## Senior Engineer Development Test

Our devices communicate via a simple Type/Length/Value protocol received in a single UDP packet. The focus of this task is to produce a decoder that takes a byte array as input and returns the decode bytes in a suitable in memory representation of your design. The decoded representation should allow known 'type' values to be recognised and interpreted as described in the specification below.

* The exercise must be completed using C#
* There is no time limit for this test however I would expect it to take no longer than 2 hours
* Your solution must compile and run first time
* You may add unit/integration tests using a test framework (and/or mocking framework) of your choice

**Note:** The specification given may change independently of the consuming applications due, for example, to device firmware changes. As such, any decoders must be resilient to receiving unknown types, corrupted, incomplete or invalid data. For this exercise these can be handled in whichever way you feel is appropriate.

Example Payload:

HEX representation: 01024B0802064D2D4B4F504103030102030401020510FEFFFFFF2D000D001800000062005A00

Decimal byte array: { 1, 2, 75, 8, 2, 6, 77, 45, 75, 79, 80, 65, 3, 3, 1, 2, 3, 4, 1, 2, 5, 16, 254, 255, 255, 255, 45, 0, 13, 0, 24, 0, 0, 0, 98, 0, 90, 0 }

The payload explained:

HEX21

|  |  |  |
| --- | --- | --- |
| **Type** | **Length** | **Value bytes** |
| 0x01 | 0x02 | 0x4B, 0x08 |
| 0x02 | 0x06 | 0x4D, 0x2D, 0x4B, 0x4F, 0x50, 0x41 |
| 0x03 | 0x03 | 0x01, 0x02, 0x03 |
| 0x04 | 0x01 | 0x02 |
| 0x05 | 0x10 | 0xFE, 0xFF, 0xFF, 0xFF, 0x2D, 0x00, 0x0D, 0x00, 0x18, 0x00, 0x00, 0x00, 0x62, 0x00, 0x5A, 0x00 |

Decimal

|  |  |  |
| --- | --- | --- |
| **Type** | **Length** | **Value bytes** |
| 1 | 2 | 75, 8 |
| 2 | 6 | 77, 45, 75, 79, 80, 65 |
| 3 | 3 | 1, 2, 3 |
| 4 | 1 | 2 |
| 5 | 16 | 254, 255, 255, 255, 45, 0, 13, 0, 24, 0, 0, 0, 98, 0, 90, 0 |

Known type interpretations:

**NOTE:** All values are transferred using Little Endian byte order.

|  |  |  |  |
| --- | --- | --- | --- |
| **Type byte** | **Expected length** | **Type Details** | **Value in example payload** |
| 1 (0x01) | 2 | Unsigned 16 bit integer | 2123 |
| 2 (0x02) | variable | ASCII encoded string | M-KOPA |
| 3 (0x03) | 3 | Software version in the form [ Major, Minor, Revision ] | 1.2.3 |
| 4 (0x04) | 1 | Power consumption enumeration. Possible values are  {  0: "Auto Mode",  1: "Low Consumption",  2: "Normal Consumption",  3: "High Consumption"  } | Normal Consumption |
| 5 (0x05) | variable | Device telemetry observation array  [  {  "Temperature":<signed 32 bit integer>,  "BatteryLevel":<unsigned 16 bit integer>,  "SolarVoltage":<unsigned 16 bit integer>,  }  ] | [  { "Temperature": -2, "BatteryLevel": 45, "SolarVoltage":13 },  { "Temperature": 24, "BatteryLevel": 98, "SolarVoltage":90 }  ] |