r

Shenzhen Hi-Link Electronic Co., Ltd

HLK-LD6002C
Communication Protocol
Fall detection

r

CONTENT

1. Application Project

For different application projects, all the messages related to TF frames are listed for users' reference and complete analysis.For the message classes and message data bits that appear in documents, they are equipped with corresponding actual projects.

1.1. Fall detection items

Message Type: Report the fall test item test results 0x0E02

The Message type is 0x0E02, and only the one-way data transfer mode is supported.

The radar sends data to the upper computer: MSG _ IND _ FALL _ STATUS					
form	Byte number	fundamental type	frame structure	Example frames	The frame meaning
SOF	1 byte	uint8	Start frame	01	To report the fall test results.
ID	2 byte	uint16	frame ID	00 00	
LEN	2 byte	uint16	Data frame length	00 01	
TYPE	2 byte	uint16	frame type	0E 02	
HEAD_CKSUM	1 byte	uint8	Head checksum	F3	
DATA	1 byte	uint8	[is_fall]	01	
DATA_CKSUM	1 byte	uint8	Data checksum	FE	

The following is the corresponding meaning of each DATA-bit:

- [is _ fall]: Judge whether to fall.
  - Value 0: normal.
  - Value 1: a fall.

Message Type: Set the installation height of 0x0E04

The message type is 0x0E04 and supports a two-way data transfer mode.

The upper computer sends the data to the radar: MSG _ IND _ FALL _ SET _HIGH					
form	Byte number	fundamental type	frame structure	Example frames	The frame meaning
SOF	1 byte	uint8	Start frame	01	Use to set the radar mounting height.
ID	2 byte	uint16	frame ID	00 00	
LEN	2 byte	uint16	Data frame length	00 04	
TYPE	2 byte	uint16	frame type	0E 04	
HEAD_CKSUM	1 byte	uint8	Head checksum	F0	
DATA	4 byte	float	[high]	00 00 20 40	
DATA_CKSUM	1 byte	uint8	Data checksum	9F	

The following is the corresponding meaning of each DATA-bit:

- [High]: Set the radar installation height, ranging from 1-5 m.

Convert to float: for example, [x_point] bit 0x66,0x66,0xA2,0x41, first spell uint 32 bit shaping, because the TF frame Data bit small end order, so the value is 0x 41A26666, and then float type strong turn, the final result is: 20.3.

```c
int  main(void )
{
unsigned int param = 0x41A 26666;
float  res = *(float  *)&param;

printf("data: %f\n", res);
return  0;
}
```

Message type: return the result 0x0E04 after setting the installation height

Return back data to the upper computer: MSG _ IND _ FALL _ RES					
form	Byte number	fundamental type	frame structure	Example frames	The frame meaning
SOF	1 byte	uint8	Start frame	01	Use to return the status after setting the radar installation height.
ID	2 byte	uint16	frame ID	00 00	
LEN	2 byte	uint16	Data frame length	00 01	
TYPE	2 byte	uint16	frame type	0E 04	
HEAD_CKSUM	1 byte	uint8	Head checksum	F5	
DATA	1 byte	uint8	[result]	01	
DATA_CKSUM	1 byte	uint8	Data checksum	FE	

When the radar receives a TYPE of 0x0E04, the radar will return the data.[result] There are two outcomes:

- Value 0: failed to set the height.
- Value 1: set the height successfully.

Message type: Get the radar parameter 0x0E06

The message type is 0x0E06 and supports a two-way data transfer mode.

The upper computer sends the data to the radar: MSG _ IND _ FALL _ GET _HIGH					
form	Byte number	fundamental type	frame structure	Example frames	The frame meaning
SOF	1 byte	uint8	Start frame	01	Get the parameters of the lower machine
ID	2 byte	uint16	frame ID	00 00	
LEN	2 byte	uint16	Data frame length	00 00	
TYPE	2 byte	uint16	frame type	0E 06	
HEAD_CKSUM	1 byte	uint8	Head checksum	F6	

The upper computer issues a command to the radar to request the data of the lower computer. The radar will reply the signal of 0x0E06, see below in details

Message type: return the result 0x0E06

Return back data to the upper computer: MSG _ IND _ FALL _ GET _ HIGH					
form	Byte number	fundamental type	frame structure	Example frames	The frame meaning
SOF	1 byte	uint8	Start frame	01	Send the parameters of the radar to the upper computer
ID	2 byte	uint16	frame ID	00 00	
LEN	2 byte	uint16	Data frame length	00 1C	
TYPE	2 byte	uint16	frame type	0E 06	
HEAD_CKSUM	1 byte	uint8	Head checksum	/	
DATA	4 byte	f loat	[high]	/	
D ATA	4 byte	float	[threshold]	/	
D ATA	4 byte	uint32	[sensitivity]	/	
D ATA	4 byte	float	[rect_XL]	/	
D ATA	4 byte	float	[rect_XR]	/	
D ATA	4 byte	float	[rect_ZF]	/	
D ATA	4 byte	float	[rect_ZB]	/	
DATA_CKSUM	1 byte	uint8	Data checksum	/	

When the radar receives a TYPE of 0x0E06, the radar will return the data. There are two results:

- Value of 0: The fetch failed.
- Value is other: get success.

Message Type: Set the fall threshold of 0x0E08

The message type is 0x0E08 and supports two-way data transfer mode.

The host computer sends the data to the radar: MSG _ IND _ FALL _ SET _ THRESHOLD					
form	Byte number	fundamental type	frame structure	Example frames	The frame meaning
SOF	1 byte	uint8	Start frame	01	The fall threshold is set, and the default radar fall threshold is 0.6m.
ID	2 byte	uint16	frame ID	00 00	
LEN	2 byte	uint16	Data frame length	00 04	
TYPE	2 byte	uint16	frame type	0E 08	
HEAD_CKSUM	1 byte	uint8	Head check and	FC	
DATA	4 byte	float	[set_threshold]	9A 99 19 3F	
DATA_CKSUM	1 byte	uint8	Data verification and	DA	

Convert to float: for example, [high] bit 0x9A, 0x99,0x19,0x3F, first spell uint 32 bit shaping, because
the TF frame Data bit small end order, so the value is 0x 3F19999A, and then the float type strong turn, the final result is: 0.6.

```
1.  int  main(void )
2.  {
3.  unsigned int param = 0x41A 26666;
4.  float  res = *(float  *)&param;
5.
6.  printf("data: %f\n", res);
7.  return  0;
```

```
8.  }
```

Message type: return the result 0x0E08 after setting the fall threshold

Return back data to the upper computer: MSG _ IND _ FALL _ SET _ THRESHOLD					
form	Byte number	fundamental type	frame structure	Example frames	The frame meaning
SOF	1 byte	uint8	Start frame	01	After setting the fall threshold, return the state.
ID	2 byte	uint16	frame ID	00 00	
LEN	2 byte	uint16	Data frame length	00 01	
TYPE	2 byte	uint16	frame type	0E 08	
HEAD_CKSUM	1 byte	uint8	Head checksum	F7	
DATA	1 byte	uint8	[value]	01	
DATA_CKSUM	1 byte	uint8	Data checksum	FE	

When the radar receives a TYPE of 0x0E08, the radar will return the data. Returned data has two outcomes:

- Value 0x 00: Failure to obtain.
- Value 0x 01: Successful acquisition.

Message Type: Set the fall sensitivity of 0x0E0A

The message type is 0x0E0A and supports two-way data transfer mode.

The upper computer sends the data to the radar: MSG _ IND _ FALL _ SET _SENSITIVITY					
form	Byte number	fundamental type	frame structure	Example frames	The frame meaning
SOF	1 byte	uint8	Start frame	01	Set fall sensitivity, initial value of 3, represent 3 frames of data.(Range 3~10)
ID	2 byte	uint16	frame ID	00 00	
LEN	2 byte	uint16	Data frame length	00 01	
TYPE	2 byte	uint16	frame type	0E 0A	
HEAD_CKSUM	1 byte	uint8	Head checksum	FB	
DATA	4 byte	uint32	[high]	03 00 00 00	
DATA_CKSUM	1 byte	uint8	Data checksum	FE	

Set fall sensitivity, initial value of 3, represent 3 frames of data.(Range 3~10)

Message type: return the result 0x0E0A after setting the fall sensitivity

Return back data to the upper computer: MSG _ IND _ FALL _ SET _ SENSITIVITY					
form	Byte number	fundamental type	frame structure	Example frames	The frame meaning
SOF	1 byte	uint8	Start frame	01	When the radar receives the TYPE of 0x0E0A, the radar will return the data. Returned data has two outcomes: Value of 0: The fetch failed. Value of 1: Success was obtained.
ID	2 byte	uint16	frame ID	00 00	
LEN	2 byte	uint16	Data frame length	00 01	
TYPE	2 byte	uint16	frame type	0E 0A	
HEAD_CKSUM	1 byte	uint8	Head check and	FB	
DATA	1 byte	uint8	[value]	01	
DATA_CKSUM	1 byte	uint8	Data verification and	FE	

Message type: height upload result 0x0E0E

Return back data to the upper computer: MSG _ IND _ FALL _ SET _ SENSITIVITY					
form	Byte number	fundamental type	frame structure	Example frames	The frame meaning
SOF	1 byte	uint8	Start frame	01	The radar automatically uploads the height information of the current target point
ID	2 byte	uint16	frame ID	00 00	
LEN	2 byte	uint16	Data frame length	00 01	
TYPE	2 byte	uint16	frame type	0E 0E	
HEAD_CKSUM	1 byte	uint8	Head checksum	FB	
DATA	1 byte	Uint 32	[value]	01 00 00 00	
DATA_CKSUM	1 byte	uint8	Data checksum	FE	

Message Type: User log message 0x0E01

Message type is 0x0E01 and only supports one-way data transfer mode

The upper computer sends the data to the radar					
form	Byte number	fundamental type	frame structure	Example frames	The frame meaning
SOF	1 byte	uint8	Start frame	01	0: Turn off the User log information 1: Open the User log information
ID	2 byte	uint16	frame ID	00 00	
LEN	2 byte	uint16	Data frame length	00 04	
TYPE	2 byte	uint16	frame type	0E 01	
HEAD_CKSUM	1 byte	uint8	Head checksum	FB	
DATA	4 byte	uint32	[value]	01 00 00 00	
DATA_CKSUM	1 byte	uint8	Data checksum	FE	

- Value of 0x 00: Close the User log information.
- Value of 0x 01: Open the User log information.

Message type: Set the alarm area parameter 0x0E0C

The message type is 0x0E0C and supports two-way data transfer mode

The upper computer sends the data to the radar					
form	Byte number	fundamental type	frame structure	Example frames	The frame meaning
SOF	1 byte	uint8	Start frame	01	Set the range of fall alarm area with the range of 0.3~1.5
ID	2 byte	uint16	frame ID	00 00	
LEN	2 byte	uint16	Data frame length	00 10	
TYPE	2 byte	uint16	frame type	0E 0C	
HEAD_CKSUM	1 byte	uint8	Head checksum	68	
DATA	4 byte	float	[rect_XL]	00 00 00 3F	
DATA	4 byte	float	[rect_XR]	00 00 00 3F	
DATA	4 byte	float	[rect_ZF]	00 00 00 3F	
DATA	4 byte	float	[rect_ZB]	00 00 00 3F	
DATA_CKSUM	1 byte	uint8	Data verification and	FF	

Set the range of fall alarm area with the range of 0.3~1.5

Message type: set the alarm area parameter return result 0x0E0C

The radar sends the data to the upper position computer					
form	Byte number	fundamental type	frame structure	Example frames	The frame meaning
SOF	1 byte	uint8	Start frame	01	0: Failure 1: Success
ID	2 byte	uint16	frame ID	00 00	
LEN	2 byte	uint16	Data frame length	00 04	
TYPE	2 byte	uint16	frame type	0E 0C	
HEAD_CKSUM	1 byte	uint8	Head checksum	FB	
DATA	1 byte	u int 8	[value]	01	
DATA_CKSUM	1 byte	uint8	Data checksum	FE	

- Value of 0x 00: Failure to obtain.
- Value of 0x 01: Successful acquisition.

Message type: Radar initialization setting parameter 0x2110

The message type is 0x2110 and supports one-way data transfer mode

The upper position computer sends the data to the radar					
form	Byte number	fundamental type	frame structure	Example frames	The frame meaning
SOF	1 byte	uint8	Start frame	01	Radar initialization setting parameters High:2.2 Threshold:0.5 Sensitivity:3 Rect_XL:0.5 Rect_XR:0.5 Rect_ZF:0.5 Rect_ZB:0.5
ID	2 byte	uint16	frame ID	00 00	
LEN	2 byte	uint16	Data frame length	00 00	
TYPE	2 byte	uint16	frame type	2110	
HEAD_CKSUM	1 byte	uint8	Head checksum	F7	
DATA_CKSUM	1 byte	uint8	Data checksum	FE	

When TYPE is 0x2110, radar initialization sets parameter.
High:2.2
Threshold:0.5
Sensitivity:3
Rect_XL:0.5
Rect_XR:0.5
Rect_ZF:0.5
Rect_ZB:0.5
The settings is related to the content of TYPE type 0x0E06

2. Programming interface

2.1. Encode the TF message

void tinyFramefTx(TF_TYPE type, uint8 *data, TF_LEN len);

Where type is the sending data type, the uint16 type, such as the personnel detection data result report, the data type is 0x0A10. See 4.2.1.6 for details

Uint8 * Data is the address that sends the data.

Len is the length of the sent data, the uint16 type.

2.2. Decoding the TF message

TinyFrameRx tinyFramefRx(void);

After successfully receiving the message, the received data is returned to a variable of the TinyFrameRx type.

The resolution of each CKSUM is shown as follows:

HEAD _ CKSUM: TF frame header checksum [Starting from the first byte to the previous byte of the HEAD CKSUM bit]

DATA _ CKSUM: TF data test and [The first byte of DATA to the last byte of the DATA CKSUM bit] The method c code for calculating C KSUM is as follows:

```
1.  unsigned char  getCksum(unsigned char  *data, unsigned char  le
2.  {
3.  unsigned char  ret = 0;
4.
5.  for  (int  i = 0; i < len; i++
6.  ret = ret ^ data[i];
7.
8.  ret = ~ret;
9.
10. return  ret;
11. }
```

2.3. Example code

If you want to analyze the demo of TF frame data (including C message in Linux environment and Python language demo) in Keil µVision5 environment, you can directly communicate with the sales.

Shenzhen Hi-Link Electronic Co., Ltd

Address： 17F,Building E, Xinghe WORLD, Minzhi Street, Long Hua district, Shenzhen 51813

Email： sales@hlktech.com

Website： www.hlktech.net