## Socket Command Responses

|  |  |
| --- | --- |
| **Mode** | **Response** |
| TCP Server | * Server start   Success: “CONNECT: <SOC ID>” followed by OK  Failure: ERROR   * After accepting new connection   “CONNECTED:<SERVER SOC ID>:  <CLIENT SOC ID>:<CLIENT IP ADDR>” followed by OK   * Data Reception   “DATA:<SOC ID>:<DATA LENGTH>:<DATA>”   * For Connection Closure   “CLOSE:<SOC ID>” |
| UDP Server | * Server start   Success: “CONNECT: <SOC ID>” followed by OK  Failure: ERROR   * Data Reception   DATA:<SOC ID>:<IP ADDRESS>:<REMOTE PORT>:  <DATA LENGTH>:<DATA>”   * For Connection Closure   “CLOSE:<SOC ID>” |
| TCP Client | * Client starts   Success: “CONNECTED:<SOC ID>” followed by OK  Failure: ERROR   * Data Reception   “DATA:<SOC ID>:<DATA LENGTH>:<DATA>”   * For Connection Closure   “CLOSE:<SOC ID>” |
| UDP Client | * Client starts   Success: “CONNECTED:<SOC ID>” followed by OK  Failure: ERROR   * Data Reception   DATA:<SOC ID>:<IP ADDRESS>:<REMOTE PORT>:  <DATA LENGTH>:<DATA>   * For Connection Closure   “CLOSE:<SOC ID>” |
| HTTP Client | * HTTP Client start   Success: “CONNECTED:<HC ID>” followed by OK  Failure: ERROR   * For Data Reception   “HDATA:<HC ID>:<Status Code>:<Data len>:<DATA>”   * For Connection Closure   “CLOSE:<HC ID>” |
| Ping | * Ping Success:   “Reply from:<IP Address>: bytes:  <No Bytes>:time:<Time>:ms” Followed by OK   * Ping Failure:   “Request timed out.  “ Followed by “OK” |
| Nhostipget | * Success   “IPv<family>:<ip address>” Followed by OK  Family can be 6/4.   * Failure:   ERROR |
| Socket Close | “CLOSE:<SOC ID>” |

Table 1: Socket Command Responses

## Command Response IDs

The following table lists the command response IDs and their descriptions:

|  |  |
| --- | --- |
| **Command Response ID** | **Description** |
| 0x0000 | AT Command Success |
| 0x0001 | AT Command Error |
| 0x0002 | AT Command Invalid |
| 0x0003 | WLAN Disconnected |
| 0x0004 | WLAN Scan Results |
| 0x0005 | WLAN Status id – 0 |
| 0x0006 | WLAN Status id -1 |
| 0x0007 | WLAN Status id -2 |
| 0x0008 | WLAN Status id -3 |
| 0x000C | Socket - Client connected to Server |
| 0x000D | Socket - Server Started |
| 0x000E | Socket – Server accepted the connection from Client |
| 0x000F | Socket Close |
| 0x0011 | TCP Data Received |
| 0x0012 | MDNS Service Registered |
| 0x0013 | HTTP Data Received |
| 0x0014 | Received Hostname Resolution |
| 0x0015 | Ping Result |
| 0x0016 | Wakeup event |
| 0x0017 | BLE Connected |
| 0x0018 | BLE Disconnected |
| 0x0019 | BLE Scan |
| 0x001C | BLE Primary Service |
| 0x001D | BLE included Service |
| 0x001E | BLE Characteristic |
| 0x001F | BLE Characteristic Descriptor |
| 0x0020 | BLE Characteristic Data |
| 0x0021 | WLAN regulatory Domain |
| 0x0022 | Software Version |
| 0x0023 | NTP Time |
| 0x0024 | MQTT Connect |
| 0x0025 | MQTT Subscribe |
| 0x0026 | BLE Bondlist |

Table 2: Command Response IDs

## BLE Asynchronous Response IDs

The following table lists the asynchronous response IDs and their descriptions:

|  |  |
| --- | --- |
| **ASYNC Response ID** | **Description** |
| 0x8000 | BLE characteristic read request. This notification will be in following format:  <uuidlen>:<uuid>:<offset>:<len>   * uuidlen: UUID length * uuid: characteristic UUID * offset: offset location to read * len: number of bytes to read |
| 0x8001 | BLE characteristic write request. This notification will be in following format  <uuidlen>:<uuid>:<offset>:<len>:<data>   * uuidlen: UUID length * uuid: characteristic UUID * offset: offset location to write * len: number of bytes to write * data: actual data to write |
| 0x8002 | BLE notify the pass key. This notification send passkey in following format.  “passkey”:<passkey>   * passkey: passkey to enter in the remote device |
| 0x8003 | BLE authentication status  “bleauth”:<status>  Status:   * 0 : indicates success * 1 : for failure, invalid passkey * 2 : for failure, oob not available * 3 : for failure, authentication requirements * 4 : for failure, confirm value failed * 5 : for failure, pairing not supported * 6 : for failure, encryption key size mismatch * 7 : for failure, command not supported * 8 : for failure, unspecified reason * 9 : for failure, repeated attempts * a : for failure, invalid parameters * b : for failure, dhkey check failed * c : for failure, numeric comparison failed * d : for failure, pairing in progress * e : for failure, cross transport key derivation error * 10 : for failure, internal timeout * 11 : for failure, remote lost bond |
| 0x8004 | BLE descriptor read request. This notification will be in following format  <uuidlen>:<uuid>:<offset>:<len>   * uuidlen: UUID length * uuid: descriptor UUID * offset: offset location to read * len: number of bytes to read |
| 0x8005 | BLE descriptor write request. This notification will be in following format  <uuidlen>:<uuid>:<offset>:<len>:<data>   * uuidlen: UUID length * uuid: descriptor UUID * offset: offset location to write * len: number of bytes to write * data: actual data to write |
| 0x8006 | Crash notification will be in the following format:  : ASSERTED |
| 0x8007 | BLE notification/indication notification will be in the following format:  <handle\_id>:<len>:<data>   * handle id: Characteristic handle ID * len: Length of data * data: Actual data |

Table 3: BLE Asynchronous Response IDs