**Modified Chrysler CCD/SCI Scanner UART protocol for SBHACK**

**Last update: 2020.02.27**

**1. Frame format**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Start of frame | Data length | | Data description | | Data | | End of frame |
| **SYNC** | **LENGTH HB** | **LENGTH LB** | **DATA CODE** | **SUB-DATA CODE** | | **PAYLOAD** | **CHECKSUM** |
| $3D | $00 | $06 | $05 | $01 | | $00 $3D $14 $B0 | $0D |

Table 1. Frame format.

1.1 **SYNC** byte: fixed value at the beginning of every message ($3D).

1.2 **LENGTH** bytes: number of bytes following (not including these two) until CHECKSUM byte is reached. Maximum frame length is limited to 1024 bytes. Size: 2 bytes.

1.3 **DATA CODE** byte: describes the source, target and command. Size: 1 byte.

1.4 **SUB-DATA CODE** byte: command extension if the command alone is not enough to describe the purpose of the message. Size: 1 byte

1.5 **PAYLOAD** byte(s): optional. Arbitrary data can be stored here.  
Size: limited to a maximum of 1024-6=1018 bytes.

1.6 **CHECKSUM** byte: all bytes, except SYNC, summed up, lower byte of the result is placed here for error detection. Size: 1 byte.

**2. Frame bytes in detail**

**2.1 DATA CODE byte description**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| DATA CODE byte | | | | | | | |
| Source | Target | | | Command | | | |
| **bit 7** | **bit 6** | **bit 5** | **bit 4** | **bit 3** | **bit 2** | **bit 1** | **bit 0** |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |

Table 2. DATA CODE byte description.

Bit 7 source of the frame:

- 1b0 ($00): COMPUTER

- 1b1 ($80): ARDUINO

Bit 6:4 | target of the frame:

- ALWAYS 3b000

Bit 3:0 | command

- 4b0000 ($00): Reset

- 4b0001 ($01): Handshake

- 4b0010 ($02): Status

- 4b0011 ($03): Settings

- 4b0100 ($04): Read data

- 4b0101 ($05): Write data

- 4b0110-4b1101 ($06-$0E): RESERVED

- 4b1111 ($0F): OK/Error

**2.2 SUB-DATA CODE byte description**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| SUB-DATA CODE byte | | | | | | | |
| **bit 7** | **bit 6** | **bit 5** | **bit 4** | **bit 3** | **bit 2** | **bit 1** | **bit 0** |
| 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |

Table 3. SUB-DATA CODE byte description.

Individual bits have no special meaning. Different commands have different SUB-DATA CODE bytes.

Related command: $00 (Reset)

- $00-$FF: RESERVED

Related command: $01 (Handshake)

- $00: Request scanner to send handshake over USB connection

- $01: Send handshake, hardware/firmware information and status too

- $02-$FF: RESERVED

Related command: $02 (Status)

- $00 RESERVED

- $01 Timestamp

- $02 Scan smbus

- $03 Smbus dump

- $04-$FF: RESERVED

Related command: $03 (Settings)

- $00: RESERVED

- $01: Current Settings

- $02: Set smbus address

- $03: Set R/W byte order

- $04-$FF: RESERVED

Related command: $04 (Read Data)

- $00: RESERVED

- $01: Read byte

- $02: Read word

- $03: Read block

- $04: Read ROM byte

- $05: Read ROM block

- $06-$FF: RESERVED

Related command: $05 (Write Data)

- $00: RESERVED

- $01: Write byte

- $02: Write word

- $03: Write block

- $04-$FF: RESERVED

Related command: $0F (OK/ERROR)

- $00: OK: general acknowledgement

- $01: ERROR: LENGTH, invalid value

- $02: ERROR: DATA CODE, invalid command

- $03: ERROR: SUB-DATA CODE, invalid value

- $04: ERROR: PAYLOAD, invalid value

- $05: ERROR: CHECKSUM, invalid value

- $06: ERROR: PACKET, timeout occurred

- $07-$FD: RESERVED

- $FE: ERROR: internal error

- $FF: ERROR: FATAL

**3. Example packets**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYNC** | **LENGTH HB** | **LENGTH LB** | **DATA CODE** | **SUB-DATA CODE** | **~~PAYLOAD~~** | **CHECKSUM** |
| $3D | $00 | $02 | $00 | $00 |  | $02 |

Table 4. Reboot scanner.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYNC** | **LENGTH HB** | **LENGTH LB** | **DATA CODE** | **SUB-DATA CODE** | **~~PAYLOAD~~** | **CHECKSUM** |
| $3D | $00 | $02 | $01  0:TX  1:handshake | $00 |  | $03 |

Table 5. Handshake request.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYNC** | **LENGTH HB** | **LENGTH LB** | **DATA CODE** | **SUB-DATA CODE** | **PAYLOAD** | **CHECKSUM** |
| $3D | $00 | $08 | $81  8:RX  1:handshake | $00 | $53 $42 $48 $41 $43 $4B | $35 |

Table 6. Handshake response.

In the PAYLOAD section the scanner responds with an ASCII encoded text:

$53 $42 $48 $41 $43 $4B = SBHACK

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYNC** | **LENGTH HB** | **LENGTH LB** | **DATA CODE** | **SUB-DATA CODE** | **PAYLOAD** | **CHECKSUM** |
| $3D | $00 | $03 | $04  0:TX  4:Read | $03  3:Block | $20  Reg 20 | $2A |

Table 7. Read block reg 0x20.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYNC** | **LENGTH HB** | **LENGTH LB** | **DATA CODE** | **SUB-DATA CODE** | **PAYLOAD** | **CHECKSUM** |
| $3D | $00 | $07 | $84  8:RX  4:Read | $03  3:Block | $20 $45 $43 $53  Reg 20  ESC | $8C |

Table 8. Echo of Read block reg 0x20.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYNC** | **LENGTH HB** | **LENGTH LB** | **DATA CODE** | **SUB-DATA CODE** | **PAYLOAD** | **CHECKSUM** |
| $3D | $00 | $05 | $05  TX,write | $02  word | $00 $00 $01  Reg 00,  data0001 | $0D |

Table 9. Write word 0001 to reg 0x00.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYNC** | **LENGTH HB** | **LENGTH LB** | **DATA CODE** | **SUB-DATA CODE** | **PAYLOAD** | **CHECKSUM** |
| $3D | $00 | $06 | $85  RX,write | $02  word | $00 $00 $01 $02  Reg 00,Data 0001,02 bytes written | $90 |

Table 10. Echo write word 0001 to reg 0x00