Current Monitoring System

Acesses:
GitHub Login: solarexeterboat@gmail.com
Passoword: BNBLf5MuTXvSsp8

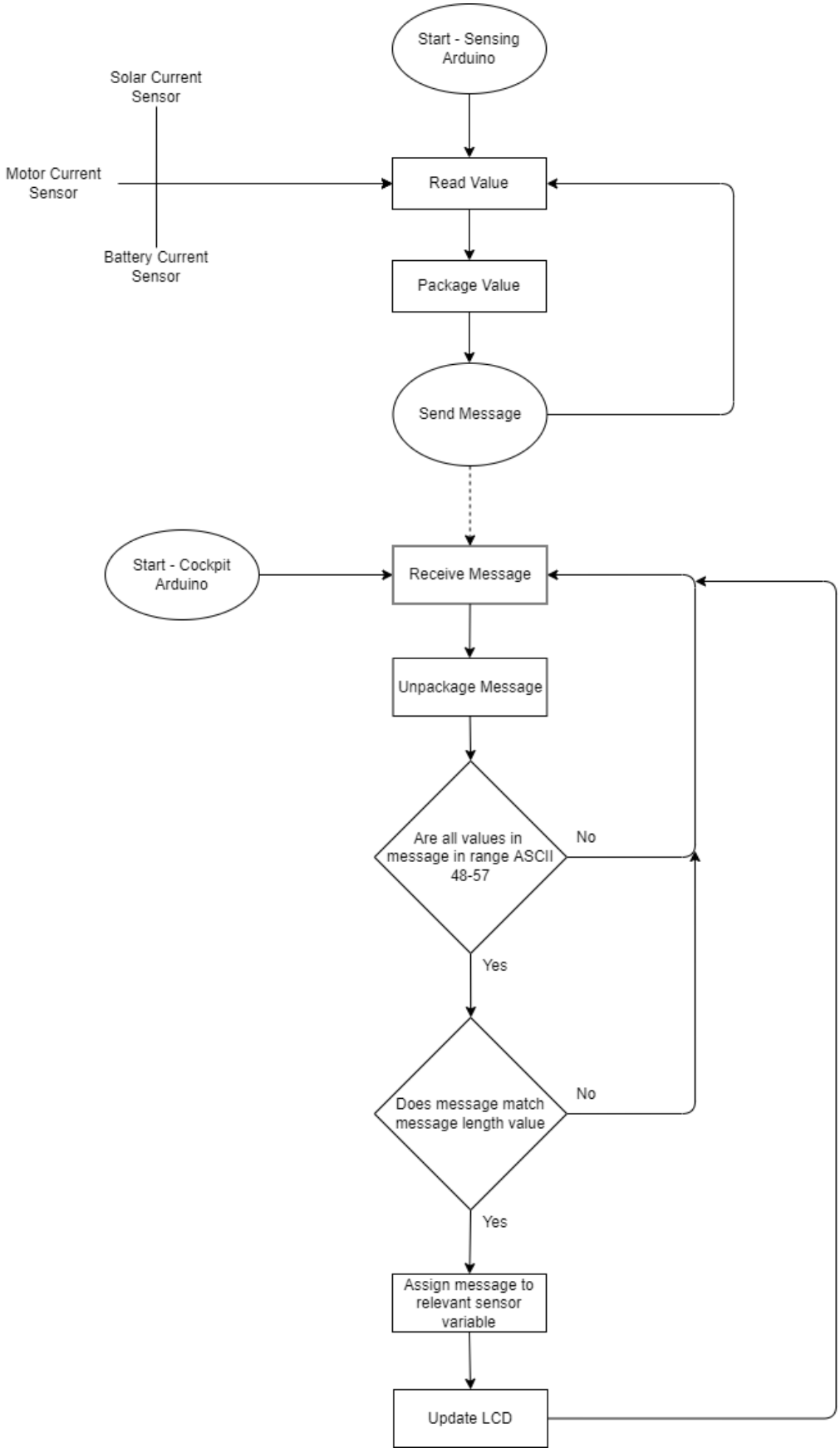
Introduction:

Designed to work on the twin Arduino system on SoleX, this script contains all the relevant functions and variables to set up and run the current sensing and monitoring system. It is designed to be extremely modular allowing for quick and easy changes moving forward.

Setup:

- Connect 2 digital pins on each Arduino to one another assign this RX and TX. (RX on the sensor board is TX on the cockpit board and vice versa)
- Ensure they Arduinos share a ground connection
- Update the code to reflect the correct pins for RX,TX and the sensor connections.
- Set which board code will be running on in Void Loop()
- Upload code to relevant Arduino.

General Flow Chart:



Global Variables:

Variable Name	Type	Description
HallEffect_SolarOutput_Pin	int	Pin that the solar output hall effect sensor is connected to
HallEffect_BatteryInput_Pin	int	Pin that the battery input hall effect sensor is connected to
HallEffect_MotorInput_Pin	int	Pin that the motor input hall effect sensor is connected to
HallEffect_SolarOutput_NominalCurrent	int	Nominal current of the solar output hall effect sensor
HallEffect_BatteryInput_NominalCurrent	int	Nominal current of the battery input hall effect sensor
HallEffect_MotorInput_NominalCurrent	int	Nominal current of the motor input hall effect sensor
Analog_V_Range	int	Max of analogue range of values from sensor
Analog_Bit_Range	int	Max of digital range of values from sensor
pinRX	int	RX pin
pinTX	int	TX pin
numBytes	byte	Size of message
softwareBaudRate	int	Communication transfer rate
delayAfterSendingMessage	int	Delay between sending messages /ms

Functions:

void setup()

Description:

Begins software serial and serial communication

void loop()

Description:

Runs the Arduino, chose between Cockpit_Arduino or Sensor_Arduino

void Sensor_Arduino()

Description:

Controls the sensor arduino located in electronics bay and connected to the hall effect sensors, simple read value and send message

void Cockpit_Arduino()

Description:

Controls cockpit arduino, includes message extraction, verification and assignment to relevant sensor

void Wrap_Send_Message(int Reading, char ID_Marker)

Parameters:

- Int Reading – What do you want to send as message, designed for int value from sensor (0-1024)
- Char ID_Marker – Identifier for relevant sensor

Description:

Sends message with Id maker, Length marker, start and end markers

Example Message Sent:

Char	<	3	S	6	0	3	>
Description	Start Maker	Message Length	Sensor ID	Message, digital reading			End Marker

void recv_Wraped_Message()

Description:

Receives a message wrapped using start and end markers to identify Contents (CITE)

Example message received when not corrupt:

Char	3	S	6	0	3
Description	Message Length	Sensor ID	Message, digital reading		

bool TrueINT(int val, int minimum, int maximum)

Parameters:

- int Val – value to test
- int Minimum – minimum value it can be
- int Maximum – maximum value it can be

Returns:

- True (1) or False (0)

Description:

Determines if a integer is within a defined range

double LEM_HASS_Current(int Digital_Sensor_Reading, int Vref, int Ipn)

Parameters:

- Int digital_sensor_reading – 10bit value to convert to analogue current reading

- Int Vref – Reference voltage see sensor data sheet
- Int Ipn – Nominal current of sensor

Returns:

- Double current – Current value (A)

Description:

Equation for conversion of 10bit digital value to analogue current reading for LEM_HASS Current Transducer

double LEM_HTFS_Current(int Digital_Sensor_Reading, int Vref, int Ipn)

Parameters:

- Int digital_sensor_reading – 10bit value to convert to analogue current reading
- Int Vref – Reference voltage see sensor data sheet
- Int Ipn – Nominal current of sensor

Returns:

- Double current – Current value (A)

Description:

Equation for conversion of 10bit digital value to analogue current reading for LEM_HTFS Current Transducer