Image Data Frame Specification

The specification below demonstrates how JPEG data is organized and passed from the camera module to the IHU. The target JPEG image size is 640x480 pixels. The JPEG algorithm processes an image in a raster-scan, in units of 8x8 pixel blocks. A reasonably sized chunk of data suitable for a payload is the compressed data from one scan line. A 'scan line' represents the raster scan of an 8-pixel tall and 640-pixel wide segment of the image. There are 60 'scan lines' within the 640x480 image.

Due to the nature of the JPEG compression, it is impossible to know *a priori* the number of bytes of the resulting compression. A compression ratio of 20:1 yields a resulting size on the order of 768 bytes for one scan line. The 20:1 compression ratio is measured from 24-bit RGB (3 bytes per pixel).

Frame Data Structure

```
typedef struct{
  uint16_t       header = ((line_id & 0x3F) << 10) |
                          (payload_length & 0x3FF);
  uint8_t        payload[payload_length];
  uint16_t       chksum;
} data_frame;
```

Description

header is a 16-bit word that contains two pieces of information. The first 6 bits are the *line_id* of the payload. The *line_id* ranges $0 \leq line\_id < 60$ and represents which line from the 60 possible scan lines are contained in the payload. The 'uppermost' line is $line\_id = 0$. The 10 LSB of *header* is the length of the frame payload *payload_length*, in bytes. Thus, the maximum payload is 1023 bytes.

payload is an array of bytes, *payload_length* long.

chksum is a 16-bit checksum of the payload

Checksum Specification

The camera module will compute a 16-bit checksum of all payloads according to:

```
  uint16_t byte = 0;
  uint16_t chksum = 0;
```

```
  for( ; byte < payload_length; byte++ )
    chksum += payload[byte];

  chksum = ~chksum + 1;
```

To verify the checksum, the checksum is computed on the receiver, then summed with the transmitted checksum. If the result is zero, then it is probable that no errors occurred.

UART Communication Specification

The IHU and Camera will communicate using 2-byte command and reply words. Only the data frame will exceed the 2-byte transmission length.

Command	**Origin**	**Destination**	**Description**
RR	IHU	Camera	Is Camera ready?
NN	Camera	IHU	The Camera is not ready
YY	Camera	IHU	The Camera is ready
TT	IHU	Camera	Transmit the next frame
{FRAME}	Camera	IHU	Data Frame