Project Proposal

Two-way Power Line Communication

Juan M. Falquez, jfalquez@gwmail.gwu.edu

Project Abstract

The device is a bridge between the serial connection of a computer and the power outlet in a home/building. Users will be able to send data through the power lines to another device connected at another power outlet. The communication protocol between the power line interfaces will be X-10, SNAP or a variation developed by myself. The idea is to use the protocol which would allow the highest throughput.

Strategy

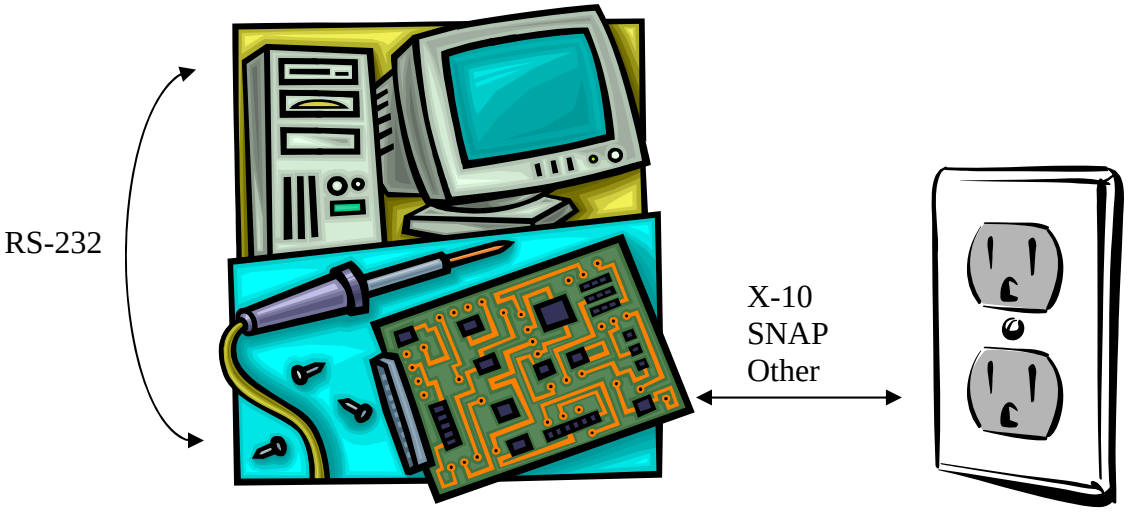

Figure 1: Hardware layout.

The bridge is based on the Z16 and using an external two-way power line interface. It will make use of GPIOs, timers and interrupts; buffer data from/to the serial port and the power line interface, as well as keep track of the zero crossings.

The software will include 3 modules: a serial port manager which will handle input/output on the serial port, a power line module which will handle input/output on the power line, and a central bridging module which will bridge/wrap serial data into the power line protocol of use (and vice-versa).

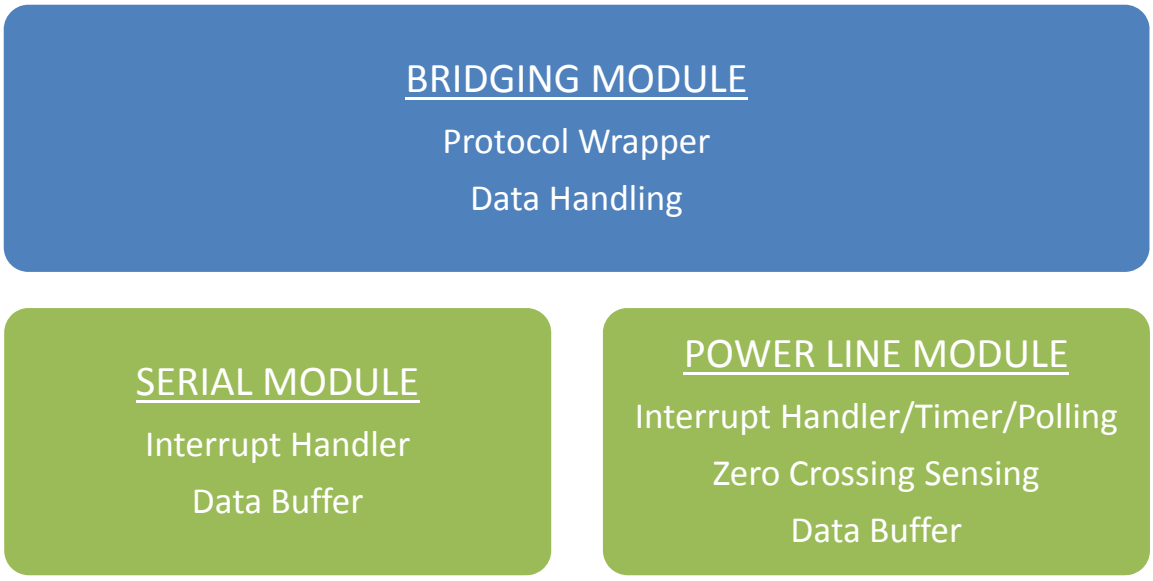

Figure 2: Software modules layout.

Unknowns

X-10 protocol has limited throughput, so an independent non-standard version of it might be used to improve performance. Implementing SNAP might be difficult since the power line interfaces might not easily allow data bursts in the zero-crossing.

Implementation Plan

Initially the project will start with native X-10 until a communication is established between the two devices. The bridge will take this serial data, wrap it in the appropriate protocol, and send it over the power line to the other computer. Thus, the input sent over HyperTerminal in one computer connected to the device will be seen on the screen of another computer connected with another device.

Afterwards, the protocol will be altered in order to improve performance. For instance, potentially removing some header, tails and address fields; removing some redundancies or sending 2 bits per cycle instead of one. Features removed from this layer might be implemented on higher layers in the protocol stack.

Finally, if time and hardware allows, native SNAP will be implemented or attempts to transmit burst of bits in each zero-crossing instead of single bits will be attempted. The throughput of each variation of protocol will be compared.