Z-Wave Bridge PCB

Contents

I needed a way to communicate the state of the doors and windows (being monitored by my home alarm system via hard wiring) to my home controller which uses Z-Wave devices.

I could have added new Z-Wave door switches to each door/window, however I chose not to for several reasons:

- My alarm wiring was done very professionally and all of the wiring is completely hidden inside the walls – so there are no visible devices. I wanted to keep this "clean" look.
- I did not want to have to maintain two sets of sensors for each door/window.

My alarm system has a serial link between the main alarm controller and the keypad/display over which the state of all the doors/windows is transmitted any time they change so that the display can show which doors/windows are open/closed.

This Bridge PCB monitors that serial communication and then activates the appropriate Z-Wave Door sensor based upon which door/window just opened or closed.

I know that I could have done this using a single Z-Wave interface and programming it to respond as a multi-switch device. However this was far more complicated as it required a full understanding of all of the Z-Wave protocol complexity, and to make things worse I was not able to actually purchase the Silicon Labs ZM5202AU-CME3R module from DigiKey – even though they show 1000 in stock they won't sell them unless you are buying thousands of them!

Components Required

Here is a link to the schematic: 76001530. and the BOM: BOM Link and the zipped Gerber files: Gerber Link.

Feel free to use the attached gerber file to order PC boards if you want. I grant permission for anyone to use this and place the PCB design into the public domain.

Here is a link to the Amazon page where you can purchase the Z-Wave door switches: Amazon Link.

Here is a link to the DigiKey page where you can purchase the Keystone Electronics 707 enclosure in which this PCB is designed to be mounted: DigiKey Link .

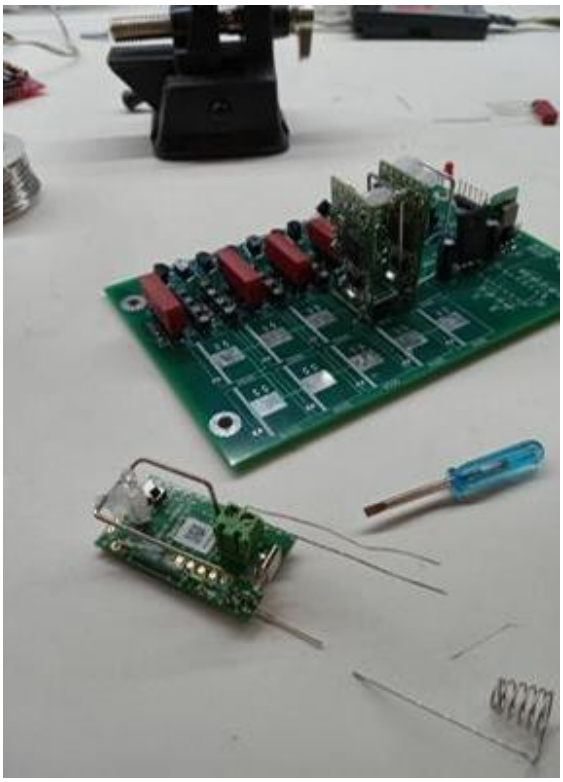
Assembly

Most of the assembly is very straightforward; however the Z-Wave door sensors require some clarification.

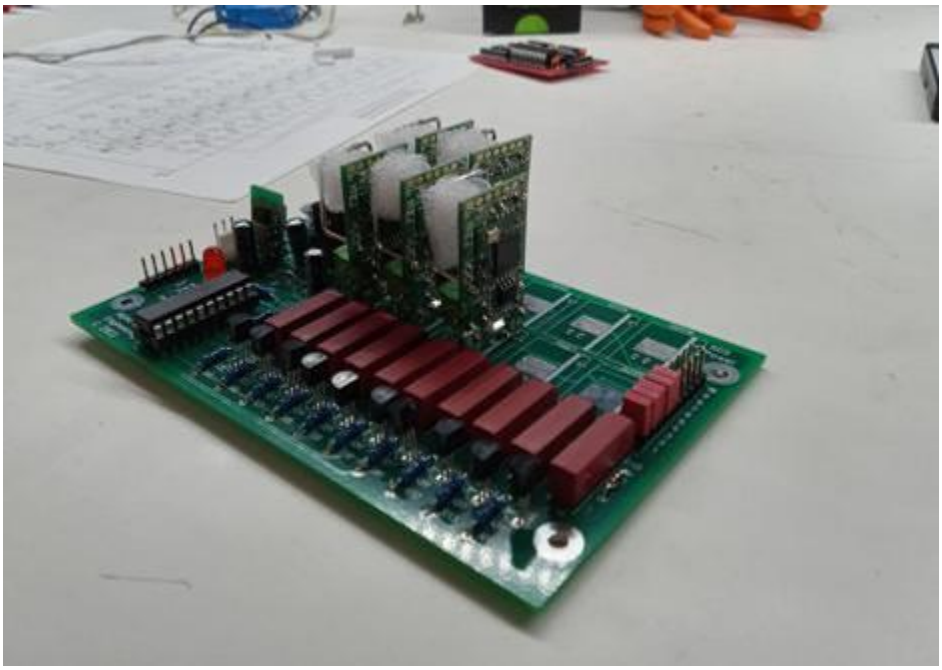
The board supports up to 10 Z-Wave door sensors – however you only need to populate as many as you actually need.

You need to discard the magnet (left in photo below) and open the plastic enclosure of the sensor (right in photo below.) You can then dispose of the battery – it will be powered by the PCB.

You need to clip the battery spring off, and insert two wires into the green external switch port (shown below) and tighten the screws until the two wires are secure. You will then insert this assembly into the PCB in such a way as the 3x protruding wires mate into the corresponding holes on the PCB and the metal battery tab is resting on the large square silver area of the PCB for that sensor. You must then solder the 3x wires in, and solder the metal battery tab to the PCB square silver area. Repeat for as many sensors as you need.



Once you are done (I used 5x sensors in this example) you need to cut out a small piece of foam for each sensor. This foam is inserted between the Z-Wave antenna bar and the “tamper” switch on the Z-Wave PCB. It serves the purpose of fooling the Z-Wave sensor into thinking it is still in its original packaging and has not been tampered with.



Pairing Sensors with Polisy Controller

Once everything is fully assembled, you need to “pair” the Z-Wave sensors with your Polisy controller. It is important it note that these MUST be paired one at a time – failure to do this will cause the controller to enter an error state which can only be resolved by power cycling the Polisy.

This is very simple because the PCB controls power to the Z-Wave sensors via the J5 header. You should start with no jumpers populated (no power to any Z-Wave sensors.) You will then enter pairing mode on the Polisy controller and add the power jumper for the next Z-Wave sensor. Once it has been paired, you simply repeat the process for each of the other sensors.

Source Code

You can download all of the source code for this project here: Source Code Zip File

This can be built using the MicroChip MPLAB IDE.