

Firmware Development of Net Power Supply with SCPI

Aegis Power Systems, Inc.



PROBLEM STATEMENT

Aegis' customer needs an industrial power supply with Ethernet SCPI capability communicating on LAN

Team's responsibility: early-stage firmware development

- TCPIP Communications
- Saving and Retrieving Network Configurations
- Parsing SCPI Commands

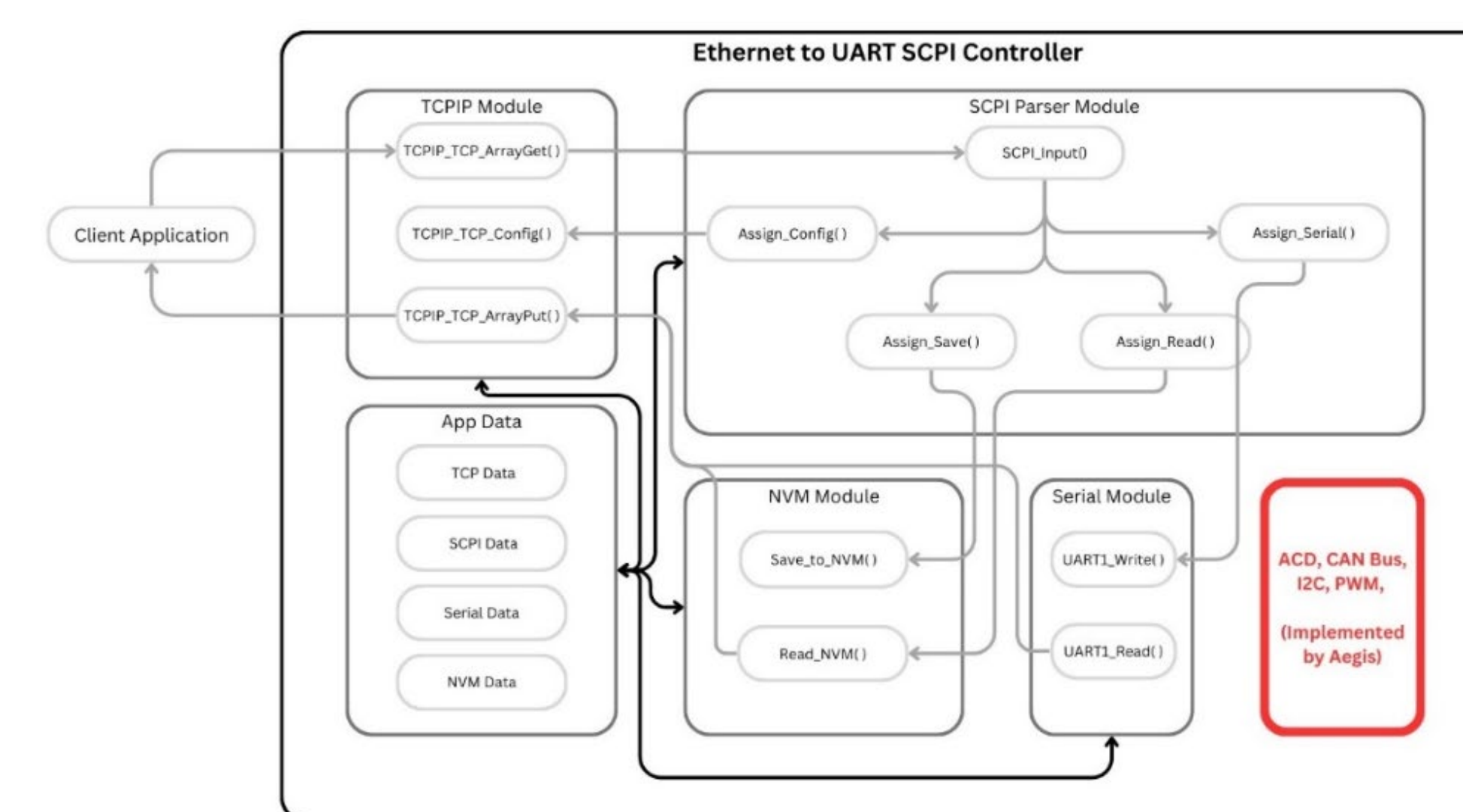
REQUIREMENTS

Req #	Requirement Description	Motivation
1	The microcontroller must with- stand temperature range of -40° to 125° Celsius	The final embedded chip will be deployed in harsh environments by the customer
2	The controller's footprint must be less than or equal to the size of a credit card	The sponsor wishes to place the controller into a weatherproof case
3	The controller must have extra GPIO pins	The sponsor wishes to add additional peripherals after receiving the controller
4	The firmware code must be well documented and readable	The sponsor plan to maintain the firmware for further improvement and bug fix
5	The Controller must communicate via TCPIP protocol	The controller will be connected to a local area network via Ethernet

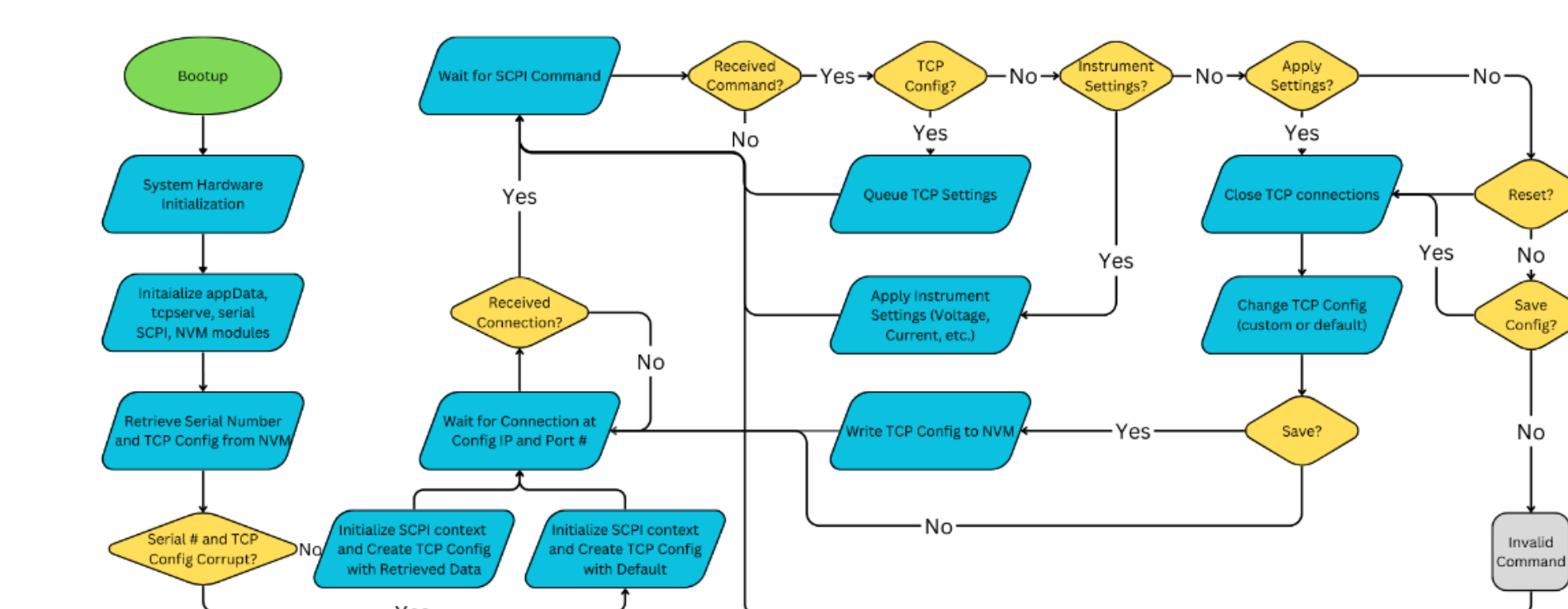
FINAL DESIGN, APPROACH, PLAN

- Finite state machine modules for code organization
- Debug console for live controller feedback
- Job assignment process
- Flag checking
- Utilization of IEEE 488.2 compliant SCPI parser library [1]

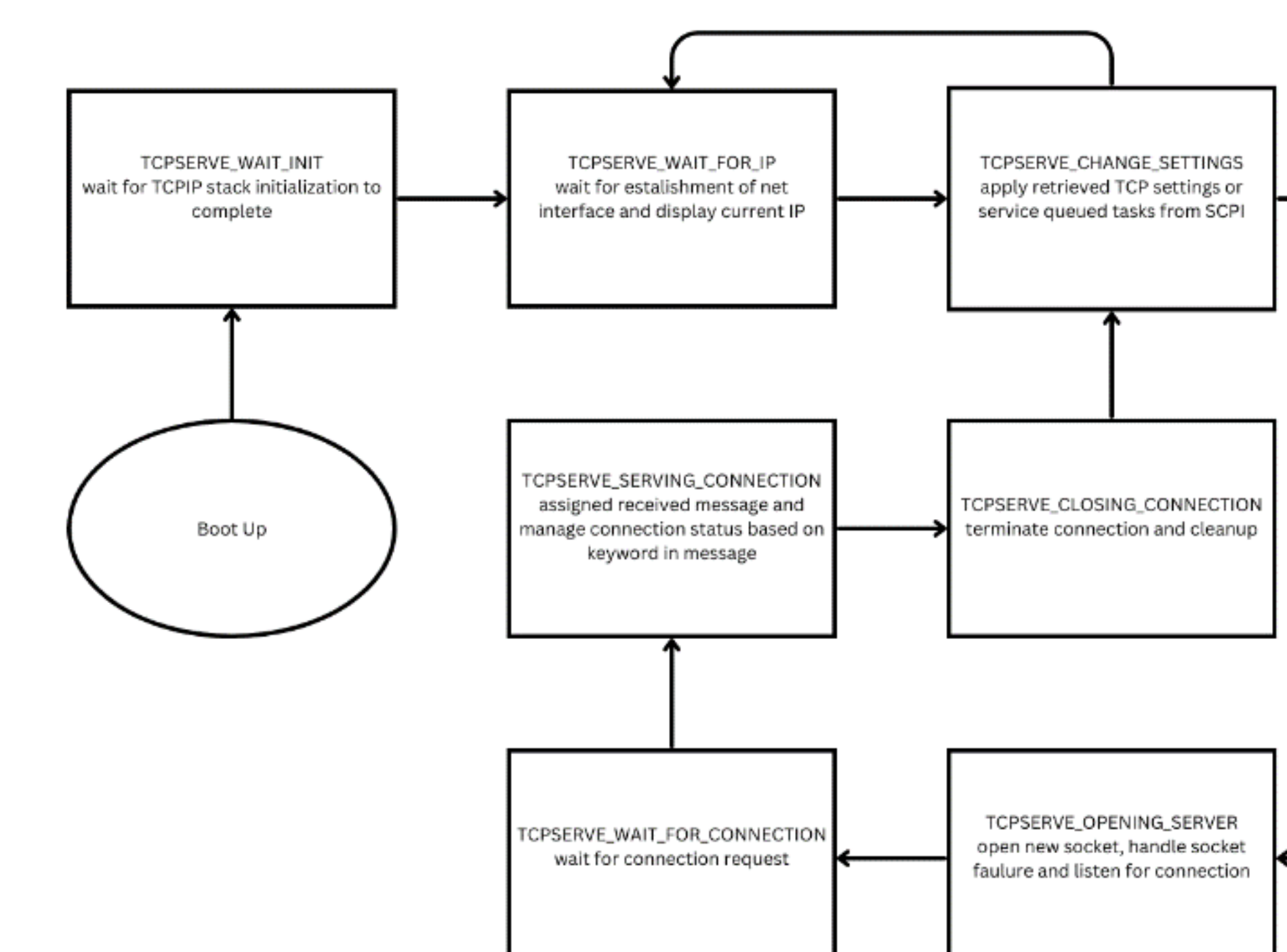
Architecture



Program Flowchart



TCP/IP module states



RESULTS

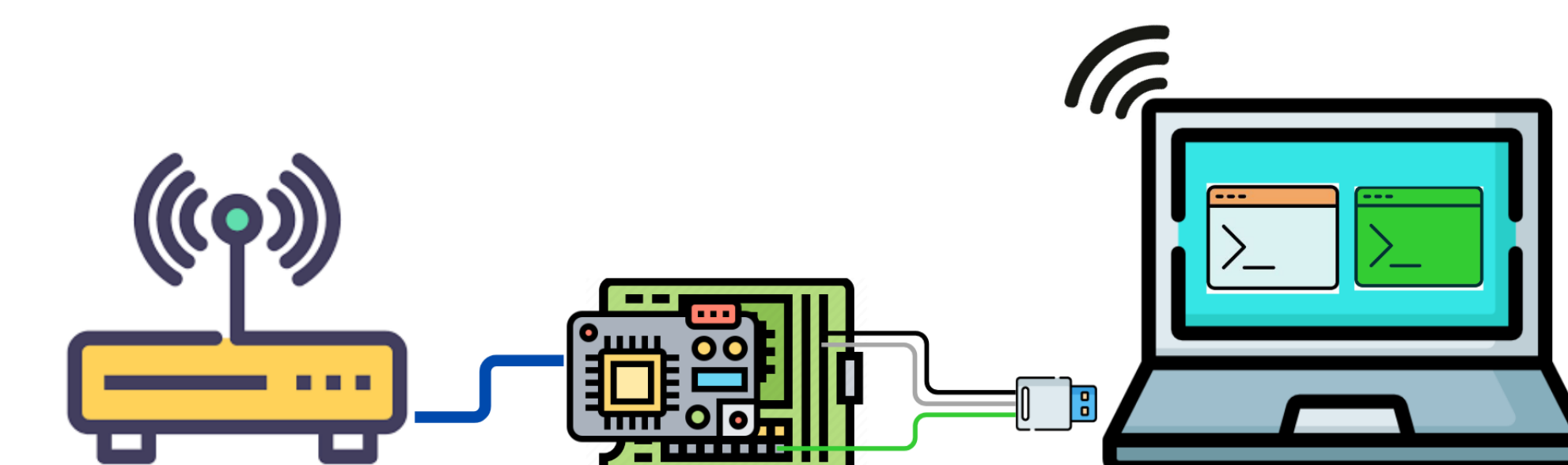
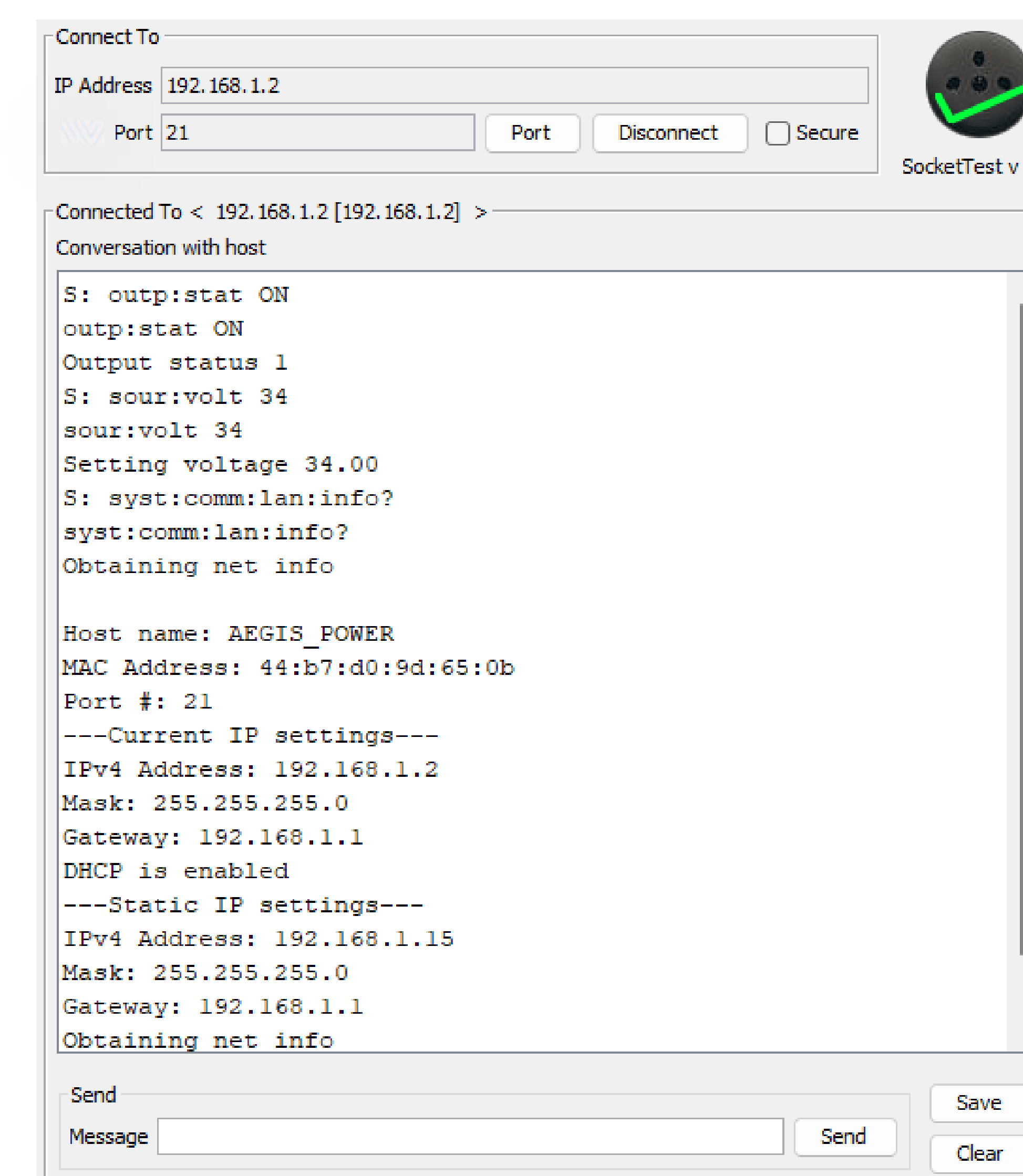
- Working server function
- Fully functional SCPI parser
- Save/retrieve/apply TCPIP configuration
- SCPI command error handling

```
netinfo
----- Interface <eth0/PIC32INT> -----
Host Name: AEGIS_POWER - NBNB disabled
IPv4 Address: 192.168.1.2
Mask: 255.255.255.0
Gateway: 192.168.1.1
DNS1: 192.168.1.1
DNS2: 0.0.0.0
MAC Address: 44:b7:d0:9d:65:0b
dhcp is ON
Link is UP
Status: Ready
>
>Waiting for Client Connection on port: 21
Received a connection
FIFO have 14
Server assigned outp:stat ON

scpi process
scpi processed
In Echo
job from SCPI/UART
SCPI 2 Serial
UART sending
```

```
FIFO have 20
Server assigned syst:comm:lan:save

scpi process
scpi processed
In Echo
job from SCPI/UART
Applying/Saving settings
cksm:853
cksm:285430164
Write P0 done
Write P1 done
Write P2 done
Save P0 result 1 1
Save P1 result 1 1
Save P2 result 1 1
Waiting for Client Connection on port: 21
```



SUMMARY AND CONCLUSIONS

The team has successfully delivered valuable results to the sponsor by meeting all project requirements and objectives. The developed firmware showcases the capability to receive, process, and execute all specified SCPI commands as requested by the sponsor. These commands encompass settings for the power supply's voltage, current, and output status.

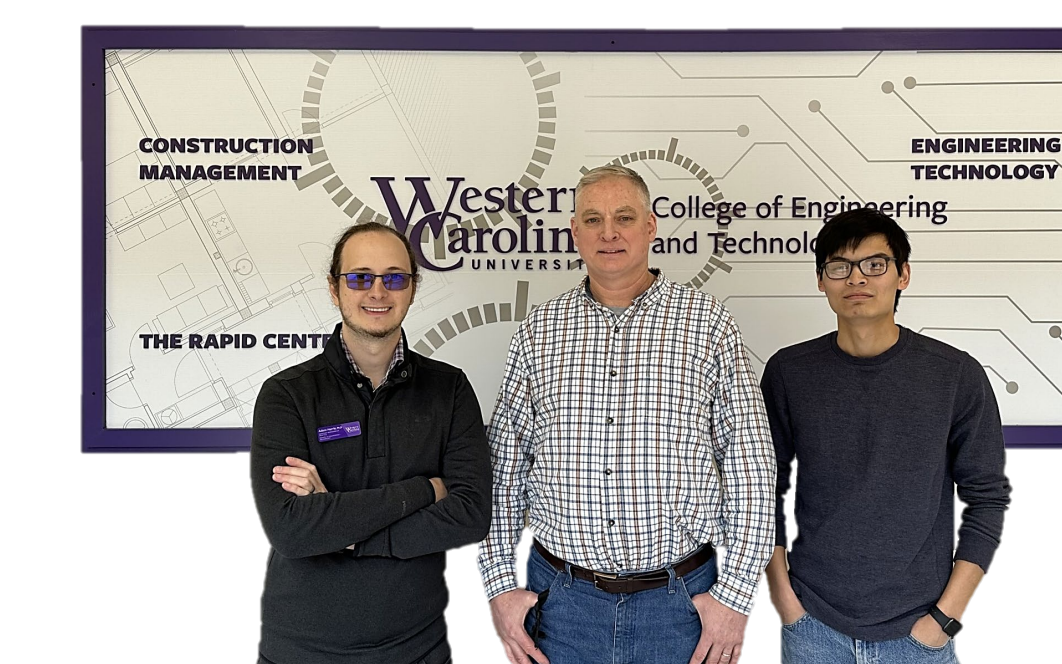
Furthermore, the controller can receive SCPI commands through TCPIP, enabling it to process output setting commands and modify network configuration settings, which include port numbers, IP addresses, and status. Additionally, a system console has been integrated into the controller, providing users with live debug messages for enhanced visibility and troubleshooting capabilities.

FUTURE WORK

For future work, the sponsor should consider implementing the current firmware into their chosen microcontroller from the PIC32MZ family. The next capstone team or sponsor should then focus on implementing I2C, PWM, and CAN Bus to control displays, cooling fans, and power modules.

TEAM & ACKNOWLEDGEMENTS

- Charles Maddrey (ECET)
- William Wright (BSEE)
- Dr. Adam Harris (Mentor, Prof. of Practice)
- Grant Hilgert (Firmware Engineer, Aegis)



References

1. IEEE 488.2 compliant SCPI parser library <https://github.com/j123b567>