Firmware Design

In this document an overview of the firmware design for the DeciZebro power system will be given. Several flowcharts will be used to illustrate our proposed power control method.

Figure 1 shows the general flowchart for the complete system. After a reset signal has been given the Microcontroller will initialize and reset all pins and counters. Next the system will check the sensors and based on that determine the next state of the power system. Unless shutdown, the values of the vregs will be updated for the ZebroBus to readout and the process will repeat.
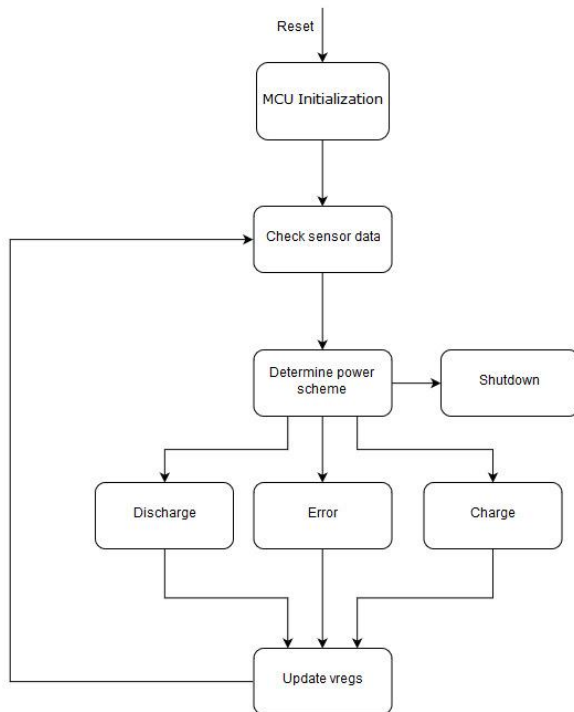

Figure 1 - General Control flowchart

Figure 2 shows what happens when there is an interrupt from the Main ZebroBus. If the ZebroBus requires a read command the vreg fields will be send to the master. If the ZebroBus writes, the corresponding values will be updated in the vreg field.
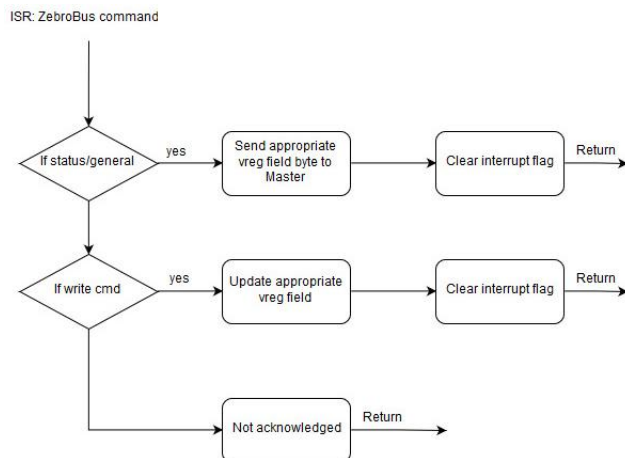

Figure 2 - Interrupt service routine ZebroBus

Initialization

Load stored values from

Check reset log
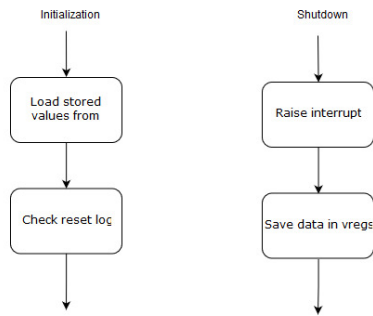
Shutdown

Raise interrupt

Save data in vregs

Figure 3 - Initialization and shutdown state

During initialization the values will be loaded from the EEPROM and the reset log will be checked. When the power system needs to shutdown it will raise the interrupt pin on the Zebrobus, so the main system can still save its data safely.
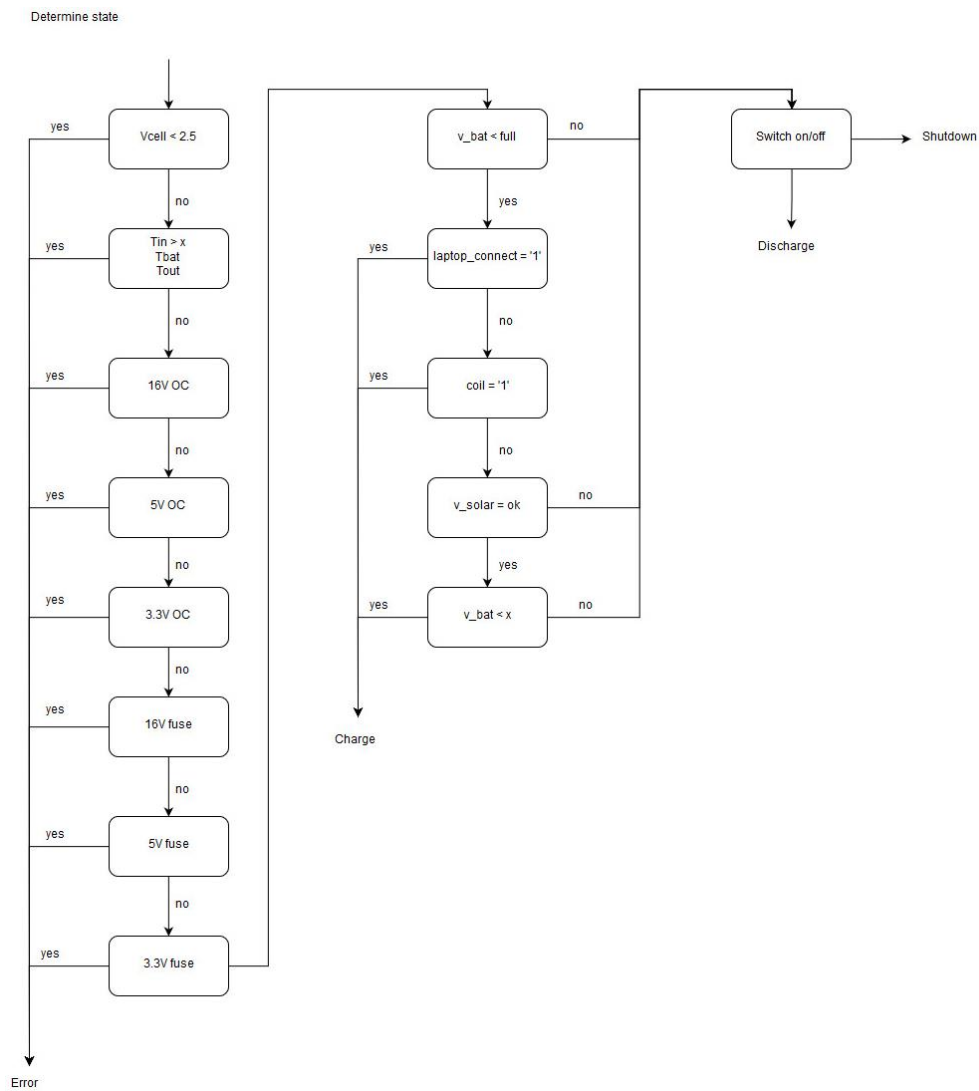
Determine state

| yes | Vcell < 2.5 | | v_bat < full | no | | Switch on/off | → Shutdown |
no

| yes | Tin > x / Tbat / Tout | | yes | laptop_connect = '1' | | | Discharge |
no | | no |

| yes | 16V OC | | yes | coil = '1' |
no | | no |

| yes | 5V OC | | v_solar = ok | no |
no | | yes |

| yes | 3.3V OC | | yes | v_bat < x | no |
no

| yes | 16V fuse | | Charge |
no

| yes | 5V fuse |
no

| yes | 3.3V fuse |

Error

Figure 4 - Determine power scheme state

Figure 4 shows the logic behind the state choice. If there is a sensor value which is not valid for the safety of our system, the error state will be raised. Depending on which inputs are available and the

state of the battery, the Microcontroller goes in either Charge or Discharge state. Which will be further elaborated in the next two schemes.
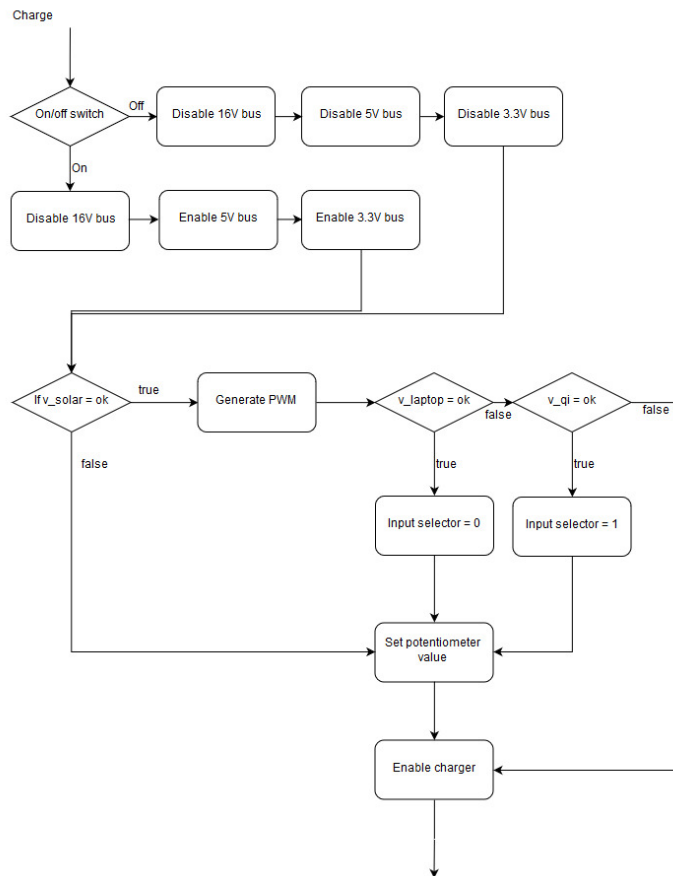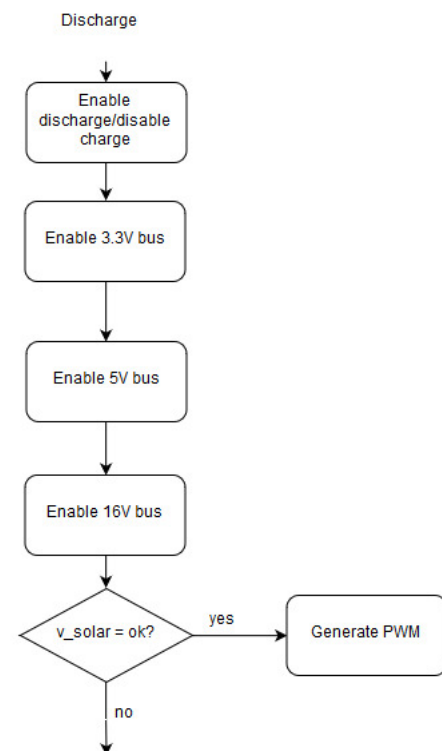
Figure 5 - Charge state

Figure 6 - Discharge state

In the Charge state the 16V output will be disabled to prevent a too high current on the powerbus. Next up the solar panel will be checked if the voltage is ok a pwm will be generated. Then if the voltage on the laptop input is ok the input selector will be set to 0, which means the cuk will be connected to the laptop charger. If this is not the case and the qi is within voltage range the qi will be selected. Afterwards the potentiometer is set to the right value and the charge will be enabled on the battery system.

In Discharge state the outputs will be enabled and if there is sun available, the solar panels can help give some power to the outputs to save battery.
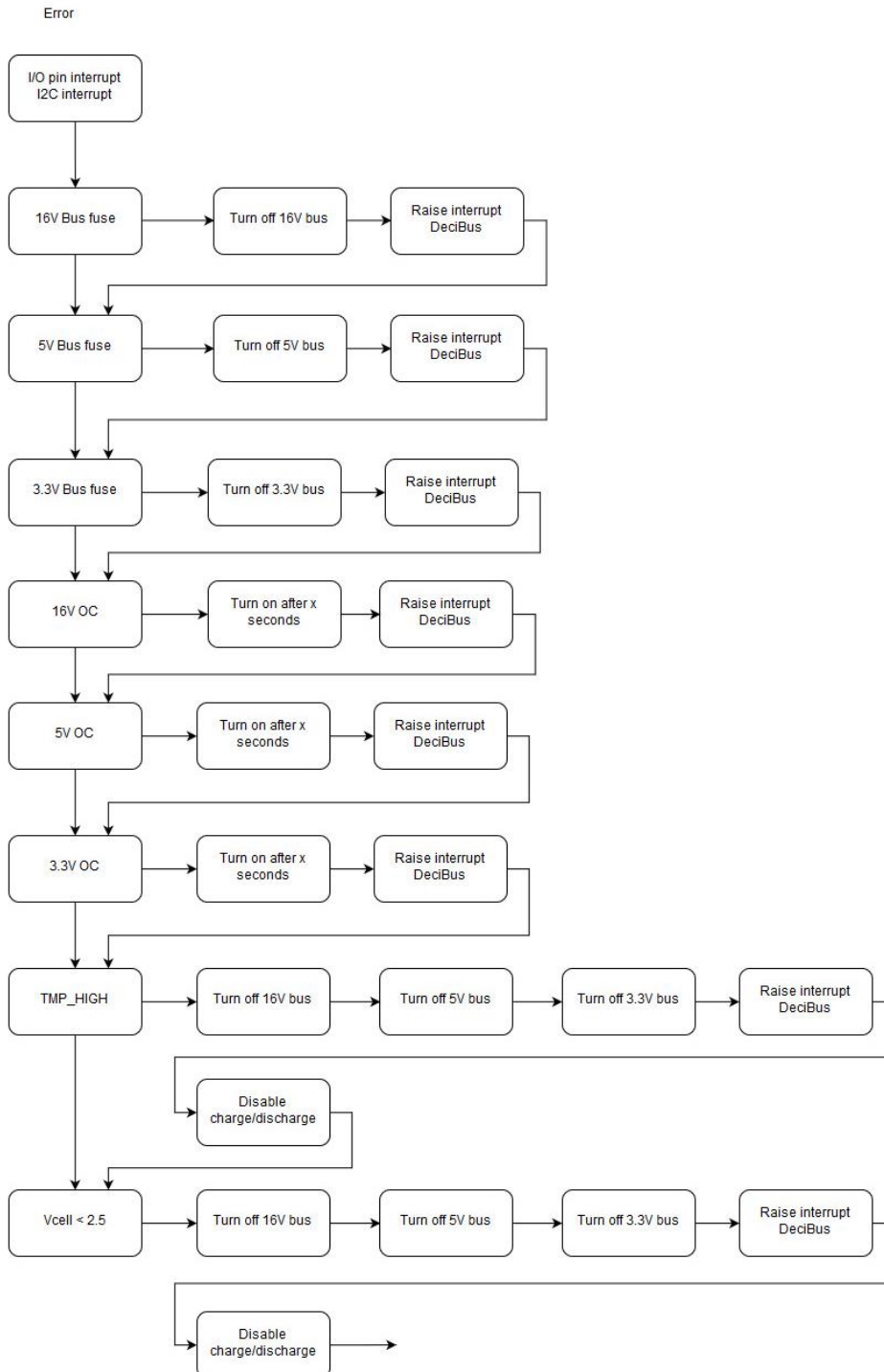
Figure 7 - Error flowchart

Figure 7 shows the error flowchart, if a fuse blown or overcurrent the specific output is automatically turned off, the interrupt pin on ZebroBus will be set high and for the overcurrent case it will be retried after some time. In case of an dead battery or too high temperature, all outputs will be turned off and the interrupt pin on ZebroBus will be raised. Also the battery management system will be disabled to prevent damage.