

Decoder für LSN50 v2 (Temperatursensor)

```
function Decoder(fPort, bytes, variables) {
var data={
  //Work mode
  Work_mode:
  {
    "0": "IIC",
    "1": "Distance",
    "2": "3ADC",
    "3": "3DS18B20",
    "4": "Weight",
    "5": "Count"
  }[(bytes[6] & 0x7C)>>2],

  //Battery, units: V
  BatV:
  {
    "0": (bytes[0]<<8 | bytes[1])/1000,
    "1": (bytes[0]<<8 | bytes[1])/1000,
    "2": bytes[11]/10,
    "3": (bytes[0]<<8 | bytes[1])/1000,
    "4": (bytes[0]<<8 | bytes[1])/1000,
    "5": (bytes[0]<<8 | bytes[1])/1000,
  }[(bytes[6] & 0x7C)>>2],

  //DS18B20, PB3, units: 釐度?
  TempC1:
  {
    "0": ((bytes[2]<<24>>16 | bytes[3])/10).toFixed(2),
    "1": ((bytes[2]<<24>>16 | bytes[3])/10).toFixed(2),
    "3": ((bytes[2]<<24>>16 | bytes[3])/10).toFixed(2),
    "4": ((bytes[2]<<24>>16 | bytes[3])/10).toFixed(2),
    "5": ((bytes[2]<<24>>16 | bytes[3])/10).toFixed(2),
  }[(bytes[6] & 0x7C)>>2],

  //ADC Channel 0, PA0, units: V
  ADC_CH0V:

```

```

{
  "0":(bytes[4]<<8 | bytes[5])/1000,
  "1":(bytes[4]<<8 | bytes[5])/1000,
  "2": (bytes[0]<<8 | bytes[1])/1000,
  "3":(bytes[4]<<8 | bytes[5])/1000,
  "4":(bytes[4]<<8 | bytes[5])/1000,
  "5":(bytes[4]<<8 | bytes[5])/1000,
}[(bytes[6] & 0x7C)>>2],

//Digital Input Status,PA12
Digital_IStatus:
{
  "0":(bytes[6] & 0x02)? "H":"L",
  "1":(bytes[6] & 0x02)? "H":"L",
  "2":(bytes[6] & 0x02)? "H":"L",
  "3":(bytes[6] & 0x02)? "H":"L",
  "4":(bytes[6] & 0x02)? "H":"L",
  "5":(bytes[6] & 0x02)? "H":"L",
}[(bytes[6] & 0x7C)>>2],

//GPIO_MODE_IT_FALLING,PB14
EXTI_Trigger:
{
  "0":(bytes[6] & 0x01)? "TRUE":"FALSE",
  "1":(bytes[6] & 0x01)? "TRUE":"FALSE",
  "2":(bytes[6] & 0x01)? "TRUE":"FALSE",
  "3":(bytes[6] & 0x01)? "TRUE":"FALSE",
  "4":(bytes[6] & 0x01)? "TRUE":"FALSE",
}[(bytes[6] & 0x7C)>>2],

//Status of door sensor,PB14
Door_status:
{
  "0": (bytes[6] & 0x80)? "CLOSE":"OPEN",
  "1": (bytes[6] & 0x80)? "CLOSE":"OPEN",
  "2": (bytes[6] & 0x80)? "CLOSE":"OPEN",
  "3": (bytes[6] & 0x80)? "CLOSE":"OPEN",
  "4": (bytes[6] & 0x80)? "CLOSE":"OPEN",
}[(bytes[6] & 0x7C)>>2],

//SHT2X,SHT3X temperature,PB6,PB7,units: 釐?
TempC_SHT:
{
  "0":((bytes[7]<<24>>16 | bytes[8])/10).toFixed(2),
  "2":((bytes[7]<<24>>16 | bytes[8])/10).toFixed(2),
}[(bytes[6] & 0x7C)>>2],

```

```

//SHT2X,SHT3X Humidity,PB6,PB7,units:%
Hum_SHT:
{
    "0": ((bytes[9]<<8 | bytes[10])/10) .toFixed(1),
    "2": ((bytes[9]<<8 | bytes[10])/10) .toFixed(1),
}[(bytes[6] & 0x7C)>>2],

//Distance,PA11,PB12,units:cm;
Distance:
{
    "1":((bytes[7]<<8 | bytes[8])/10) .toFixed(1),
}[(bytes[6] & 0x7C)>>2],

//ADC Channel 1,PA1,units:V
ADC_CH1V:
{
    "2":(bytes[2]<<8 | bytes[3])/1000,
}[(bytes[6] & 0x7C)>>2],
//ADC Channel 4,PA4,units:V
ADC_CH4V:
{
    "2":(bytes[4]<<8 | bytes[5])/1000,
}[(bytes[6] & 0x7C)>>2],

//DS18B20,PA9,units: 釐?
TempC2:
{
    "3":((bytes[7]<<24>>16 | bytes[8])/10).toFixed(2),
}[(bytes[6] & 0x7C)>>2],
//DS18B20,PA10,units: 釐?
TempC3:
{
    "3":((bytes[9]<<24>>16 | bytes[10])/10).toFixed(2),
}[(bytes[6] & 0x7C)>>2],

//Weight,PA11,PB12,units:g;
Weight:
{
    "4":(bytes[7]<<24>>16 | bytes[8]),
}[(bytes[6] & 0x7C)>>2],

//interrupt count
Count:
{
    "5":(bytes[7]<<24 | bytes[8]<<16 | bytes[9]<<8 | bytes[10]),
}[(bytes[6] & 0x7C)>>2],

```

```
}  
  return data;  
}
```