ADAS Communication Protocol

Contents

Server side configuration source code

NOTE: This is a "test server" version, dedicated for initial functionality testing (which could also be done via cloudview platform) with 1-2 cameras. Multithreaded (and multiple device) handling on the server side is up to the client to do additional development and/or necessary optimizations and modifications, to support more than 1-2 cameras at the same server instance.

Optionally - platforms that already support camera integrations could be used.

Together with the communication protocol, this server side source code can be downloaded and implemented upon personal servers for the ease of use and configuration, please see index.js, debug.js and protocol.js which can be copied into the server.

Server files & source code can be downloaded here.

Please see index.js, debug.js and protocol.js files. These files contain required support for camera downloads and they can be inserted straight into server. The folder can also be run as server and will work with file downloads, so if personal server is not available, there is always a possibility to use this default server.

All that is required is an open Port on the computer and that port has to be written in run_server.bat file. Also by clicking on Readme file, you will be able to see the settings which you can enter into that run_server.bat file by right clicking the mouse button on it and choosing edit option. These settings include choices like DualCam download, ADAS file download or Auto mode, Metadata

inclusion and etc.

In this case, Notepad++ is chosen but you can choose any text editor in order to update the file.

General command structure

General communication packet structure is as in table below. It consists of CMD_ID (2 bytes), Data length of command and payload.

Command ID	Data length	Data
2 bytes	2 bytes	[data length] bytes

Modified file transfer protocol

Initialization packet On connection, the device sends an initialization packet.

Header (0x0000)	Protocol ID	IMEI	Settings
2 bytes	2 bytes	8 bytes	4 bytes

Protocol ID – just a reference for the protocol version that is running on device (for server cross compatibility with older versions).

Settings flag contains information on what is available for download. Structure provided:

Settings, 4 B			
Byte 3	Byte 2	Byte 1	Byte 0
*			

Close session command (CMD_ID 0x0000) In case when device connects to the server, but server does not expect it to connect, server will respond by sending CLOSE command after which connection will be terminated. This command is also used then device connect to server for custom file sending and server finishes to send all custom files to device.

Command ID	Data length
0x0000	0x0000

Request file path (CMD_ID 0x000C)

First after connecting to server and seeing bit set in init packet the server should send file path request command.

Command ID	Data length	Blank field (2 bytes)
0x000C	0x0002	0x0000 (always same value)

File path response (CMD_ID 0x000D)

This is a response to file path request (0x000C) command.

Command ID	Data length	Blank field (Variable)
0x000D	1-512	<file path, or \0>

File request command (CMD_ID 0x0008)

After device is connected for file upload server initiates file transfer by sending FILE REQ command.

Command ID	Data length	File Identifier
0x0008	2 bytes	File path has to be requested first via 0x000C command.

Device should answer with START command described above indicating size and CRC of requested file.

Start file transfer command (CMD_ID 0x0001)

After device received file request command from server (0x0008) device sends START command with file data (file size in bytes).

Command ID	Data length	File Size (4 bytes)	Blank field (2 bytes)
0x0001	0x0006	0x12345678	0x0000 (always same value)

Resume file transfer command (CMD_ID 0x0002)

In a response to the START command a RESUME command must be sent from server.

Command ID	Data length	Packet offset (4 bytes)
0x0002	0x0004	0x00000000

To begin file transfer from the start, offset should be set to 0 (4 bytes value). In case when the file transfer was interrupted, to resume file transfer, offset can be set to the desired value (1 ≤ [offset] ≤ [file packets]).

Synchronize file transfer command (CMD_ID 0x0003)

In a response to the RESUME command SYNC command is sent from device.

Command ID	Data length	File offset (4 bytes)
0x0003	0x0004	0x00000000

By sending SYNC command it is ensured that next data command will contain file data starting from the specified offset.

File data transfer command (CMD_ID 0x0004)

After sending SYNC command file data transfer is started by sending DATA commands.

Command ID	Data length	File data (up to 1024 bytes)	Data CRC (2 bytes)
0x0004	0x0402	...	...

File data is split into 1024-byte parts, each part wrapped into DATA command and sent.

Note: if command with a bad CRC is received, RESUME command should be sent with last valid file offset, after receiving RESUME command, server will stop sending DATA commands and continue communication from "Resume file transfer command (CMD_ID 0x0002)" step.

CRC polynomial expression: ***0x8408***

Initial value, when calculating CRC, is previously received packet (CMD ID 0x0004) CRC value.

File transfer status command (CMD_ID 0x0005)

After file transfer is completed and no more files are required from device, server should send COMPLETED command to the device (this command doesn't work after executing repeat init command 0x0009 – in this case the server should send close session 0x0000 command mentioned before).

Command ID	Data length	Status (4 bytes)
0x0005	0x0004	0x00000000

In case of server using invalid arguments, commands or not following the file request flow, the device will send this command with Status field set to one of the few possible error codes. List of possible ones provided below.

Status value (hexadecimal)	Description	Notes
0x00000000	File transfer process completed	Sent from server
0x00000002	Failed to close GPRS	Sent from device
0x00000003	Failed to close socket	Sent from device
0x00000005	Invalid response from server to init packet	Sent from device
0x00000011	This error code forces the device to disconnect from server	Sent from device. Possible causes: · „MDAS-9" special ID (1039) is not active · The requested file is not available by camera

After COMPLETED command device should disconnect from the server.

Initialization packet repeat command (CMD_ID 0x0009)

When sent, the initialization packet is repeated. This is used, when all of the files are downloaded and additional check is carried out for any additional files, that may have been captured during the download operation.

File transfer visual flow