

```

#include <SPI.h>

#define LTC_CS PB12 //set ADC chip select MCP3551 LEITURA DE TENSÃO HARDWARE CS SPI2
#define LTC2400_SDO PB14

#define sspin PA1 //MCP3551 LEITURA DE CORRENTE-SPI2

#define LT_enable PB10//LT1970
#define slaveSelectPin PA4 //max5136 HARDWARE CS SPI1

#define SPI_SETTINGS          SPISettings(4000000, MSBFIRST, SPI_MODE0) // 4 MHz clock, MSB first, mode 0

//          MOSI  MISO  SCLK
SPIClass SPI_2(PB15, PB14, PB13);

void setup() {

pinMode (slaveSelectPin, OUTPUT);//max5136
pinMode (LT_enable, OUTPUT);//lt1970

////////////////////////////////MCP3551

pinMode (LTC_CS, OUTPUT); //MCP3551 LEITURA DE TENSÃO
digitalWrite(LTC_CS, HIGH); //set LCT2400 chip select pin HIGH to disable

pinMode(sspin,OUTPUT);//MCP3551 LEITURA DE CORRENTE
digitalWrite(sspin,HIGH);//for single conversion

SPI.begin();

SPI_2.begin(); //Initiallize the SPI 2
SPI_2.beginTransaction(SPI_SETTINGS);

void loop() {

digitalWrite(LTC_CS, LOW);//ativa leitura de tensão
volt=readMCP3551();

digitalWrite(sspin,LOW);//ativa leitura de corrente
current_1000 = readMCP3551();

}

//////////////////////////////// THIS SUBROUTINE CRASHES THE PROCESSOR //////////////////////////////////

int maxWrite(int address, int value1 , int value2) {
// take the SS pin low to select the chip:
// hspi->beginTransaction(SPISettings(500000, MSBFIRST, SPI_MODE0));
digitalWrite(slaveSelectPin,LOW);
// send in the address and value via SPI:
SPI.transfer(address,SPI_CONTINUE);
SPI.transfer(value1,SPI_CONTINUE);
SPI.transfer(value2,SPI_CONTINUE);
// take the SS pin high to de-select the chip:
digitalWrite(slaveSelectPin,HIGH);
}

// =====
// read one word from 22-bit ADC device  MCP3551

unsigned long readMCP3551() {
union{
int32_t values;
uint8_t aa[4];
} c ; // allow access to 4-byte word, or each byte separately

while(digitalRead(LTC2400_SDO)){//STM32

c.aa[2]=SPI_2.transfer(0x00);
c.aa[1]=SPI_2.transfer(0x00);
c.aa[0]=SPI_2.transfer(0x00);
c.aa[3]=0x00;

digitalWrite(sspin,HIGH);//MCP3551_CORRENTE
digitalWrite(LTC_CS, HIGH); //MCP3551_TENSÃO

//check if overflow has occurred-----
if((c.aa[2]&(1<<6))|(c.aa[2]&(1<<7))){
c.aa[2]&=~(1<<6);
// Serial.println(c.values);
// Serial.println("case1");
}
//check if sign bit is affected. if so, since it is two's compliment,
// subtract it from 2^N
else
if(c.aa[2]&(1<<5)){
c.values=0x400000-c.values;
}

return(c.values); // return unsigned long word
} // end readword()

```