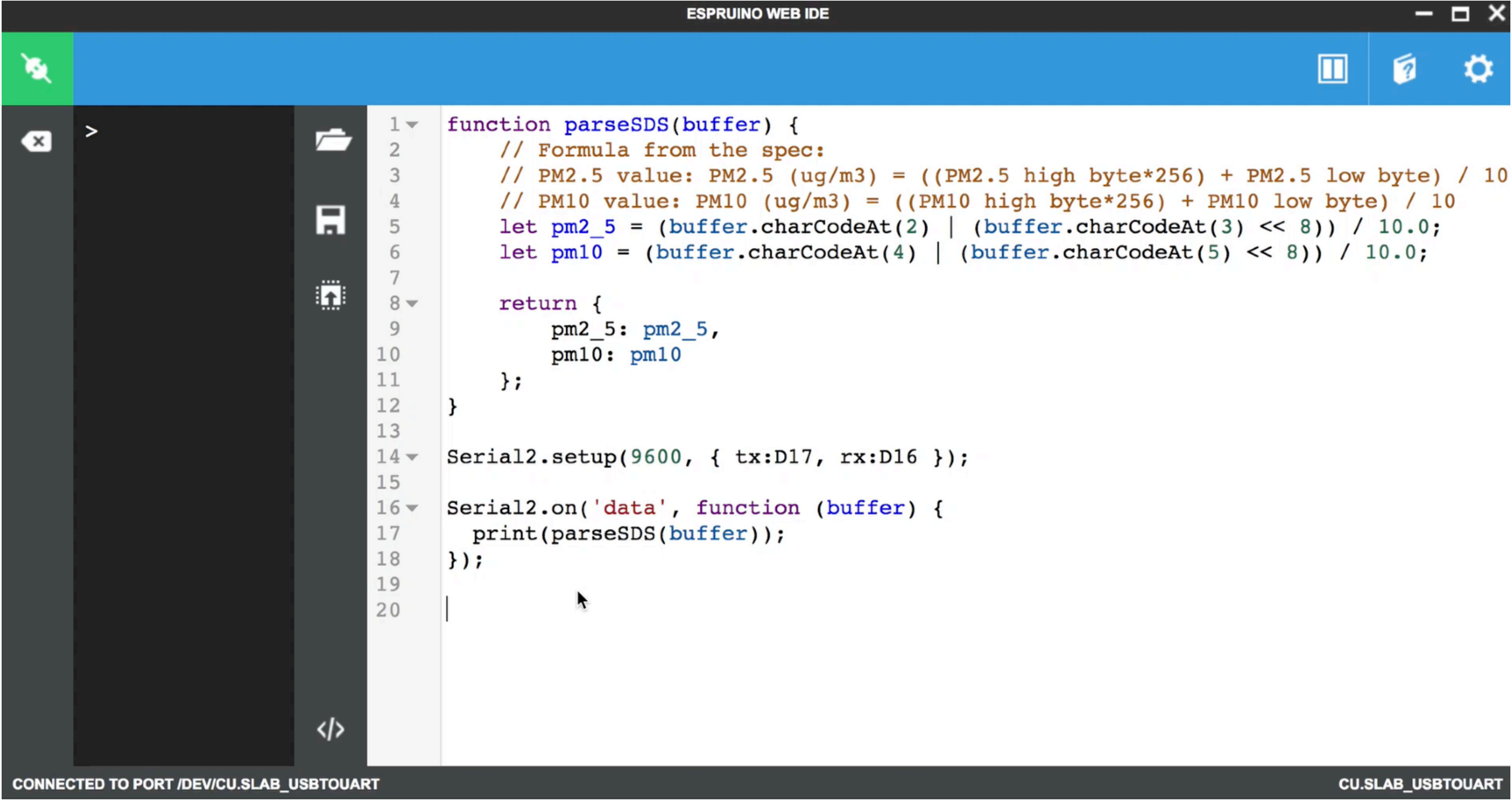
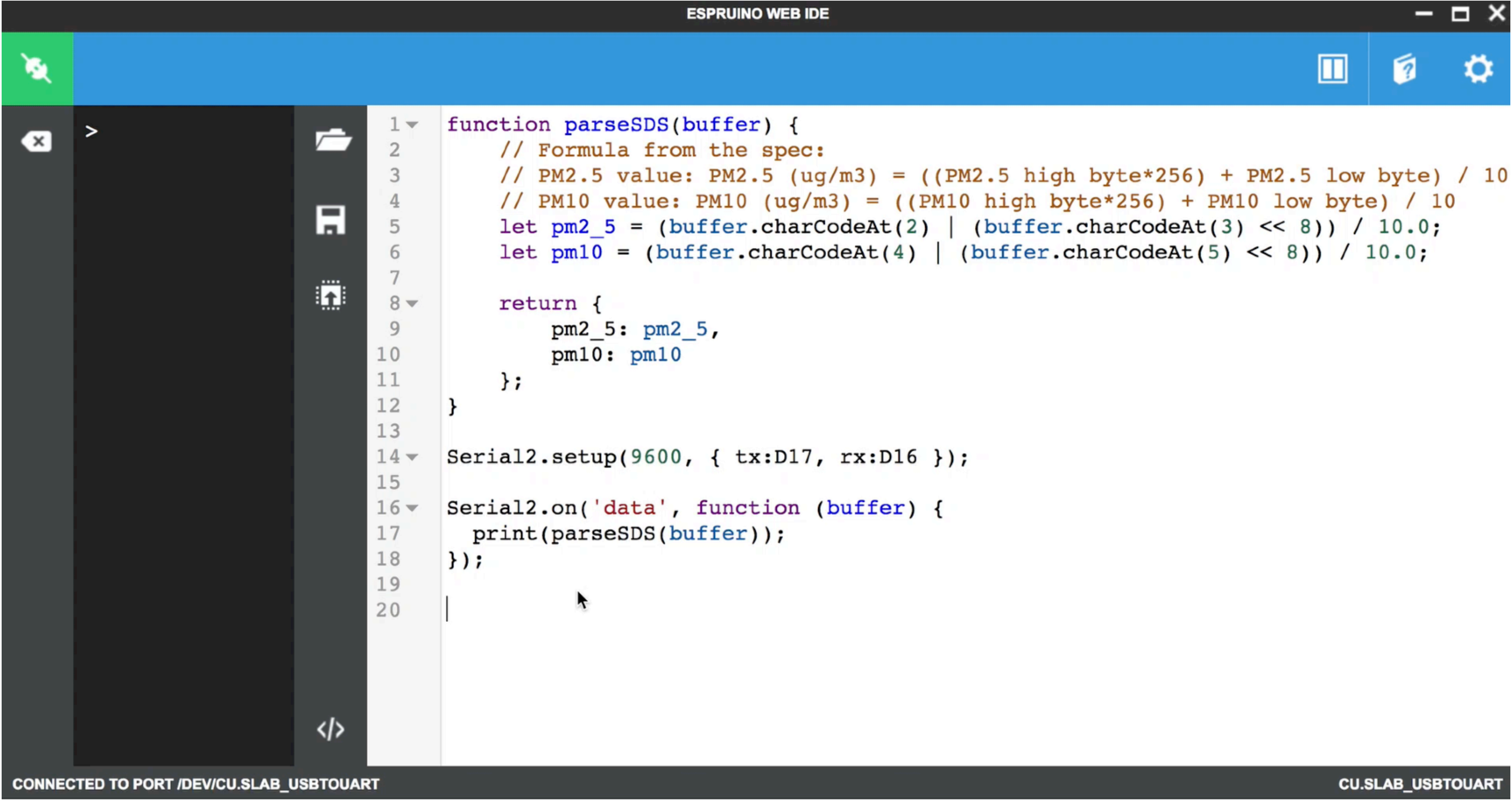
ESP32 Source Code


ESPRUINO WEB IDE
>
function parseSDS(buffer) {
// Formula from the spec:
// PM2.5 value: PM2.5 (ug/m3) = ((PM2.5 high byte*256) + PM2.5 low byte) / 10
// PM10 value: PM10 (ug/m3) = ((PM10 high byte*256) + PM10 low byte) / 10
let pm2_5 = (buffer.charCodeAt(2) | (buffer.charCodeAt(3) << 8)) / 10.0;
let pm10 = (buffer.charCodeAt(4) | (buffer.charCodeAt(5) << 8)) / 10.0;
return {
pm2_5: pm2_5,
pm10: pm10
};
}
Serial2.setup(9600, { tx:D17, rx:D16 });
Serial2.on('data', function (buffer) {
print(parseSDS(buffer));
});
CONNECTED TO PORT /DEV/CU.SLAB_USBTOUART
CU.SLAB_USBTOUART


ESPRUINO WEB IDE
>
function parseSDS(buffer) {
// Formula from the spec:
// PM2.5 value: PM2.5 (ug/m3) = ((PM2.5 high byte*256) + PM2.5 low byte) / 10
// PM10 value: PM10 (ug/m3) = ((PM10 high byte*256) + PM10 low byte) / 10
let pm2_5 = (buffer.charCodeAt(2) | (buffer.charCodeAt(3) << 8)) / 10.0;
let pm10 = (buffer.charCodeAt(4) | (buffer.charCodeAt(5) << 8)) / 10.0;
return {
pm2_5: pm2_5,
pm10: pm10
};
}
Serial2.setup(9600, { tx:D17, rx:D16 });
Serial2.on('data', function (buffer) {
print(parseSDS(buffer));
});
CONNECTED TO PORT /DEV/CU.SLAB_USBTOUART
CU.SLAB_USBTOUART

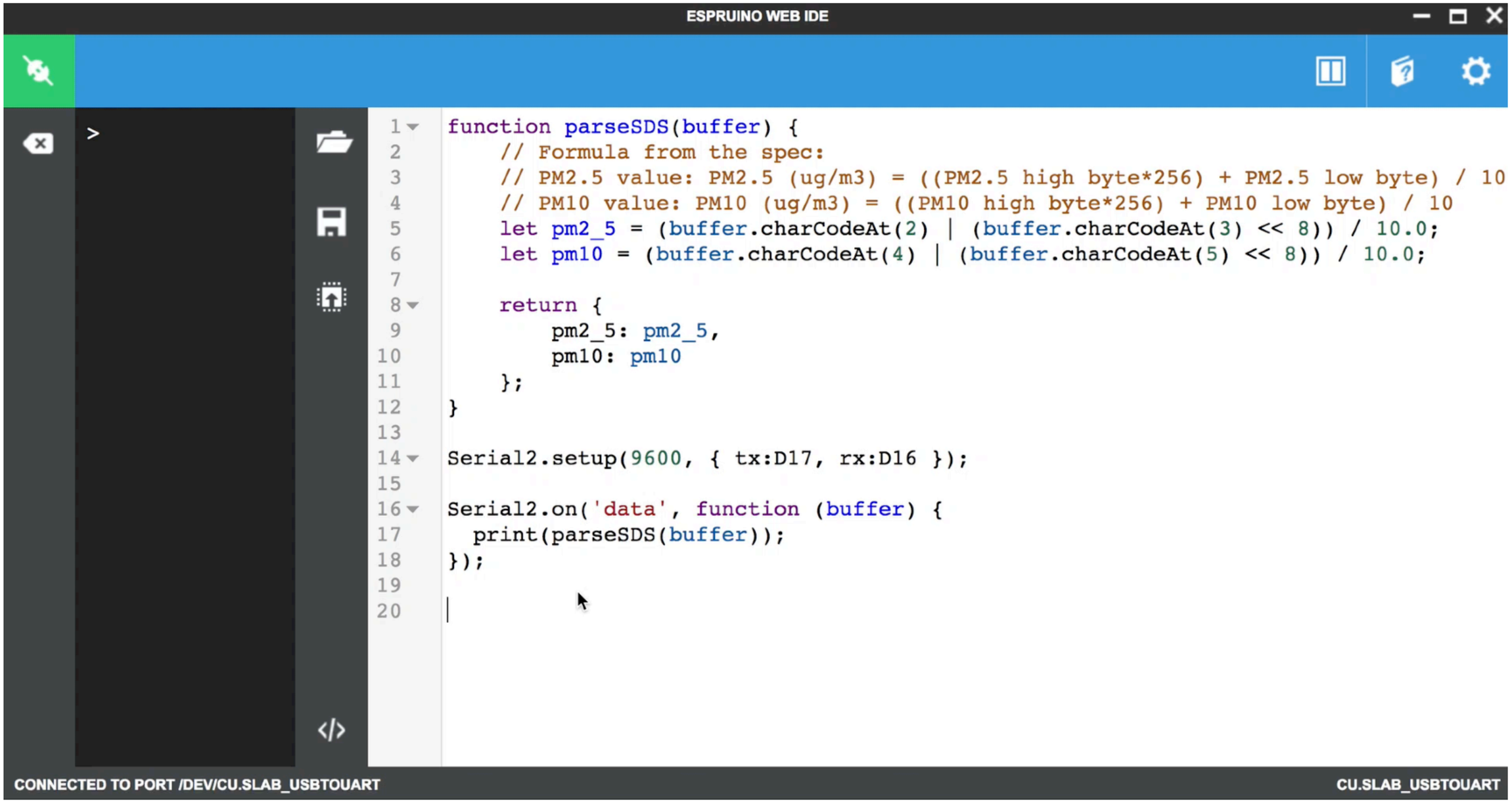
```
function parseSDS(buffer) {
    // Formula from the spec:
    // PM2.5 value: PM2.5 (ug/m3) = ((PM2.5 high byte*256) + PM2.5 low byte) / 10
    // PM10 value: PM10 (ug/m3) = ((PM10 high byte*256) + PM10 low byte) / 10
    let pm2_5 = (buffer.charCodeAt(2) | (buffer.charCodeAt(3) << 8)) / 10.0;
    let pm10 = (buffer.charCodeAt(4) | (buffer.charCodeAt(5) << 8)) / 10.0;

    return {
        pm2_5: pm2_5,
        pm10: pm10
    };
}

Serial2.setup(9600, { tx:D17, rx:D16 });

Serial2.on('data', function (buffer) {
  print(parseSDS(buffer));
});
```

ESP32 Source Code



Broadcast data with HTTP