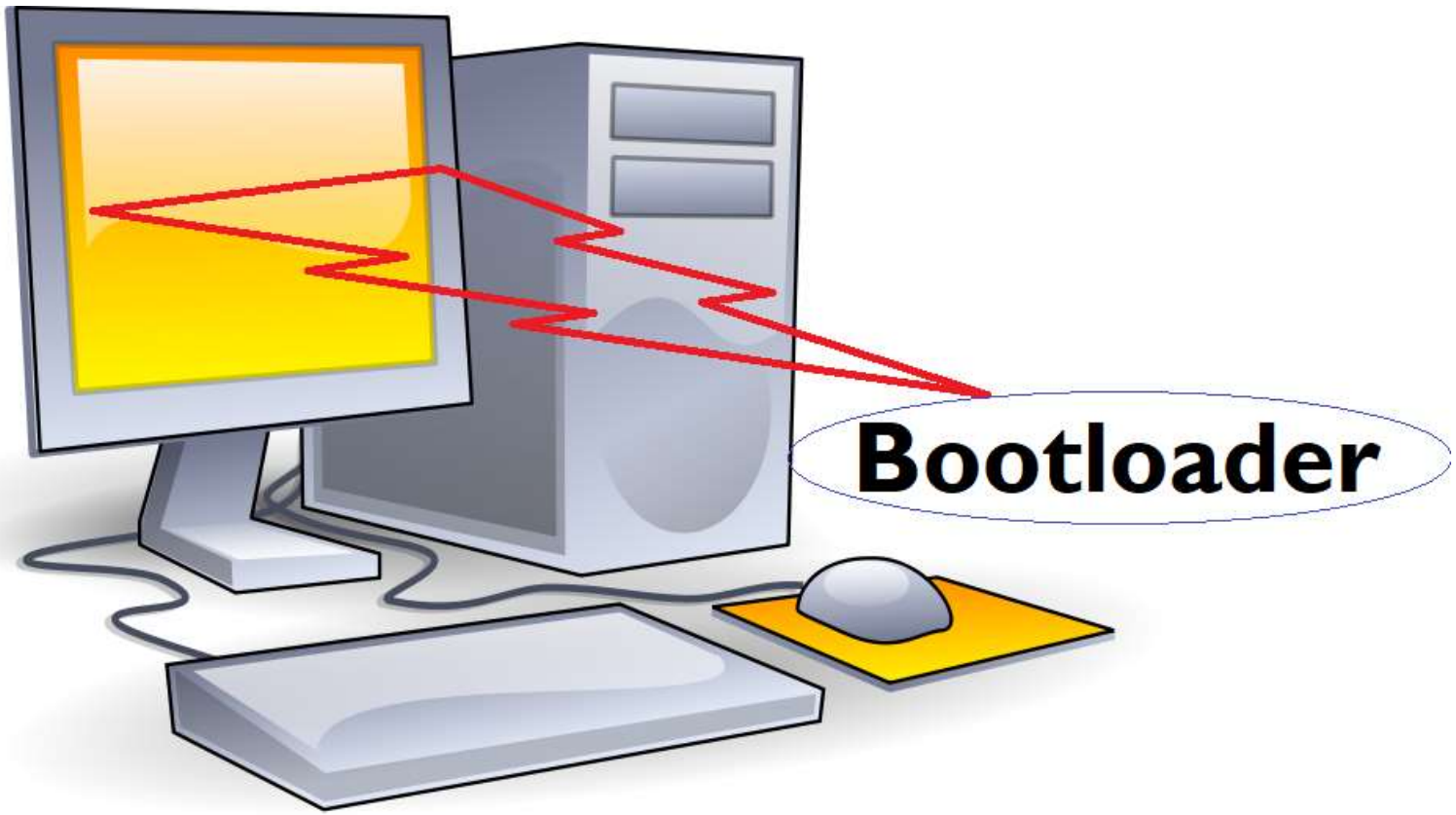
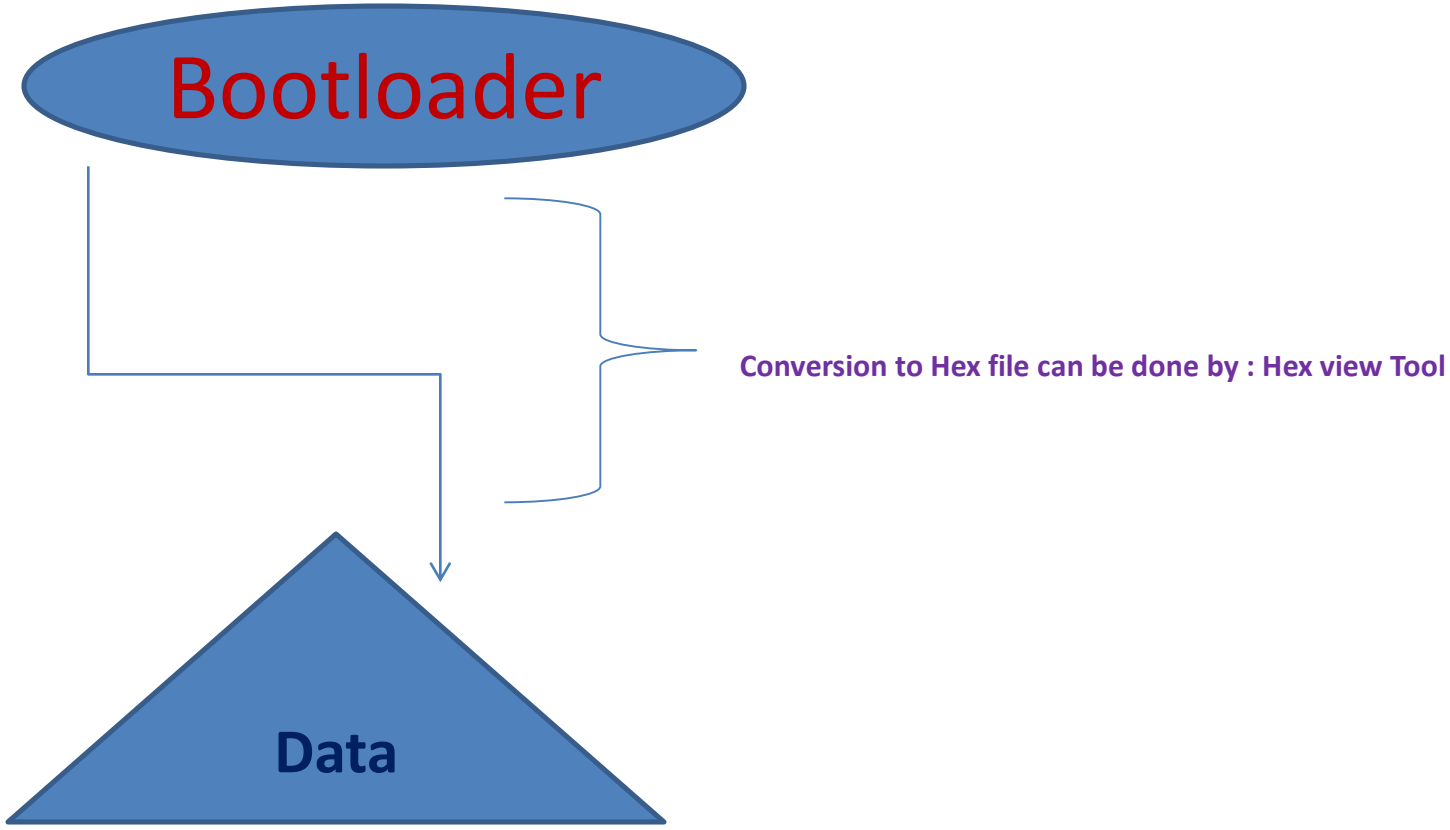
Request Upload

- The Request Upload service is used by the tester to initiate **the data** transfer from the ECU to Tester
- The above mode of transferring data is called Upload
- Once the ECU receives the Request Upload message request then the ECU should expect process to receive data before it sends a positive response message.

Assumption Scenario




Bootloader
Conversion to Hex file can be done by : Hex view Tool
Data

Hex File

01 4D 13 DE 01 21 01 10 10 01 01 .. … ……

02 4D 12 DE 01 41 01 10 10 01 01 .. … ……

03 4D 15 DE 01 21 01 05 10 01 01 .. … ……

04 4D 17 DE 01 27 01 10 10 01 01 .. … ……

05 4D 18 DE 01 61 01 10 10 01 01 .. … ……

06 4D 12 DE 01 21 01 10 10 01 01 .. … ……

07 4D 1B DE 01 81 01 10 10 01 01 .. … ……

Terminologies – **Upload Download Functional unit**

Data Format Identifier

- Data format Identifier is a Data parameter which is represented as one-byte value with each nibble encoded separately. The high nibble specifies the Compression Method & the low nibble specifies the Encrypting Method.
- The value 00 hex specifies that no Compression Method nor Encrypting Method is used.
- Values other than 00 hex are vehicle-manufacturer-specific.

Terminologies – **Upload Download Functional unit**

Address and Length Format Identifier

- The number of bytes used for the Memory Address and Memory Size parameters is defined by Address and Length Format Identifier (low and high nibble).
- It is also possible to use a fixed Address and Length Format Identifier & **unused bytes** within the Memory Address or Memory Size parameter are padded with the value 00 hex in the higher range address locations.
  - **Bit 7-4 : length of memory size parameter**
  - **Bit 3-0 : length of memory address parameter**

Terminologies – **Upload Download Functional unit**

Memory Size (Un Compressed)

- During the Transfer Data service the size of memory used by the server is compared with amount of data transferred & uncompressed memory size.
- This increases the programming security.
- The number of bytes used for this size is defined by the high nibble (bit 7 - 4) of the Address Format Identifier.

Terminologies – **Upload Download Functional unit**

Memory Address

- The parameter Memory address is the **starting address** of server memory to which data is to be written.
- After recognizing the starting address server will keep on transfer a data to consecutive address till the end byte of data
- The number of bytes used for this address is defined by the low nibble (bit 3 - 0) of the **Address Format Identifier.**
- Byte #m in the Memory Address parameter is always the least significant byte of the address being referenced in the server.
- The most significant byte of the address can be used as a memory Identifier.

Terminologies – **Upload Download Functional unit**

Length Format Identifier

- This parameter is a one-byte value with each nibble encoded separately:
  - bit 7 - 4: length (number of bytes) of the Max Number Of Block Length parameter.
  - bit 3 - 0: reserved by standard to be set to 0 hex.
- The format of this parameter is compatible to the format of the address and Length format Identifier parameter contained in the request message, except that the lower nibble has to be set to 0 hex.

Max Number of Block Length

- This parameter is used by the Request Download positive response message to inform the client how many data bytes shall be included in each Transfer Data request message from the client.
- This length reflects the complete message length.

Terminologies – **Upload Download Functional unit**

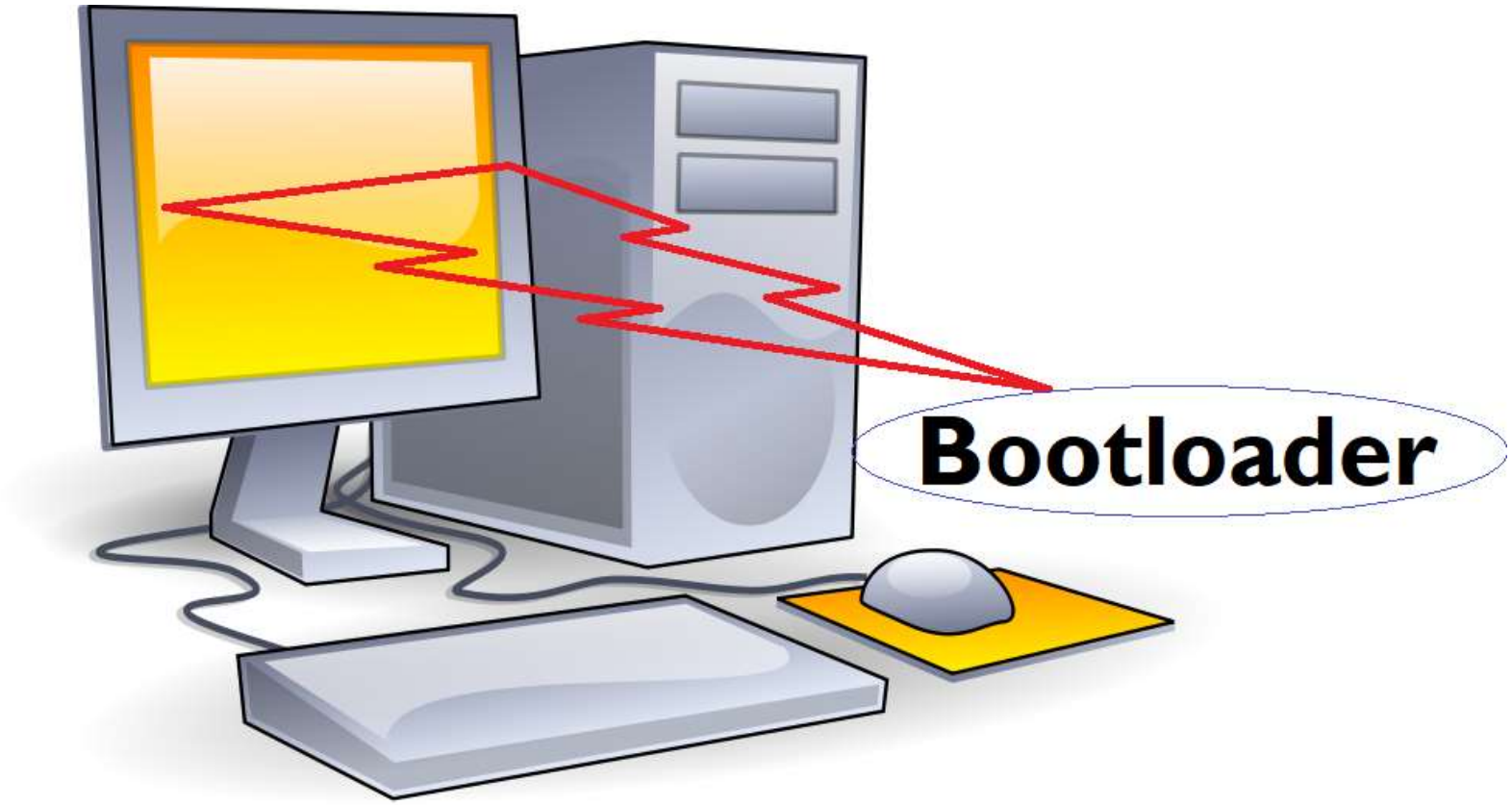
Block Sequence Counter

- The Block Sequence Counter parameter value starts at 01 hex with the first Transfer Data request that follows the
- Request Download (34 hex) or Request Upload (35 hex) service. Its value is incremented by one for each subsequent
- Transfer Data request. At the value one-byte value with each nibble encoded separately

Checksum

- To compare the uncompressed data and compressed data check sum is used at the end.
- Once the process gets started check sum will note down the end value address and which compare before the entire data transfer done.
- If the check sum matches then it is considered as data transferred successfully

Request Upload



Request Upload

- The Request Upload service is used by the tester to initiate **the data** transfer from the ECU to Tester
- The above mode of transferring data is called Upload
- Once the ECU receives the Request Upload message request then the ECU should expect process to receive data before it sends a positive response message.

Positive Response

34 – Service Id
00 – Data Format Id
11 - Address & Length format Id
20 - Memory Address
01 - Memory Size

04 - PCI Length
74 - Response Sid
30 - Length Format Id
02 10 11 - Max Number of block length

List of NRCs Supported – 0x22

- 0x13 Incorrect Message Length
- 0x22 Conditions Not Correct
- 0x31 Request Out of Range
- 0x33 Security Access Denied
- 0x70 Upload Download Not Accepted

This NRC shall be returned if:

- **Data Format Identifier is not valid**
- **Mode of Operation is not valid**
- **File Size Parameter Length is not valid**
- **File Path And Name Length is not valid**
- **File Size Uncompressed is not valid**
- **File Size Compressed is not valid**
- **File Path And Name is not valid**