```
function decodeUplink(input) {

    var decoded = {};
    var bytes = input.bytes;

    if (input.fPort == 1) {//TELEMETRY

        //decode header
        decoded.base_id = bytes[0] >> 4;
        decoded.major_version = bytes[0] & 0x0F;
        decoded.minor_version = bytes[1] >> 4;
        decoded.product_version = bytes[1] & 0x0F;
        decoded.up_cnt = bytes[2];
        decoded.battery_voltage = ((bytes[3] << 8) | bytes[4]) / 1000.0;
        decoded.internal_temperature = ((bytes[5] << 8) | bytes[6]) / 10 -
100;
        decoded.networkBaseType = 'lorawan';
        decoded.networkSubType = 'tti';

        var it = 7;

        if(decoded.minor_version >= 3){
            it = 7;

            //Luftfeuchte ist bei allen Varianten enthalten
            decoded.humidity = bytes[it++];

            if (decoded.product_version & 0x01) { // Co2 und Druck sind
enthalten wenn subversion bit0 = 1, andernfalls 0
                decoded.pressure = (bytes[it++] << 8 | bytes[it++]);
                decoded.co2_ppm = (bytes[it++] << 8 | bytes[it++]);
            } else {
                it += 4;//Werte sind 0  aus kompatibilitäts Gründen, daher
überspringen
```

```
            }

            decoded.alarm = bytes[it++];//Alarm-Level, entspricht grün, gelb,
rot

            //FIFO Werte wegwerfen (1 byte fifo size, 1 byte period, 7 bytes
pro fifo eintrag)
            it += 2 + bytes[it] * 7;

            decoded.dew_point = ((bytes[it++] << 8) | bytes[it++]) / 10 -
100;

            // Wandtemperatur und Feuchte enthalten wenn subversion bit 2 = 1
            if (decoded.product_version & 0x04) {
                decoded.wall_temperature = ((bytes[it++] << 8) | bytes[it++])
/ 10 - 100;
                decoded.therm_temperature = ((bytes[it++] << 8) |
bytes[it++]) / 10 - 100;
                decoded.wall_humidity = bytes[it++];
            }

        }else{
            it = 7;

            //Luftfeuchte ist bei allen Varianten enthalten
            decoded.humidity = bytes[it++];

            if (decoded.product_version & 0x01) { // Co2 und Druck sind
enthalten wenn subversion bit0 = 1, andernfalls 0
                decoded.pressure = (bytes[it++] << 8 | bytes[it++]);
                decoded.co2_ppm = (bytes[it++] << 8 | bytes[it++]);
            } else {
                it += 4;//Werte sind 0  aus kompatibilitäts Gründen, daher
überspringen
            }
```

```
            decoded.alarm = bytes[it++];//Alarm-Level, entspricht grün, gelb,
rot


            //FIFO Werte wegwerfen (1 byte fifo size, 1 byte period, 7 bytes
pro fifo eintrag)
            it += 2 + bytes[it] * 7;


            //Taupunkt seit minor version 2 bei alle Varianten enthalten
(ausnahme früher versionen subversion 2, daher byte prüfen)
            if (decoded.minor_version >= 2 && bytes[it] ) {


                decoded.dew_point = bytes[it++] - 100;
            }


            // Wandtemperatur und Feuchte enthalten wenn subversion bit 2 = 1
            if (decoded.product_version & 0x04) {
                decoded.wall_temperature = bytes[it++] - 100;
                decoded.therm_temperature = bytes[it++] - 100;
                decoded.wall_humidity = bytes[it++];


            }



        }
    }


    return {
        data: decoded,
        warnings: [],
        errors: []
    };


}
```