TE-EPC-2020-131-LD

Coding Challenge

Georgios Leventis

February 10, 2021

The challenge was completed in full. I provide the code for what was asked, as well as, demo files for the tmod.h, supervision.h, libmod.a and libsupervision.a. The make file can be found within the Release folder and it should be run from this directory.

For the shake of the exercise I made these assumptions:

- The ADC channels have the same binary value allocation for both of the signal types. As such there is no difference in the conversion of the binary values although the range of the voltage meters is 10V, whereas the range of the Ampere meters is 8V.
- I used the C++ type of output streams (not C `stdout`).
- The list of Sensors returned to the user is the general list type with Sensors which also carries the values of the binary reading. This could easily be changed to a more specific type should that be considered appropriate and needed.
- The values inserted for the scaling factor and offset are in float format.
- The signal type can either be a numeric (1 or 2) or the actual strings demonstrated in the examples.
- Atomics have been used for thread synchronization as most modern processors support them.
- The most recent allowed C\C++ standard was used. Devtools-7+ are needed for compilation on Centos7.
- The YAML schema used is:

```
Sensor Readings:
   hardware-address: xx.xxx:
      Current: xx.xxx
      Minimum: xx.xxx
      Maximum: xx.xxx
```

The project should be complete and self explanatory. Due to extreme time constraints I miss specific code comments but should you need any assistance or explanation to understand why I am doing any part of it I will be more than happy to explain it.

I will try to update this readme file online at its github repository where the code can also be reviewed. The github link is:

https://github.com/geoleven/CernRTSTempManagement

Thank you for taking the time to review my code.