

```

/*
*****
This code uses include files and parts of the example
provided by Andrew Davies of Hobby Components LTD.
I acknowledge the assistance of HCMP180_Example from
their website.
*****/
char Version[14] = "CkptVlt1.6"; // This is the first correct working
prototype. To be evalated
/*****
This is clean up version dated 09/07/2025
*****

MPX Data Types
0  Special Data
1  Voltage
2  Current
3  Variometer
4  Speed
5  Rev Counter
6  Temperature
7  Direction
8  Height / Short Distance
*/

//Analog Address
#include <SoftwareSerial.h>

int i = 0;
int Vlt = 0; //The voltage value
int Dum = 25; // Dum variable #1
int Datach0 = 2; //Data channel select bit0
int Datach1 = 3; //Data channel select bit1
char Address = 2; // Default MPX data channel address. Can be changed
char DataIn = 0;
char Descript = 1; // MPX data code for Voktage
char FirstByte = 0;
char SecondByte = 0;
char thirdByte = 0;
int val = 0;
float anaval = 0;
int analogPin = A7;
SoftwareSerial mySerial(10, 11); // RX, TX set up for Arduino

void setup()
{

// The next 5 lines setup the data channel:2, 3, 4 or 5
pinMode(Datach0, INPUT_PULLUP); //Sets I/O pin #2 to Input-Pullup
pinMode(Datach1, INPUT_PULLUP); //Sets I/O pin #3 to Input-Pullup
pinMode(analogPin, INPUT); //set Address A7 to high impedance input
if((digitalRead(Datach0) == 1) && (digitalRead(Datach1) == 0)) Address
= 3;
if((digitalRead(Datach0) == 0) && (digitalRead(Datach1) == 1)) Address
= 4;
if((digitalRead(Datach0) == 0) && (digitalRead(Datach1) == 0)) Address
= 5;
// Default data channel is 2
FirstByte = Address * 16; // Part 1 of FirstByte

```

```

FirstByte = FirstByte + Descript; // Part 2 of FirstByte
Serial.begin(38400); // Setup serial comms at 38400Baud

    Serial.println(VerSion); //Output Version in ASCII at 38400

}

void loop() // run over and over
{
    Vlt = val;

    if (Serial.available())
    {
        DataIn = Serial.read(); // Input data from MPX Receiver
        if(DataIn == Address) // Check for MPX data address
        {
            delay(1); // Wait till MPX Rx ready for sensor data
            // Now write previous Voltage input - delay of ~ 1/10 Seconds
            Serial.write(FirstByte); // This has data address and type
            Serial.write(SecondByte); // First 7 bits of Voltage + Alarm bit
            Serial.write(thirdByte); // Second 8 bits of Voltage

            if (Vlt >= 0) // Look for +Ve Voltage. (Slope use only)
            {
                Dum = Vlt & 127; // Trim Vlt to 7 bits
                SecondByte = Dum * 2; // and shift Left into SecondByte
                Dum = Vlt / 128; // Trim Voltage for next 8 bits
                thirdByte = Dum; // set thirdByte
            }

        }
    }

    val = analogRead(analogPin); // read the input pin
    // anaval = val;
    // anaval = anaval/2.05;
    // val = anaval;
    val = val/5.45; // This is the bodge factor to make the result match the
    battery
}

```