Protocol Version 2 Proposal

Message size between 15 - 84 bytes
[**start byte | options | data length | fragment no**. | **destination | command/data**]

section	size (bytes)
start byte	1
options	1
data length	1
fragment no.	2
destination	10
command/data	0-69

Start Byte

Single byte. Indicates the beginning of a message. Value is always 0x0D

Options

One byte. Contains bit-masked flags used to set various options in the transmission. Most of the fields are reserved for now. This should allow the protocol to add features over time.

Bit values:
7 - Binary/ASCII transmission indicator. A value of 0 indicates that the data block is ASCII text. A value of 1 indicates that the data is a binary transmission
6 - reserved
5 - reserved
4 - reserved
3 - reserved
2 - reserved
1 - reserved
0 - reserved

Data Length

One byte. Indicates the length of the data in the command/data field. Due to limitations in the Zigbee protocol the max value is 69. A value of 0 is valid.

Fragment Number

A two-byte field used to indicate the fragment number of this transmission. If the size of the data block supplied for transmission is greater than 69 bytes the message will be split into multiple messages, each with a decreasing fragment number. The maximum number of fragments for a single message is 65536 (2^16). This makes the maximum data size of the original data block 4521984 bytes (a little over 4.3 MB). The transmission rate of the Zigbee network will probably not be fast enough to stream live video, but this should be enough for a single message to hold a high quality uncompressed image capture (or very high if the module does automatic compression). If a message from a remote module to a driver with a data block of size 200 is queued it will be broken into the following fragments:

message 1: fragment no 2, data block size 69
message 2: fragment no 1, data block size 69
message 3: fragment no 0, data block size 62

Note that the fragment numbers appear in decreasing order. This allows the backend to know the total number of fragments remaining to be received, the ordering of the fragments, and the estimated size of the transfer. On the receiving side the Coordinator will notice an incoming message addressed to the driver with a fragment number of 2. It can set aside a byte buffer to accommodate the remaining fragments, and, when they arrive, add those fragments onto the buffer. When the third message (fragment 0) is received it appends that message's data onto the buffer, reads all the data in the buffer as a byte[], passing it to the driver for processing.

Destination

The unique name of the remote module or driver that this message should be routed to. The length of this field is fixed at 10 bytes. The data is 8-bit ASCII. Every driver and remote module should have a unique name that is 10 characters or less, and only use characters found in the 8-bit ASCII table. If the destination name is less than 10 bytes it is left shifted, and padded on the right with 0x00 bytes (the null character in ASCII).

Command/Data Field

The payload of the transmission. The max size of the data field is 4521984 bytes/characters. If the command/data is larger than 69 bytes then the message will be split into as many fragments as needed. Since each fragment is, in itself, a complete message there will be a significant overhead introduced when fragmenting the message. If possible, the command/data field should be kept under 69 bytes.