 byte), followed by new value (1 or more bytes)
02h	Public Data Request	Source is requesting public data from destination
04h	Public Data Content	Source is sending public data to destination
08h	Execute Process	Process Reference # (1 byte), followed by modifier (1 or more bytes)
20h	Firmware	Data Transfer of arbitrary # bytes, in multiple messages, as firmware update.

Set Parameter

Parameter	Parameter Name	Description	Value

Reference #			Bytes

Execute Process

Process Reference #	Process Name	Description

Outgoing Announcement / Broadcast Messages

Announcement Reference #	Announcement Name	Description

Debug Module (DBUGM)

Function Type ID : 01h = 1
Port Number : Assigned

The Debug Module is essentially a smart terminal connection, allowing debug access to the Hub. It has no personality of it's own. It will register with the Hub as needed. It will be implemented as software on a Windows based computer, and used for testing and initialization.

Public Access Data

Item	**Description**	**Bytes**	**Format**

Supported Incoming Module-Module Commands

Command Type #	Command Type	Content

Outgoing Announcement / Broadcast Messages

Announcement Reference #	Announcement Name	Description