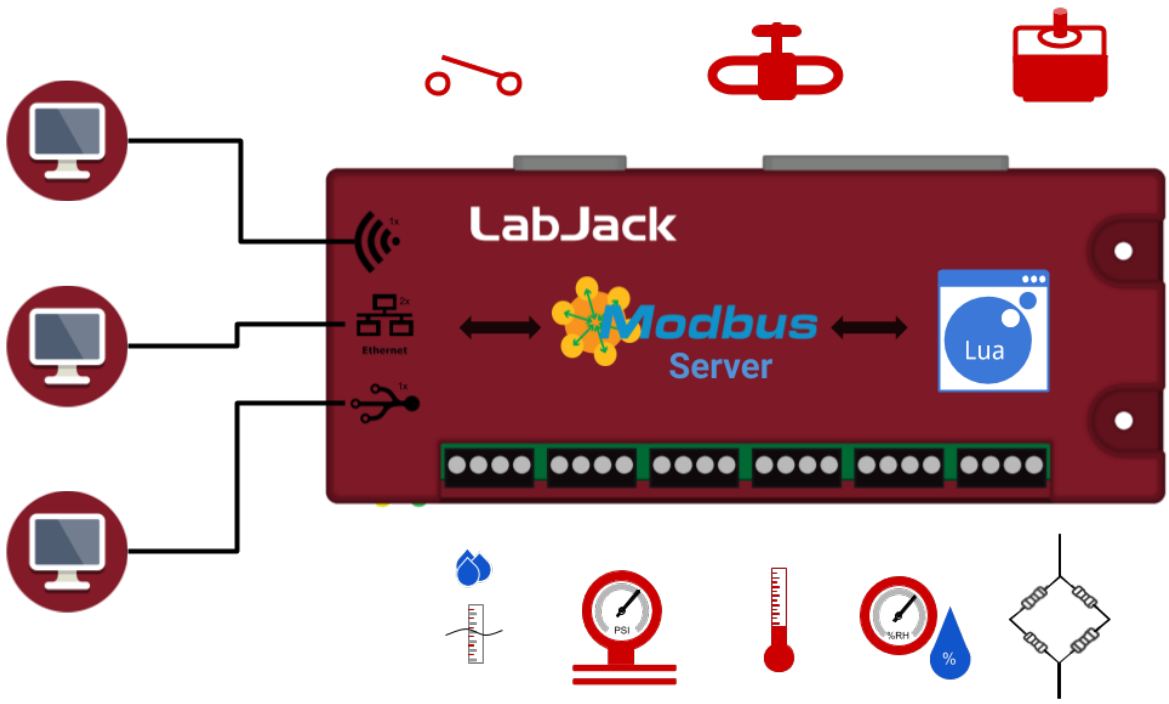
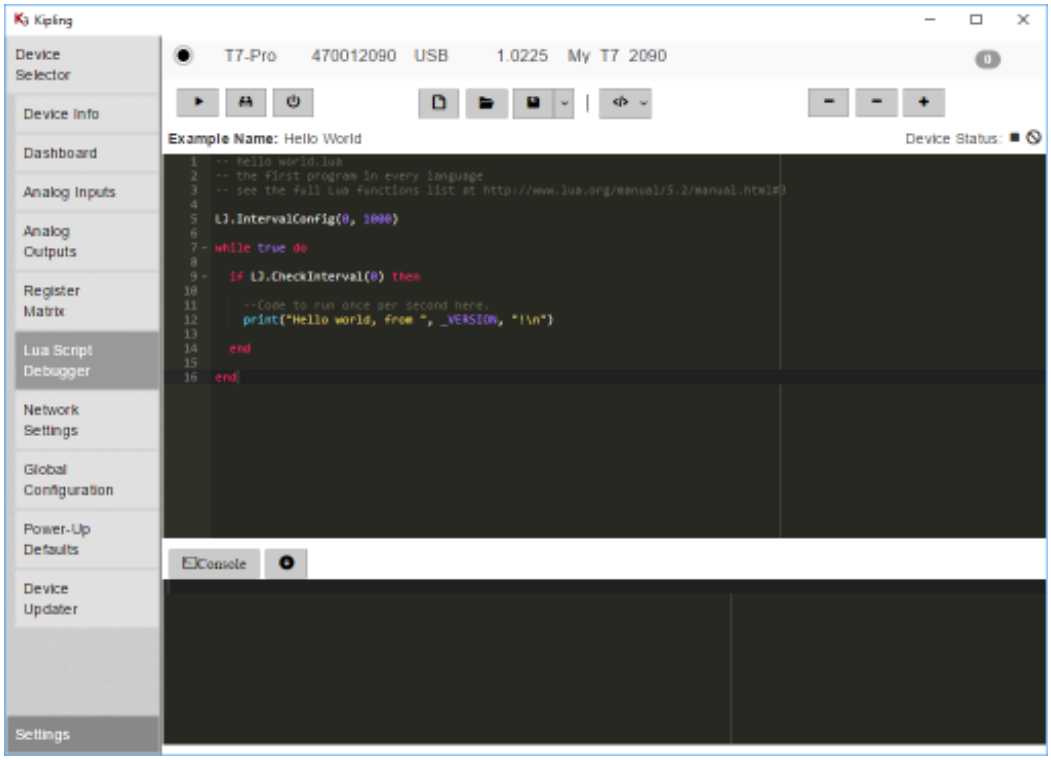
Published on *LabJack* (https://labjack.com)

Standalone Operation: One of the key advantages of T-Series Lua Scripting

Submitted by LabJack Support on Fri, 03/22/2019 - 19:34

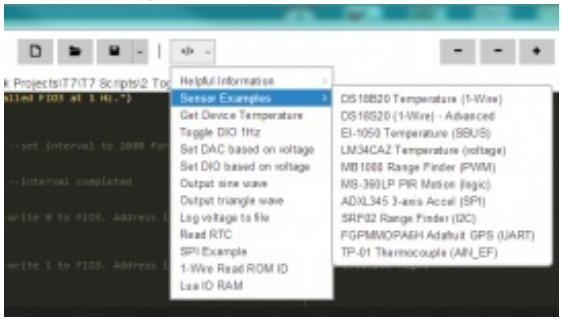


While running a Lua script, the T4, T7 or T7-Pro can operate without computer involvement. User-specified operations (feedback loops, logging, PID loops) can be conducted via an on-board script, which was not possible in previous generations of LabJack hardware. Below is a screenshot of the Lua Scripting section in our free cross-platform configuration program, Kipling.



Autonomous scripting operation is common for embedded devices like Arduino, but there are some key advantages of any LabJack T-Series, compared to other embedded solutions:

- Full access to all device features within script: (24-bit ADC, 14 analog channels, 23 digital I/O, 10 counters, I2C, PWM, etc.)
- If you require more I/O capability, we have a host of accessories that can be added without extra code or wiring.
- No code is compiled on the host PC, so there is no need to setup some kind of compiler/interpreter on your system. Simply send the Lua scripts to the device as a basic text file using Kipling (free, cross-platform), and the T-Series device returns all feedback, including print statements, compiler errors, and all other debugging information. This debugging information is shown in Kipling in the console, all you do is click the Run button.
- You get to write code in Lua, which can be easier to learn than C or C++.
- Multitasking: Any LabJack T-Series can be running a script, and also responding to external requests at the same time.
- There are dozens of simple examples built into Kipling, so you don't have to dig around to find example code.



- The Lua scripts are easy to write because scripts utilize the same high-level Modbus TCP address system as is normally used to access the device. Just look up the address in the Modbus Map and type them in your script.
- Scripts can utilize the T7-Pro's battery-backed RTC to trigger events every day, hour,

weekday, etc. Alternatively, use the RTC to write timestamps for each data point stored in your .csv files.
- Store up to millions of data points to theT7s 2GB microSD card in .csv files.
- It's very easy for an external computer to access variables churning around in a script through the use of a designated set of Modbus registers we refer to as USER_RAM.
    - For example: You have a Lua script that reads a temperature and sets a digital I/O based on the temperature, but you'd like to also save the temperature to a file on the computer or 'check-in' on the temperature from time to time using a Laptop. You can add this functionality with only 1 line of code in Lua:

      ```
      MB.W(46000, Temperature_ABC)
      ```

      This code updates the Temperature_ABC value in one of the RAM registers and then any external computer can read that stored value at any time, over any of the communication interfaces (USB/Ethernet/WiFi). Other embedded platforms typically don't have WiFi or Ethernet built-in, so users would need to add extra hardware for those platforms.

- Lets say you want to modify the operation of a T7 running a script based on some external event coming in over your computer (like getting an email or input from another device like a microphone). You could write a lightweight program on the computer which sends a signal to the script with USER_RAM.
    - For example: You have a PID script that performs the necessary actions, but you want to periodically change a setpoint using a Laptop or cell phone. Simply connect to the T-Series over WiFi/Eth and write the new setpoint value to USER_RAM0_F32.

Summary

On-board Lua scripting on T-Series devices enables powerful control and flexibility when compared to other embedded frameworks. On-board scripting combines well with features such as the T7's high resolution ADC, convenient communication interfaces, and availability of OEM versions.
