„STYPE A5“ TRACKING PROTOCOL

Connectivity options:

Option 1 - Serial connection

Serial connection - RS-422 interface
Baud rate: 38400
Parameters: 1 start, 1 stop bit, even parity

Option 2 – Ethernet socket connection

UDP over IP packages.
IP addresses and destination port are configurable on the Stype kit console.

Default source IP address:	10.10.10.10
Default destination IP address:	10.10.10.11
Default port:	6301
Source port (fixed):	0
Destination MAC address (fixed):	ff:ff:ff:ff:ff:ff (broadcast)

Data sending is synchronized with corresponding video blackburst signal, so one package is being sent in each video field (or frame for progressive formats).

This is a binary protocol. Constant length of each packet equals 54 bytes.

Package format

Pos	Name	Size (bytes)	Value Format	Description
0	Header	1	Const	Fixed value: 0xA5
1	X	4	Float, little endian	Position: + Right (meters)
5	Y	4	Float, little endian	Position: + Up (meters)
9	Z	4	Float, little endian	Position: - Look (meters)
13	Pan	4	Float, little endian	Orientation: + Pan to the right (degrees)
17	Tilt	4	Float, little endian	Orientation: + Tilt up (degrees)
21	Roll	4	Float, little endian	Orientation: + Roll Clockwise (degrees)
25	FovX	4	Float, little endian	Horizontal field of view (degrees)
29	FovY	4	Float, little endian	Vertical field of view (degrees)
33	Focus	4	Float, little endian	0-close, 1-far (infinite)
37	k1	4	Float, little endian	First radial distortion harmonic (mm^{-2}), normalized to 8.8 mm x 6.6 mm CCD diagonal
41	k2	4	Float, little endian	Second radial distortion harmonic (mm^{-4}), normalized to 8.8 mm x 6.6 mm CCD diagonal
45	Center-X	4	Float, little endian	Horizontal center zoom shift (in pixels)
49	Center-Y	4	Float, little endian	Vertical center zoom shift (in pixels)
53	Checksum	1	Unsigned char	Unsigned sum of all preceeding bytes

All floating point numbers are sent in a little-endian format.

Explanation of field values:

X – Camera eyepoint position in meters along the X axis. Description of axis orientation is described later in the document on illustration.

Y – Camera eyepoint position in meters along the Y axis.

Z – Camera eyepoint position in meters along the Z axis.

Pan – Camera Pan movement. Represented in degrees <-180, 180]. Description of Pan, tilt and roll orientation is described later in the document.

Tilt – Camera Tilt movement. Represented in degrees <-180, 180].

Roll – Camera Roll movement. Represented in degrees <-180, 180].

FovX – Horizontal field of view. Represented in degrees <0, 180].

FovY – Vertical field of view. Written in degrees <0, 180]. Aspect ratio of FovX and FovY is not fixed over the entire lens range, but it is changing over the zoom/focus range. Therefore it is advisable to implement separate FovX and FovY in a render engine, altough using a fixed value will provide usable result.

Focus – Position of a lens focus point. 0-closest, 1-infinite. This value can be used for generating picture with depth of field effect.

K_1 – Lens radial distortion first harmonic parameter. Represented in (mm^{-2}).

The method used for radial distortion calculation in not important, since the Stype kit calibration is done visually. So far it has been tried with the following formulas (center shift offset removed for the sake of simplicity):

- Distorted = Undistorted * ($1 + r*K_1 + r^2*K_2$)
- Undistorted = Distorted * ($1 + r*K_1 + r^2*K_2$)

K_2 – Second radial lens distortion harmonic. Written in (mm^{-4}).

CenterX – Horizontal mounting shift of the lens optical axis from the center of the CCD chip. Represented in pixels.

CenterY – Vertical mounting shift of the lens optical axis from the center of the CCD chip. Represented in pixels.

Checksum – Unsigned sum of all previous bytes. C++ code example of a generating checksum from a received package is given on the next page.

Checksum C++ code example

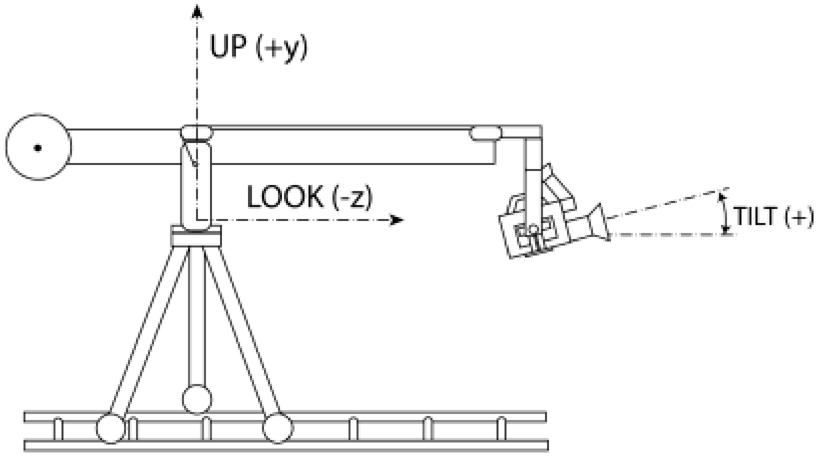
We calculate checksum by adding the unsigned char representation of all previous bytes together.

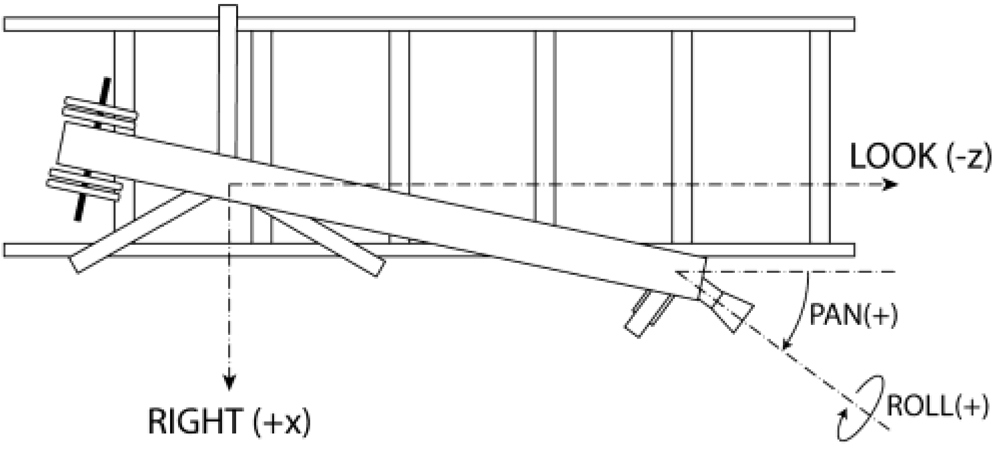
```
union {
      unsigned char element [54];   //used for checksum calculation
      struct StypePacket {
            unsigned char header;
            float X;
            float Y;
            float Z;
            float Pan;
            float Tilt;
            float Roll;
            float FovX;
            float FovY;
            float Focus;
            float K0;
            float K1;
            float CSX;
            float CSY;
            unsigned char checksum;
      } StypeARPacket;
};
unsigned char check = 0;
for (char x=0; x<53; x++) check += StypeARPacket.element[x];

if (check == StypeARPacket.checksum) printf("CheckSum is OK!");
```

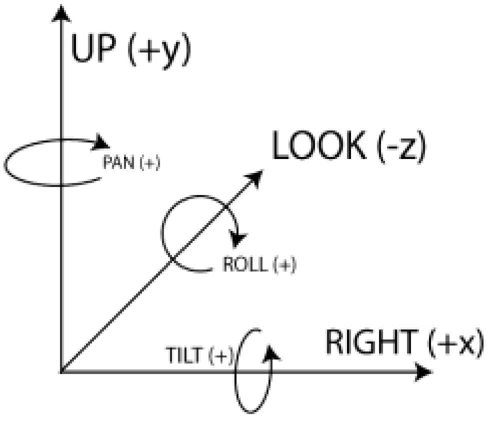
Although Ethernet packets already have multiple headers and checksums built in by the IP and Ethernet protocol, we are still using our own header and checksum calculation for situations where data is sent over the serial connection.

Axis orientation

Pictures show the direction of each axis, and direction of rotation. The zero point of a studio can be set anywhere in space easily through our interface, but the default value is show on picture.


UP (+y)
LOOK (-z)
TILT (+)


LOOK (-z)
PAN(+)
RIGHT (+x)
ROLL(+)


UP (+y)
PAN (+)
LOOK (-z)
ROLL (+)
TILT (+)
RIGHT (+x)