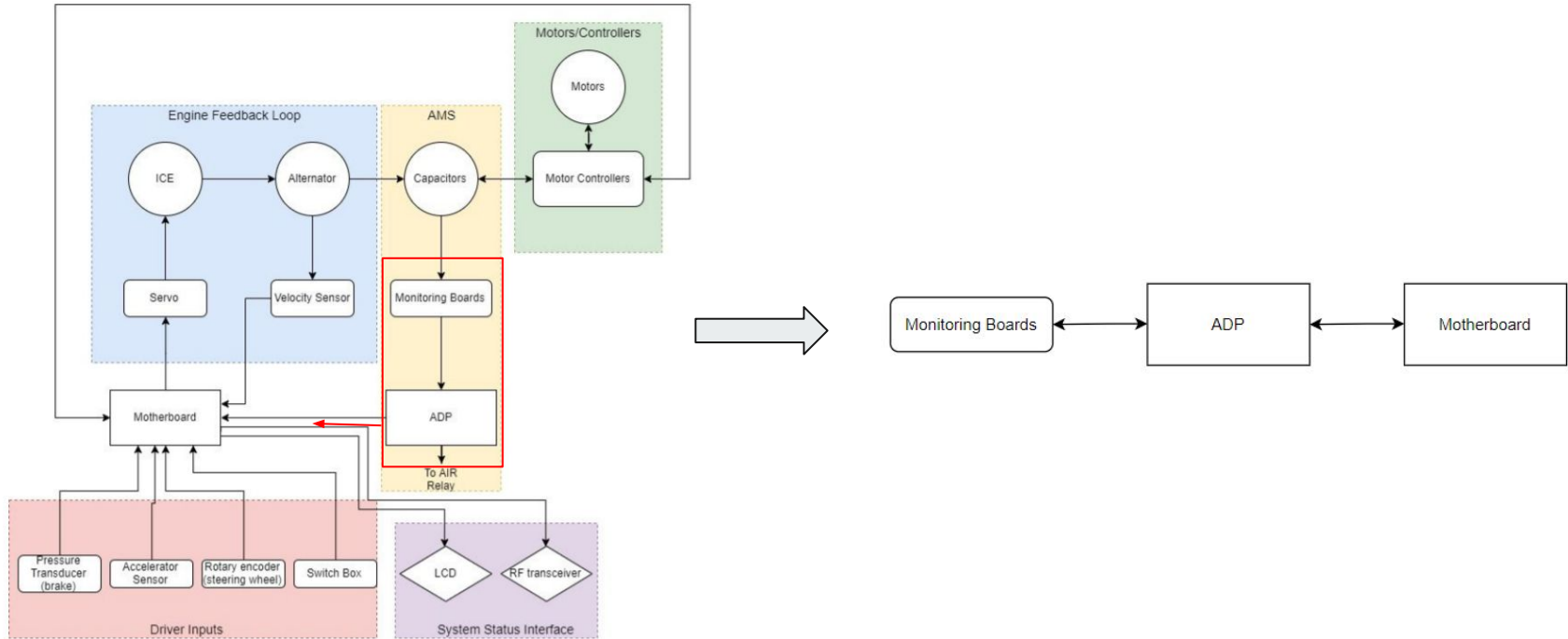




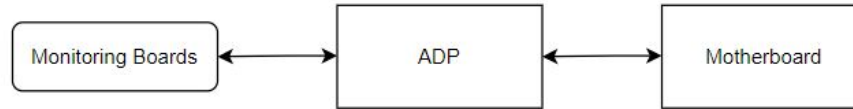
AMS System Guide

This guide assumes basic knowledge of the AMS subsystem and of the project in general

Overview

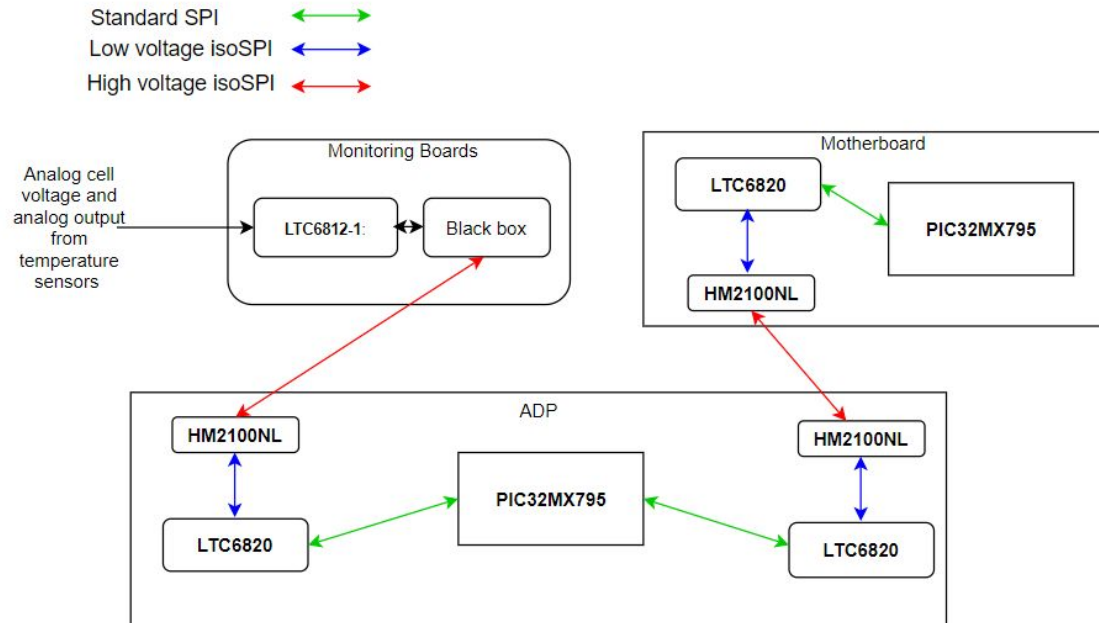


Overview



- The term “**monitoring boards**” is used interchangeably with the term “**demo boards**”
- “**ADP**” is used interchangeably with “**AMS motherboard**”

Overview



isoSPI/LT6820

- The ADP has two isoSPI modules
- It is the master in the interface between ADP and monitoring boards
- It is the slave in the interface between ADP and motherboard
- In our system, the transformer is replaced by the HM2100NL

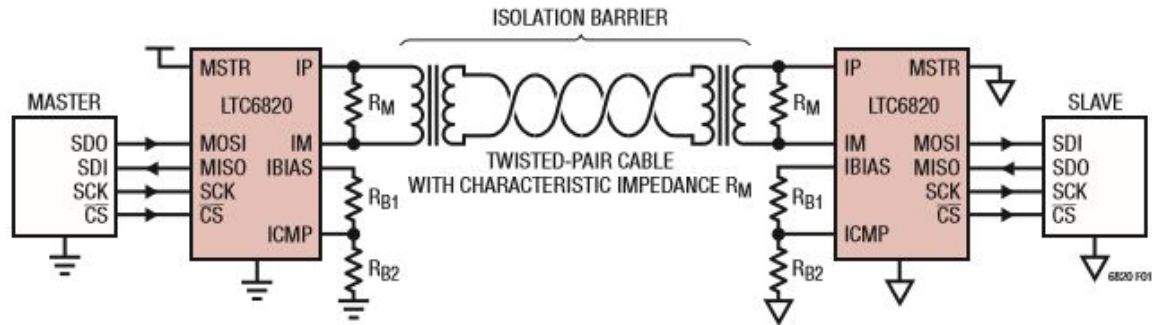
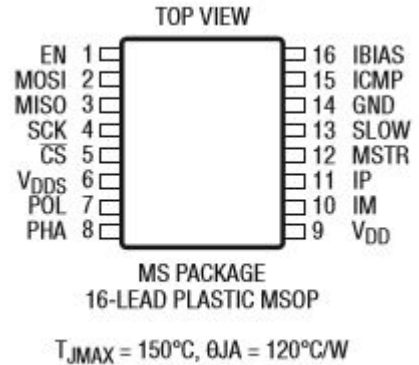
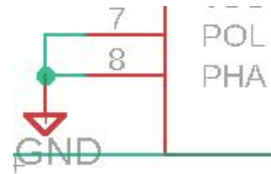


Figure from LT6820 data sheet

isoSPI/LT6820 - important pins

- The LT8620 can operate in any SPI mode
- Pins 7 and 8 control mode selection
- Current configuration is MODE 0
- See “Debugging” slides for more info



POL (Pin 6/Pin 7): SPI Clock Polarity Input. Tie to V_{DD} or GND. See Operation section for details.

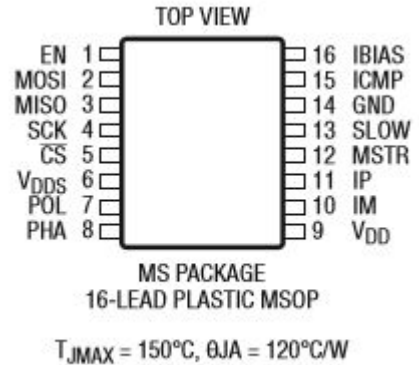
PHA (Pin 7/Pin 8): SPI Clock Phase Input. Tie to V_{DD} or GND. See Operation section for details.

Table 4. SPI Modes

MODE	POL	PHA	DESCRIPTION
0	0	0	SCK Idles Low, Latches on Rising (1st) Edge
1	0	1	SCK Idles Low, Latches on Falling (2nd) Edge
2	1	0	SCK Idles High, Latches on Falling (1st) Edge
3	1	1	SCK Idles High, Latches on Rising (2nd) Edge

isoSPI/LT6820 - important pins

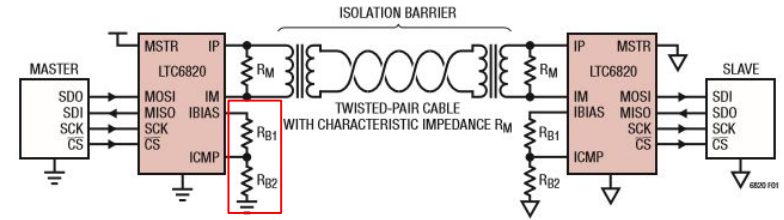
- The interface uses a window comparator
- $V_{IP} - V_{IM}$ (pin 11 - pin 12) is measured and compared to V_{TCMP} (pin 15)
- This is interpreted as +1, -1, or 0



isoSPI/LT6820 - important pins

- The comparator voltage is set with a voltage divider
- The current out of IBIAS also sets the drive current
- 1k ohm resistors are used in our to set $V_{TCMP} = 1V$
- Pulses are transmitted and received on IP, IM

ICMP (Pin 14/Pin 15): Isolated Interface Comparator Voltage Threshold Set. Tie this pin to the resistor divider between IBIAS and GND to set the voltage threshold of the interface receiver comparators. The comparator thresholds are set to 1/2 the voltage on the ICMP pin.



IBIAS (Pin 15/Pin 16): Isolated Interface Current Bias. Tie IBIAS to GND through a resistor divider to set the interface output current level. When the device is enabled, this pin is approximately 2V. When transmitting pulses, the sink current on each of the IP and IM pins is set to 20 times the current sourced from pin IBIAS to GND. Limit the capacitance on the IBIAS pin to less than 50pF to maintain the stability of the feedback circuit regulating the IBIAS voltage.

isoSPI/LT6820 - SPI to isoSPI

Data is transmitted in pulses consisting of a either $+V_A$ or $-V_A$ followed by a pulses of the opposite sign. CS changes are transmitted as long pulses and MOSI data is transmitted as short pulses, as seen in the two tables below.

Table 1. isoSPI Pulse Types

PULSE TYPE	FIRST LEVEL	SECOND LEVEL	ENDING LEVEL
Long +1	$+V_A$ (150ns)	$-V_A$ (150ns)	0V
Long -1	$-V_A$ (150ns)	$+V_A$ (150ns)	0V
Short +1	$+V_A$ (50ns)	$-V_A$ (50ns)	0V
Short -1	$-V_A$ (50ns)	$+V_A$ (50ns)	0V

Table 2. Master Communication Events

SPI MASTER EVENT	TRANSMITTED PULSE
\overline{CS} Rising	Long +1
\overline{CS} Falling	Long -1
SCK Latching Edge, MOSI = 1	Short +1
SCK Latching Edge, MOSI = 0	Short -1

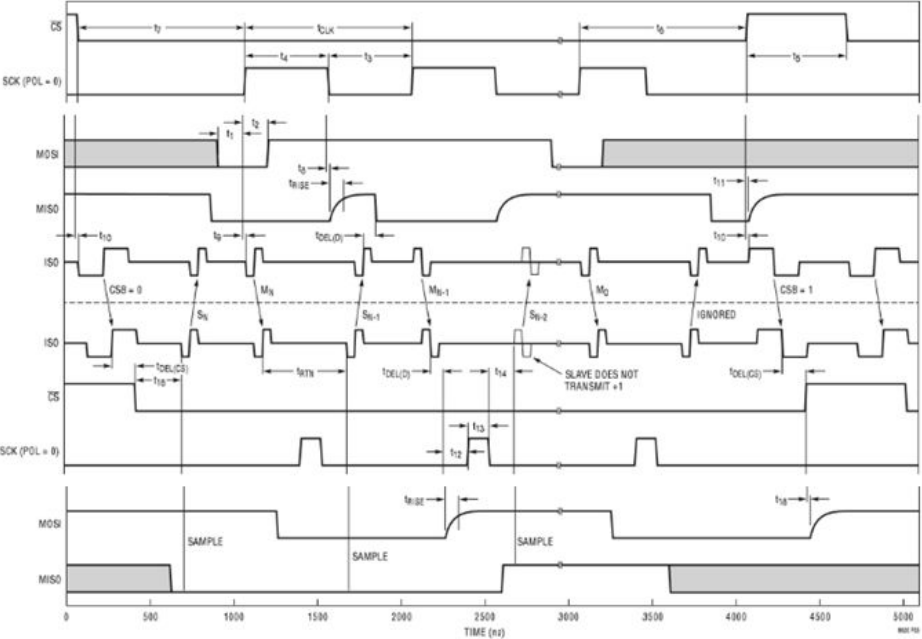
isoSPI/LT6820 - SPI to isoSPI

When the slave side receives a pulse it transmits a return pulse to set the state of MISO on the master side (just as in normal SPI operation). The table below describes this operation where MOSI and MISO describe the standard SPI interface on the slave side.

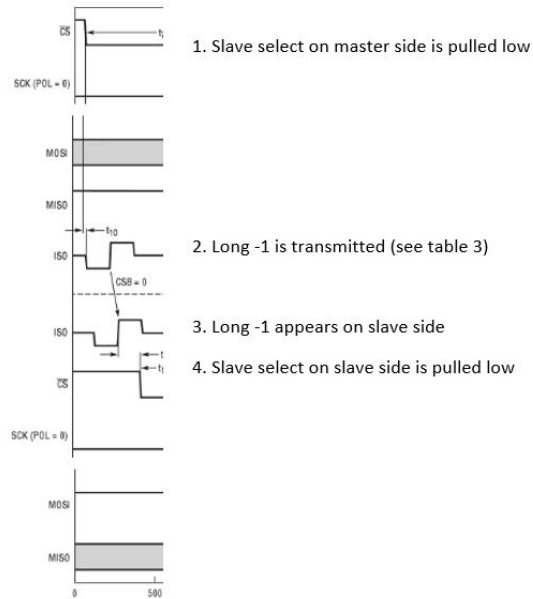
Table 3. Slave SPI Port Output

RECEIVED PULSE	SPI PORT ACTION	RETURN PULSE
Long +1	Drive \overline{CS} High	None
Long -1	Drive \overline{CS} Low	Short -1 Pulse if MISO = 0 (No Return Pulse if MISO = 1)
Short +1	1. Set MOSI = 1 2. Pulse SCK	
Short -1	1. Set MOSI = 0 2. Pulse SCK	

isoSPI/LT6820 - timing diagram

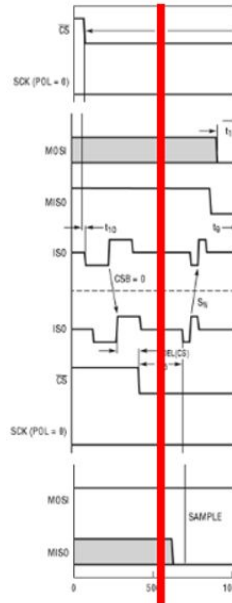


isoSPI/LT6820 - timing diagram example



At the start of transmission (1) the slave select pin on the master side is driven low by the standard SPI master. (2) In response, the LT6820 on the master side transmits a long -1 (see table 3) through the isoSPI interface. (3) The long -1 appears on the slave side of the isoSPI interface. (4) In response, the slave select pin on the slave side is driven low by the LT6820.

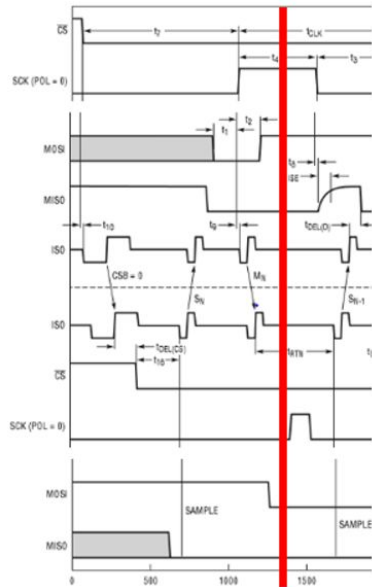
isoSPI/LT6820 - timing diagram example



1. MISO on slave side is sampled low
2. Short -1 is transmitted by slave side (see table 3)
2. Short -1 appears on master side
3. MISO on slave side is driven low

The standard SPI slave is now operating and driving the MISO pin is low. The LT6820 on the slave side (1) samples the MISO pin on the slave side and (2), transmits the corresponding pulse through the isoSPI interface. (3) That pulse the appears on the master side. (4) In response the MISO pin on the master side is driven low by the LTC6820.

isoSPI/LT6820 - timing diagram example

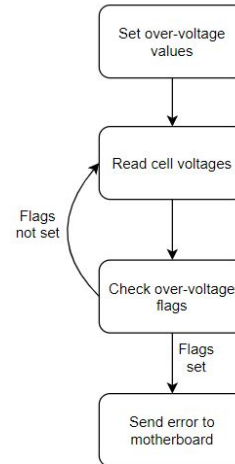


1. Falling edge of clock on master side
2. MISO on master side returns to high
3. MISO on master side is driven low
4. short -1 appears on master side
5. short -1 is transmitted by slave side
6. MISO on slave side is sampled low

The standard SPI slave must respond, but first (1) the standard SPI master drives the clock low. (2) The LT6820 on the master side returns the MISO pin to its default state. Now the slave side responds. First, (3) the LT6820 on the slave side samples the MISO pin. (4) The LT6820 on the slave side transmits the corresponding pulse through the isoSPI interface which (5) appears on the master side. (6) The LT6820 on the master side drive the MISO pin low.

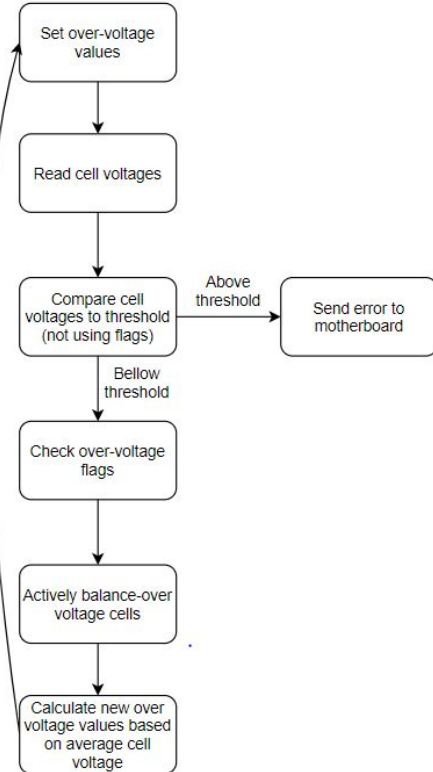
Code

- A complete flowchart is drawn on the whiteboard in the SAE lab
- Basic operation involves setting over-voltage values, initiating voltage read, then checking over-voltage flags



Code

- Code that implements cell balancing should look like this:





Other Info.

DC2350A-A (demo boards):

The demo boards are commercially available PCBs that utilize the LTC6812-1 battery monitor chip. Much of the information in the LTC6812-1 data sheet may be disregarded, such as pin functions and maximum ratings, as the external circuitry is already built on the demo boards.

The demo boards take their power supply from the cells that they are monitoring (in this case the capacitor bank.) Therefore, if the capacitors are discharged below a minimum voltage the demo boards will not function.



Other Info.

Demo Manual DC2350A Document - Possible Points of Confusion:

The “DEMO MANUAL DC2350A” lists a variety of options for connecting to the demo boards such as using the **DC2792** and **DC2026** together. NONE OF THESE ARE USED. Even the section titled “Using Other SPI Master Boards” is not directly relevant as that refers to connecting to the demo board using a standard SPI interface. The real system is most closely related to using the **DC1941** and **DC2026**. However, the functions of both boards are built into the AMS motherboard. Therefore, much of the information in “DEMO MANUAL DC2350A” is not directly useful.

However, we can extrapolate some information from this manual. Page 8 gives instructions for setting jumpers on the **DC1941** when it is connected between the **DC2026** and the demo boards. Cross referencing this with the DC1941’s own manual (which isn’t included but is easily found online) leads to the conclusion that the demo boards work on SPI mode 3 (with POL and PHA both high.) Therefore, we believe that pins 7 and 8 on the LTC6820s should be pulled high.

There is currently a voltage divider made of through-hole type resistors soldered vertically to the general purpose input-output pins on the demo boards. They are there to simulate the voltages from temperature sensors that will eventually be added to the system.



Other Info.

Some debugging tips:

There are two **DC2026 Linduino** boards in the SAE Team lab. The **DC2026** can be used to directly connect a PC to the demo boards. There are a variety of Arduino sketches that can be uploaded to the **DC2026** to run diagnostics. Some of these sketches are also useful as a reference for developing additional code to run. Instructions are located in the “DEMO MANUAL DC2350A” document.

Demo Board Jumpers:

There are 4 jumpers on the demo boards. JP1-JP3 must be set to 0 if using the **DC2026** (because the **DC2026** uses standard SPI) and to 1 if using isoSPI.

There is a method for running the demo boards independently of the battery stack outlined on page 2 of the “DEMO MANUAL DC2350A” document. Resistors are connected in series among the cell inputs on the demo board and voltage of sufficient magnitude is applied (we used two of the 12 V batteries from the hybrid lab.)



Other Info.

LTC6812-1:

This is the “brain” of the demo boards. It contains multiple ADCs to read the voltages from the 15 capacitor cells as well as analog inputs from the GPIO pins. The GPIO pins are used for temperature sensors.

The external hardware for the **LTC6812-1** is already set up on the demo boards. The most important information for this project is contained in the “Operation” section of the data sheet, specifically between pages 56 and 66. This section contains command tables for writing to the **LTC6812-1**.

Code example

```
void checkVolts(void){
    CS=0;
    CS_test = CS;

    WriteReadSPI(0x03);
    WriteReadSPI(0x60);
    // Write PEC
    char x[2] = {0x03,0x60};
    unsigned short u = pecl5(x,2);
    unsigned short highbyte = (u >> 8);
    unsigned short lowbyte = (u & 0x00FF);
    WriteReadSPI(highbyte);
    WriteReadSPI(lowbyte);
    WriteReadSPI(0x00);
    u = WriteReadSPI(0x00);
    while(u == 0x00){
        u = WriteReadSPI(0x00);
    }

    CS=1;
    CS_test = CS;
}
```

Command to start ADC cell voltage conversion

Start Cell Voltage ADC Conversion and Poll Status	ADCV	0	1	MD[1]	MD[0]	1	1	DCP	0	CH[2]	CH[1]	CH[0]
---	------	---	---	-------	-------	---	---	-----	---	-------	-------	-------

Figure from table 36 in data sheet

Here MD[1] = 1, MD[0] = 0, DCP = 0, CH[2-0] = 0.

Referencing table 37, this means:

- operation in either 7kHz or 3kHz mode, depending on the stats of the ADCOPT bit in Configuration Register Group A (which is written in another function.)
- The discharge timer is not permitted.
- All cells will be measured.

Code example cont.

```
void checkVolts(void){
    CS=0;
    CS_test = CS;

    WriteReadSPI(0x03);
    WriteReadSPI(0x60);
    // Write PEC
    char x[2] = {0x03,0x60};
    unsigned short u = pec15(x,2);
    unsigned short highbyte = (u >> 8);
    unsigned short lowbyte = (u & 0x00FF);
    WriteReadSPI(highbyte);
    WriteReadSPI(lowbyte);
    WriteReadSPI(0x00);
    u = WriteReadSPI(0x00);
    while(u == 0x00){
        u = WriteReadSPI(0x00);
    }

    CS=1;
    CS_test = CS;
}
```

Every command must be followed by a packet error code (or PEC.)
There is a function called pec15 written by the 2018-2019 team that computes the PEC.

Wait for transmission to complete



Next steps

- Verify operation with current system (this will take some time)
- Rewrite code in a way that doesn't use over-voltage flags