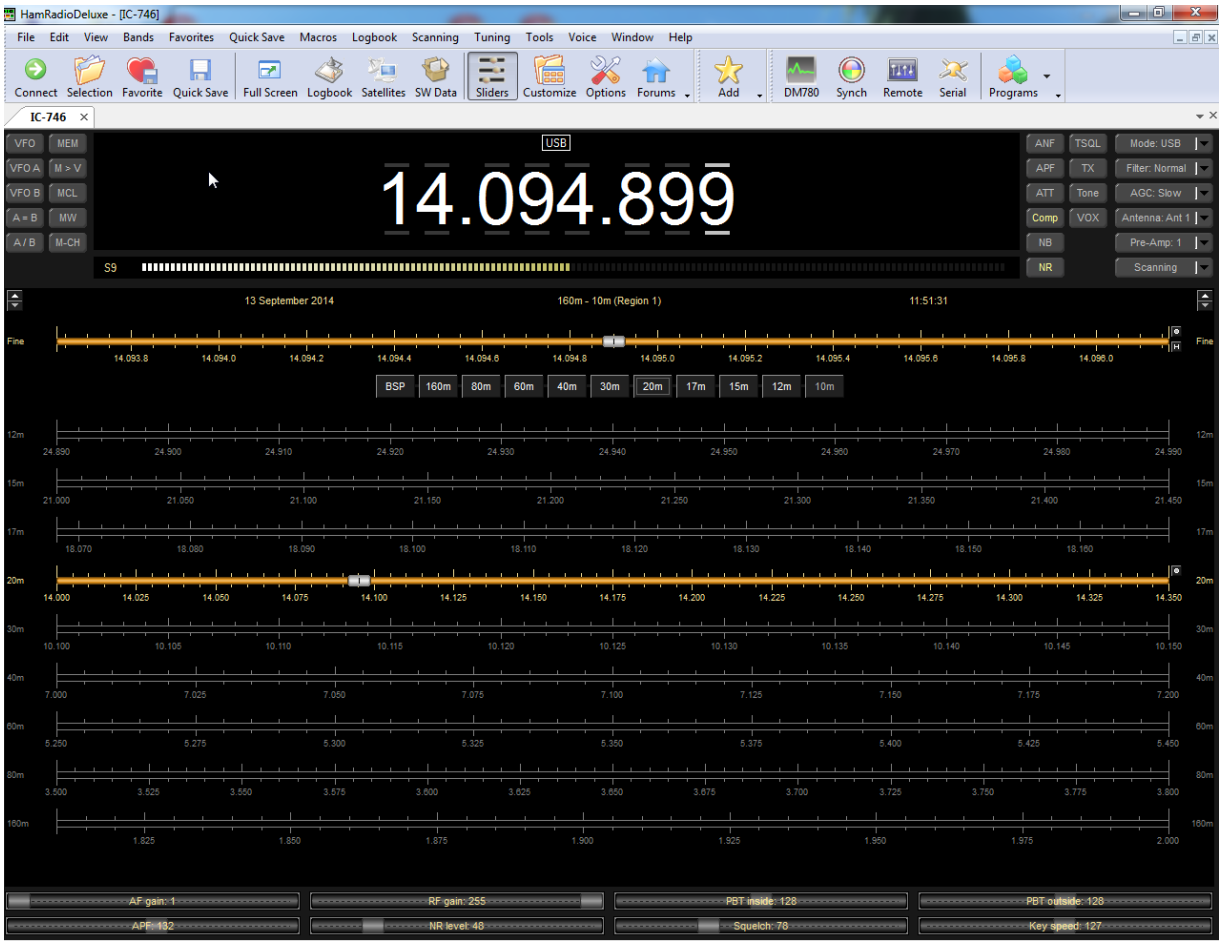
Powering up your PC (and transceiver) remotely

The basic concept

To 'boot-up' your computer from a remote location is a very useful feature. I needed to do this to use the HRD Software ('Ham Radio Deluxe': remote operation of amateur radio) effectively.

Initially I relied upon Wake Up On Lan (WOL). Although intended for local networks (LAN) it can usually be executed over the Internet (WAN).

It worked well, until I was using my home amateur radio station for 2m FM QSO. After several 'overs' my local computer (laptop) lost the connection via HRD. I had been in 'transmit' mode so was left not knowing if my home station was still in transmit mode… embarrassing!



HRD operating through an ICOM IC-746 transceiver

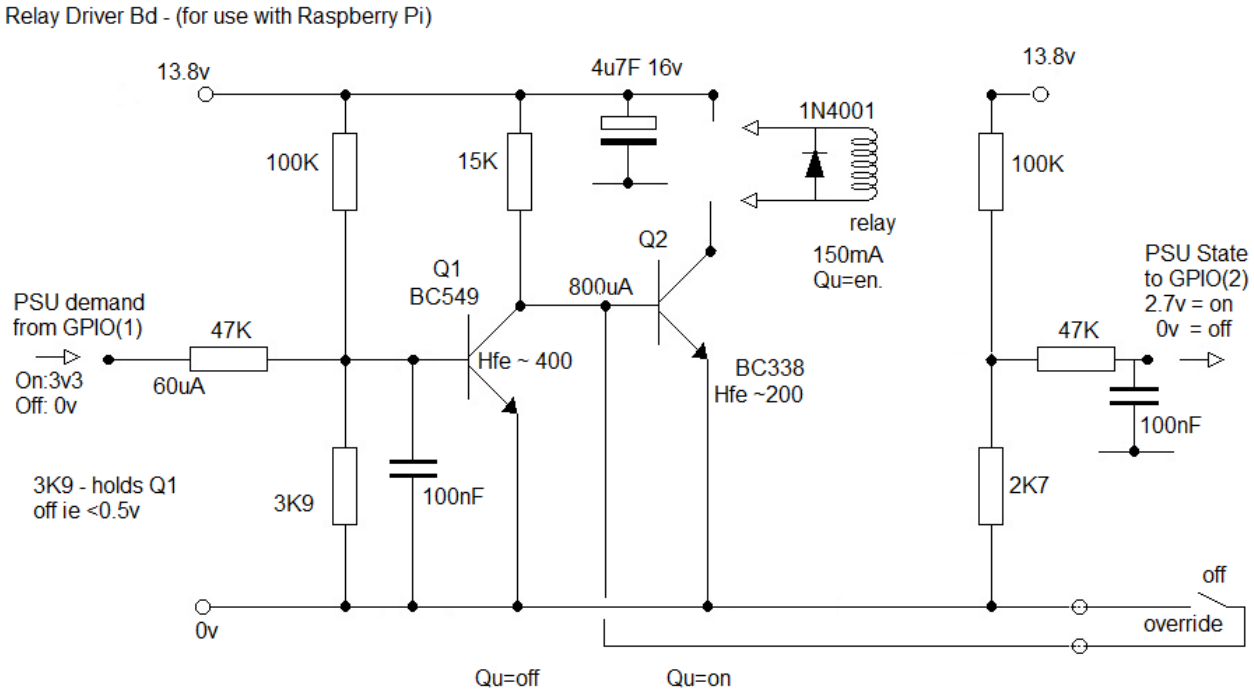
My first attempts at overcoming this situation was:

a. Install a webcamera to view the 'shack', so that at least I could see what was happening.
b. Install software to allow remote shutdown ('Airytech Switch-off' see: http://www.airytec.com/en/switch-off/)

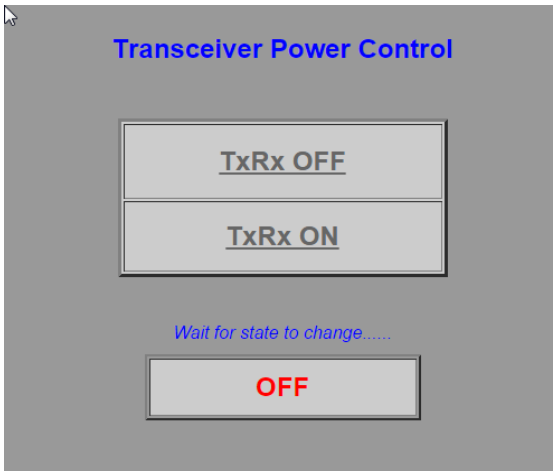
That was fine for the basic control of the computer, but wouldn't necessarily resolve a situation where the transceiver was latched in transmit mode.

The initial design

So the next phase was to utilise a Raspberry Pi (RPi) computer to remotely switch the power (12v) to the transceiver. It was part way established, as I was already using an RPi as a Web server (Apache software) and had a simple website up and running. This runs constantly (< 3 watts consumption) and had proved to be totally reliable. All I needed to add was a transistor switch (RPi GPIO buffering) and relay. A link to a webpage that offered a toggle on/off button, plus a couple of lines of PHP script which called an equally short code segment in Python.



This worked well enough, it also incorporated simple transceiver 'status'. This was achieved from a sample of the 12v transceiver supply (level changed using a pot-divider), connected to second GPIO (input) port.



This meant that the on/off status could be passed back to the webpage (Python to PHP code segments again) where it would be displayed each time the page was accessed or whenever a transceiver on/off demand was made.

But it let me down!

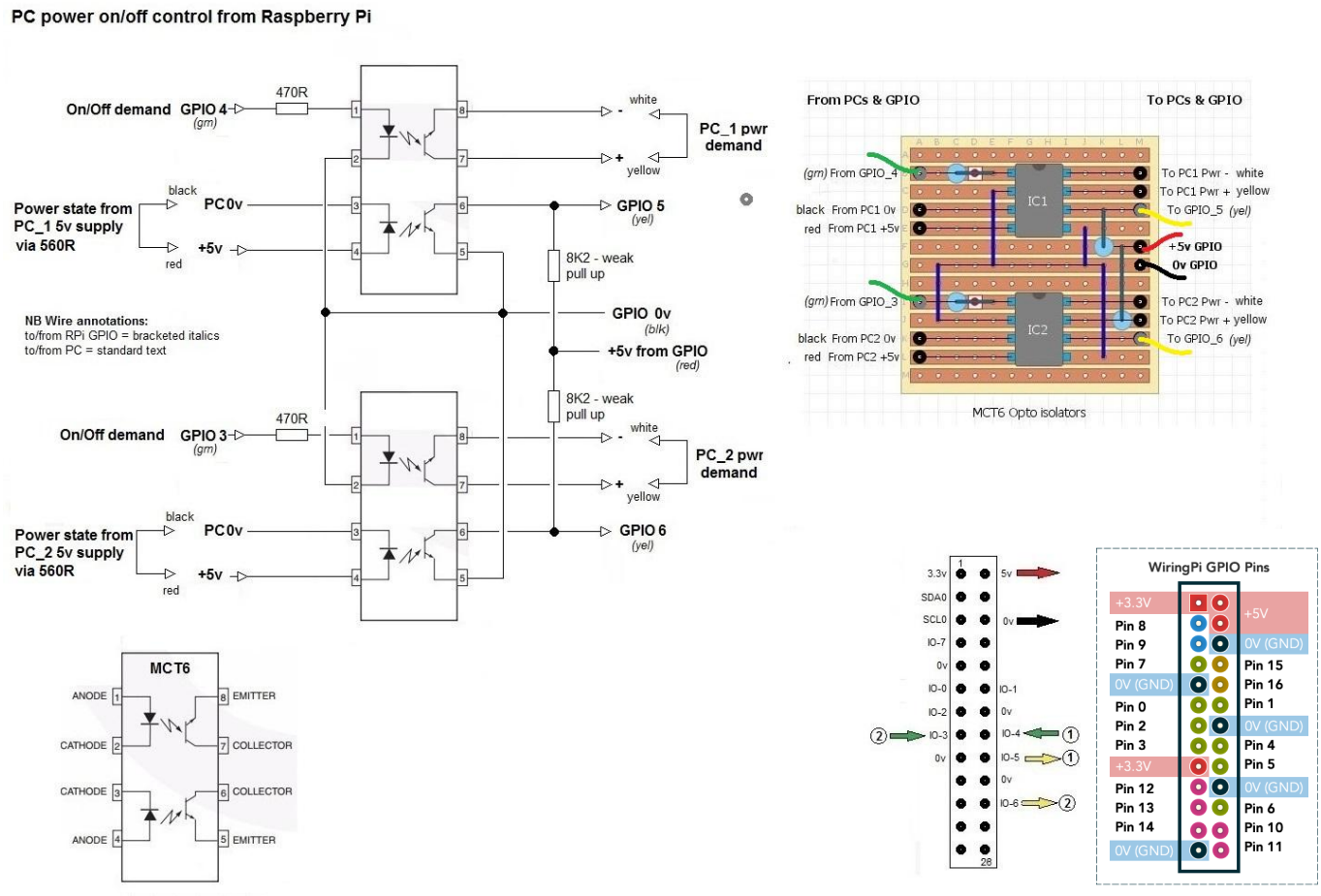
That should have been sufficient for operating the radio station remotely. However there were still two flaws in this theory. The first flaw, WOL cannot function if the computer has 'crashed' and stalled (a condition not unknown for a Microsoft OS) where only a hard-reset (ie full reboot) will resolve the problem.

The second flaw was realised when my ISP (Virgin Media) kindly upgraded my Internet connection from 20Mbps to 50Mbps. While very welcome, it required a replacement modem/router.

This was the 'SuperHub' a rebadged Netgear device with proprietary firmware that was designed to be easy to use and as secure as possible. One feature that had been removed was its ability handle the 'magic packets' used for WOL. This meant that the remote switch-on of the computer was no longer possible over the Internet (WAN).

A solution is found

Having used a Raspberry Pi to control power to a transceiver it seemed that it would be easy enough to power-on a couple of local computers. I really needed access to two PC systems, one that handles the security cameras and one that is used for general day-to-day functions. This time I used a couple of *opto-isolators* to route a 'demand power on' that was applied across the 'power on' pins of the computer's mainboard. The current state of the computer was sampled from the +5v (from a spare drive connector). I did try using the 'power-on' state from the computer mainboard, but the current was insufficient for both the front panel LED and the opto-isolator LED.



Having constructed and installed this hardware power control solution, I read an article suggesting that as WOL is designed to function reliably within a LAN, it could be invoked through a 'local' web server. As I already operated a 'local' web server based upon a Raspberry Pi running Apache Server software. Using this, I was able to route power control demands on the Internet to the local server. A PHP command from the website calls a simple BASH script file, this utilises the 'wakeonlan <mac addr>' command which transmits a 'magic-packet' to any nominated MAC address on the LAN.

It worked well, so once again WOL was a practical approach to controlling the computer systems power-on. The hardware power switching remained the best option for the computer that was connected to the transceiver, for it would overcome the possibility of a system-crash while transmitting.

That is pretty much it – but someone asked me what would happen if the RPi crashed ('A linux OS crashing – surely not!'). Anyhow I do have an answer, the RPi & modem/router are connected to a small Uninterruptable Power Supply (UPS). This is connected via USB to the main PC. It provides a software utility that includes UPS power on & off. Hopefully all power on/off aspects have been addressed. Well... apart from a mains power blackout!

Web page menus for systems control