GLOUCESTER AMATEUR RADIO & ELECTRONICS SOCIETY

G4AYM
G2HX

RSGB

WWW.G4AYM.ORG.UK

RAGCHEW

JUNE 2022

From the Editor

2022 is an historic year for the nation with the Platinum Jubilee of the Queen, and for GARES our **100th anniversary**, which we will be celebrating on **Saturday 13th August** at Down Hatherley Village Hall. It makes me feel very old that I've been a member for 50 of those years - but I was young when I joined!

In this issue **Tony G4HBV** continues his series **"A Brief History of Radio"** with Part 7 giving an overview of the role of radio communication in the RMS Titanic disaster.

At a recent club meeting **Vernon G0HTO** passed me some details of a You-tube video clip of a very rare German valve which is essentially a complete TRF radio within the glass envelope, requiring just an external tuned circuit and loudspeaker. https://www.youtube.com/watch?v=Afzd8nbkqdo

At the same meeting **Graham G8DLW** told me that he has recently acquired a **Heathkit HW30**, more commonly known as the **"Twoer".** I featured this rig in the May 2021 "Ragchew" Vintage Column which can be viewed on the club website in the Library / Ragchew section and I look forward to an HW30 to HW30 AM QSO!

Andrew G4IVD has given me details of a link to the East Anglian Film Archive and the film is entitled **"Wartime Radio: The Secret Listeners".** This 30-minute documentary is well worth watching https://eafa.org.uk/work/?id=1041744

At the club **"Bring and Show"** meeting held a while ago, **John M0NRZ** brought along the various iterations of his **CI-V project**. In "**Evolution of a Multi-Purpose Arduino Based Controller"** he describes both the hardware and software challenges he faced. At the same meeting, a good number of members brought along items of interest and I've managed to make up a page showing the wide range of equipment displayed.

After some six years of editing and producing "Ragchew" I have finally taken the sad decision to step aside and hand the reins over to a successor - hopefully someone in the club will step forward and pick up where I have left off! I still have plenty of archive material filed away to appear in future issues. Thank-you to all the members who have supported me by submitting articles and I take this opportunity wish my successor all the very best!

Best 73 and Stay Safe

Brian G4CIB

Contest Round-Up

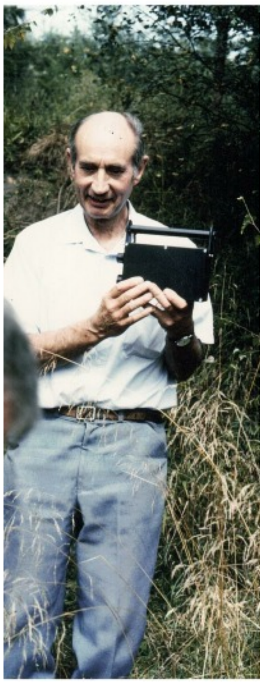
A small group of club members continue to support the club entry in the **VHF/UHF UKAC** series of contests. The club is in **23rd position** in the **Overall Table. Gary M0XAC** heads the table by a good margin, followed by **Tony G8JAY, Nick M0NYY, Barry M0HFY** and **Graham M0XGL.**

In the **80m CC (Club Championship),** the club is in **9th position. Mike M4N (G4IZZ)** leads the table, closely followed by **Martin G4ENZ.** Logs from **Bob M0NQN, Gary M0XAC, Tony G4CMY** and **George M0HWT** help to add to the total score.

If any newer members need any help in getting started in contesting, speak to any of the above members who will be more than happy to assist you.

From The Archive

DF Hunt - Beechenhurst 1992



L: Alan G1IFF gets a fix on the Top Band Fox.

Below: Brian G4CIB deep in the undergrowth takes a bearing on the Fox

Vernon G0HTO operated the 2m station along with **Tony G4HBV** the Top Band station.

The hunt was won by **Brian G4CIB** who went on to win the DF Trophy

A Brief History of Radio – Part 7
by Tony G4HBV

We jump now to 1912 for the sinking of the RMS Titanic. I am only going to deal with the radio aspects of the disaster, for the whole story is one of ineptitude and obscuration, right up to the British enquiry into the loss.

In those days, the radio equipment and the operators on the Titanic were supplied by the Marconi Company and the ship's callsign was MGY. Ice warnings were heard by the Titanic's operators from several ships on 11th April. On 14th April Titanic was in range of the station at Cape Race, Newfoundland, but because of a failure of the Titanic's transmitter the day before, the operators were wholly occupied with passing personal messages from passengers.

The last ice warning came from a nearby ship, the Mesaba, but it seems, in the backlog of passenger messages, that it somehow was ignored. Just before midnight on 14th April the collision occurred. An ice warning from a nearby ship, the Californian, stopped in the ice-pack, was brusquely cut short by the Titanic's operator shortly before the collision. After further attempts to warn the Titanic, the wireless operator on the California switched his set off and went to bed. Commercial pressure on the Titanic's operators had destroyed their chance of avoiding the disaster.

47 minutes after the collision, the Titanic started sending the distress calls of CQD and SOS. Several ships nearby and the Cape Race station received these signals. The Carpathia 58 miles away put on all speed towards the Titanic's reported position, but of course the Californian, only 10 miles away with its radio closed down, heard nothing …… and so the disaster became unavoidable.

A Peculiar Problem
By Brian G4CIB

Following on from **Vernon G0HTO's** article "A Peculiar Problem" in the last "Ragchew" I can now report on a peculiar problem I encountered a year or so ago during lock-down with my 10 watt TS120V rig. This has been a good old workhorse, indeed in 2001 it was the rig I used on our annual trip to Lundy, with a sloping dipole antenna slung down from the top of the old lighthouse. I hadn't used the rig for some time and decided it needed to be powered up to check that it still worked. Connecting it to my doublet antenna fed with open wire feeder and suitable matching unit all seemed in order. Without any warning the output suddenly dropped to zero. Surely I couldn't have blown the PA transistors? Feeling a bit miffed, I put the rig aside to be worked on at some future date. Looking around on the internet I found an article by **EI3FEB** who had purchased a non-working TS120V and set about restoring it. One thing that did strike home was that the PA transistors were fairly difficult to get to and would require some skill to replace. Time to put it away and go and do something a bit easier! This was one occasion where the club misses the expertise of **George G7GQC.** Well, a few months ago George and Jenny turned up to one of our meetings and I explained the problem to him. He suggested that it would be worthwhile investigating the main relay in the rig (designated RL1) which not only switches the antenna but also the various power rails between receive and transmit. It's an Omron relay and like the PA transistors, a pig to get to. It has a clear plastic cover but is not sealed, so I tried a squirt of switch cleaner in the general direction of the relay. For a brief time the transmitter gave full power before dying. So at least the PA transistors were still OK! Further searching on the internet, I discovered a web site **UK Vintage Radio (Repair and Restoration).** Several posters on the website suggested I try a product called De-Oxit to spray on the relay contacts. Vernon tells me he had a similar problem with oxidised relay contacts and using a small hand-held modeller's drill in one hand and the nozzle of a vacuum cleaner in the other hand he managed to drill a small hole in the relay cover to enable him to squirt the cleaner on to the relay contacts.

Evolution of a multi-purpose Arduino based Controller

Introduction

I retired from full time employment in mid 2016, and had decided to add Ham Radio to my other hobbies (electronic design & writing code for micro-controllers). Fortunately, my career (which spanned Radio and TV equipment Maintenance, Television Broadcast Engineering, and CCTV System Engineering) had adequately prepared me to tackle the RF side of Ham Radio, so the learning curve for the License exams was biased towards the regulations and operating etiquette and a refresh of RF principles. By December that year I had acquired a pass in all 3 classes of exam (adopting the callsigns M6NRZ, 2E0NRZ and M0NRZ), and had purchased the "new kid on the block" - an Icom IC 7300 HF Transceiver.

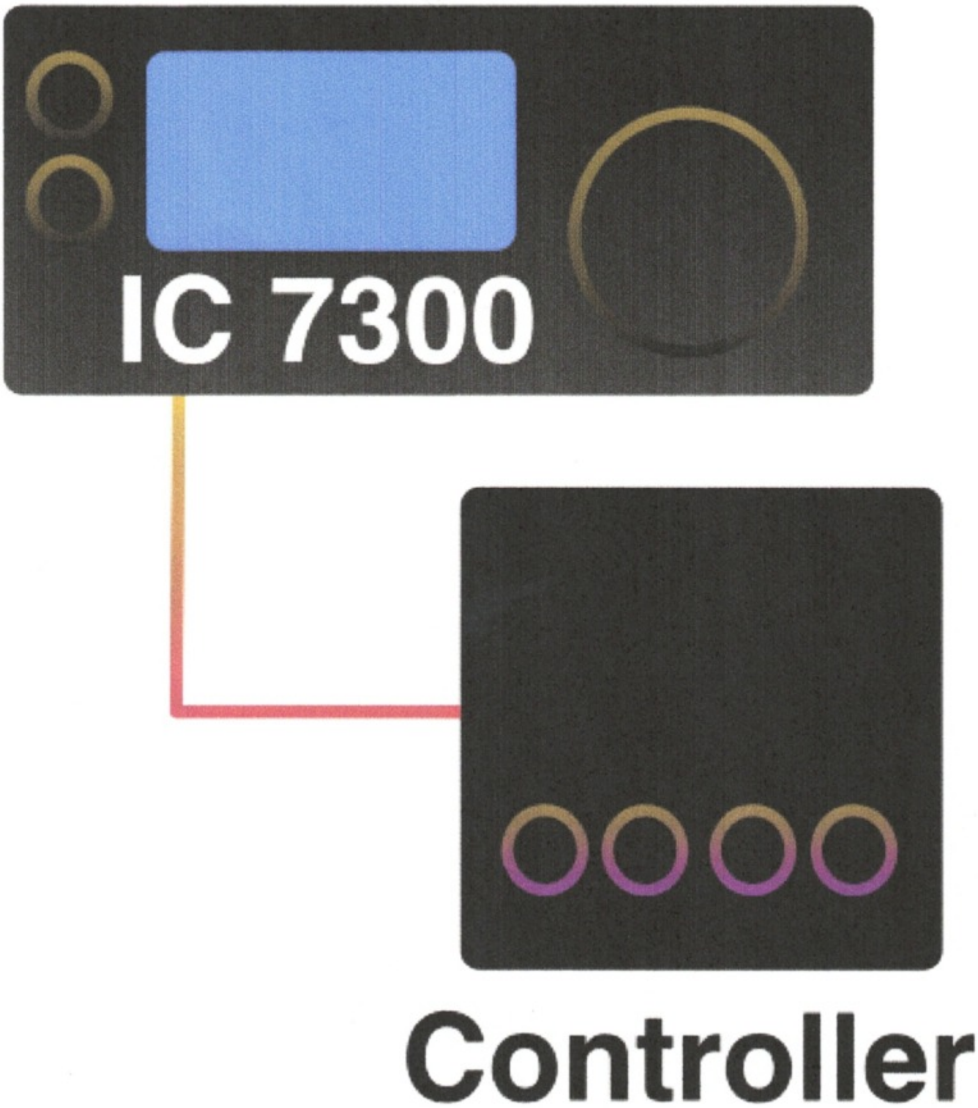
It is a great radio. The designers have done a brilliant job cramming so much functionality into such a small volume, but for me, there lay a problem... I felt that the decision put Mode change and Band change within the touchscreen menu system to be an operational annoyance.

Early in 2017, after a little research into the remote control "CI-V protocol", I wondered if perhaps I could remedy this. There were commands available which implied that I could perhaps have some physical buttons or switches external to the radio, and use them to trigger the sending of CI-V commands. One of my hobbies (from about 6 or 7 years earlier) was tinkering with Arduino micro-controllers, and I'd learnt to programme using 'C' about 20 years before that. The Arduino's version of C / C++ was ideal for me to tackle this project : a blend of hardware and software (and a little luck).

But that project had to wait (for about 18 months) while other tasks took priority.

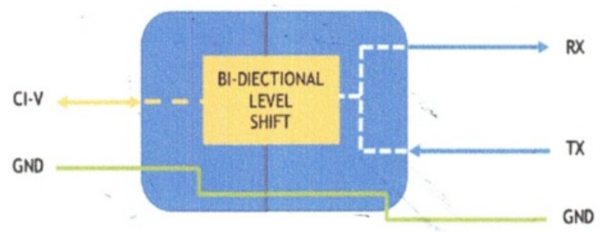
First Principles : the hardware

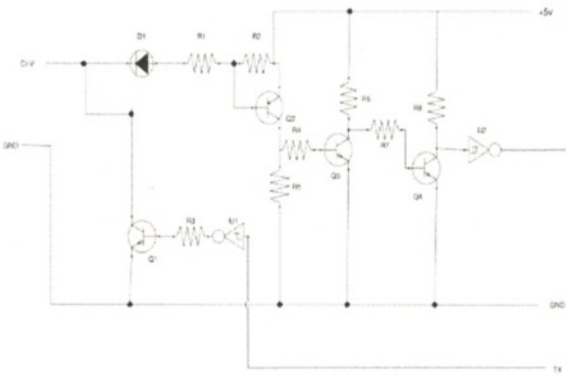
In early 2018, I was at last in a position to pursue this project in earnest. The plan was fairly simple : build a proof of concept device – basically a small bi-directional circuit to convert TTL (5v logic) RX / TX signals to/from the CI-V bus via the Remote port of the 7300, and then develop the control software to allow the controller to send commands to change Mode or Band.



The CI-V protocol is a little like 10base2 ethernet, all attached devices have unique addresses and all share the "one wire" for both RX and TX. With a Google search, I found a few home brew CI-V solutions but they were fairly simplistic – in particular with regards to the "messaging" software *. The hardware too was often just two resistors and a diode.

Bi-directional Level Shift

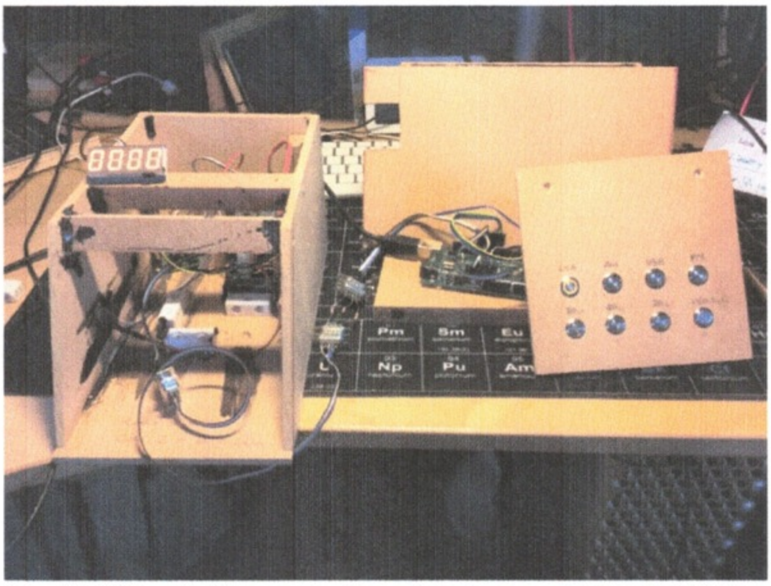




Instead I chose to more or less replicate the hardware within the IC 7300 – a few transistors plus a couple of Schmitt inverters - initially assembled on a perf board (actually I built two), and later several custom PCBs**.

With a interface CIRCUIT built on Perf Board, using a MEGA 2560 Arduino I sent an appropriate CI-V message (although not fully adhering to the protocol) and it worked - I COULD change Modes! I might have stopped there, but for a chance conversation with a fellow Ham, I described what I was doing and he asked *"Can you get the S-meter data from the radio via the Remote port and display it ? " Jeff (W6FCC -SK).*

Proof of Concept



Late 2018 Proof of Concept - (Pro mini micro-controller & "MDF enclosure"):
8 buttons : 4 for Mode change, 4 for Band change.

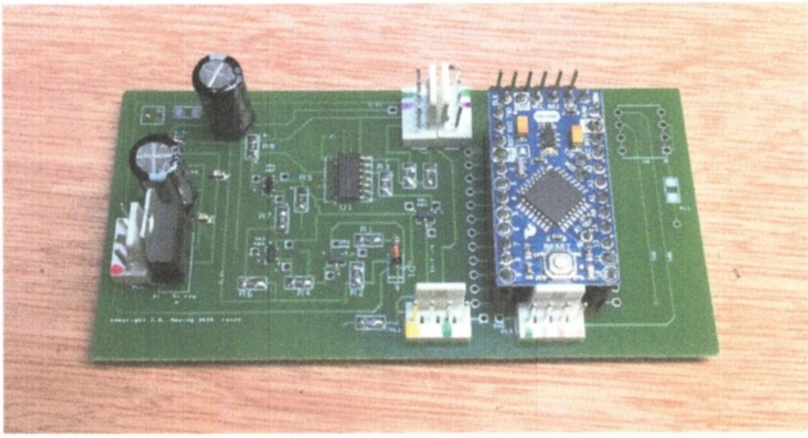
The Printed Circuit Board

I needed to build at least three devices, two for my radios (I had acquired an R8600 receiver which from an operational point of view has a similar reliance on interaction with it's touchscreen!), and one controller device for my friend (Jeff - who wanted an S-meter too).

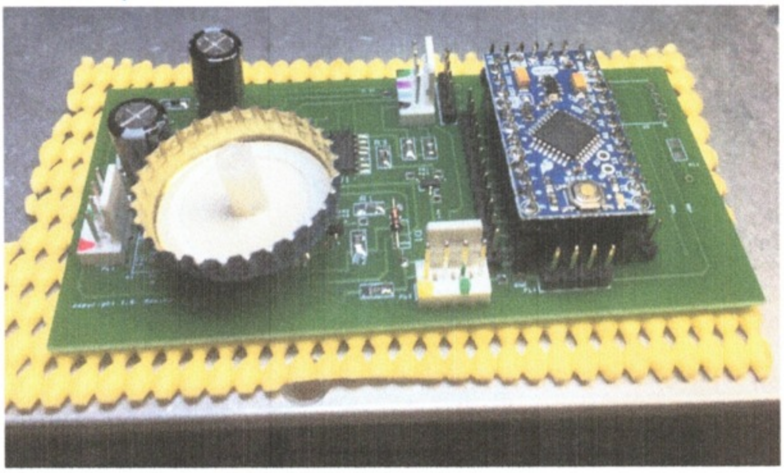
It was a slog and a half building it with Perf board. Also, if anyone else wanted to make one, I thought that a PCB would be a great starting point for them.

I had previously only designed a couple of small "through hole" PCBs, but (if I used SMD components - a first for me) I could squeeze TWO identical track layouts on to one 100 x 100mm "standard size" board. The first board was junk – it had so many errors – too many to fix by hand. The second version (Type 1b) worked

Pro-mini board (99 x 49mm) Type 1b



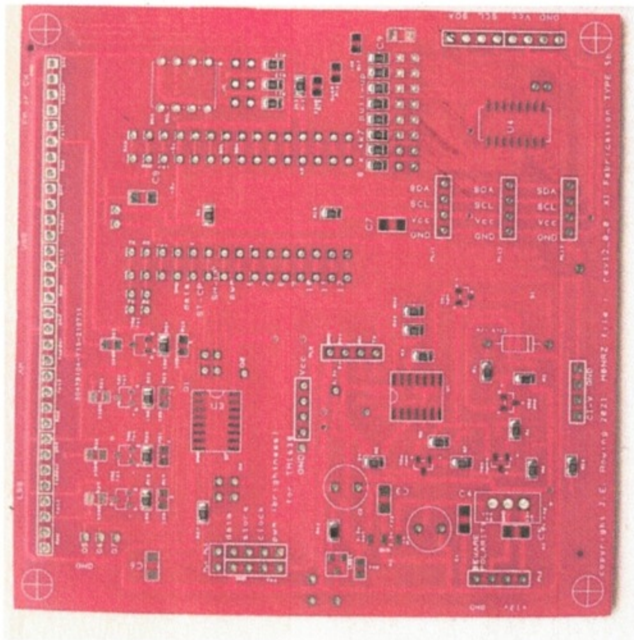
Necessity is the Mother of Invention



Running with 13.8v input, the 5v regulator needed an impromptu heat sink. What better than an associate of the stuff that cooled me down too.

After a few months of software development (more about that shortly) memory space was running out, and it became necessary to change the board to host an Arduino Nano EVERY.

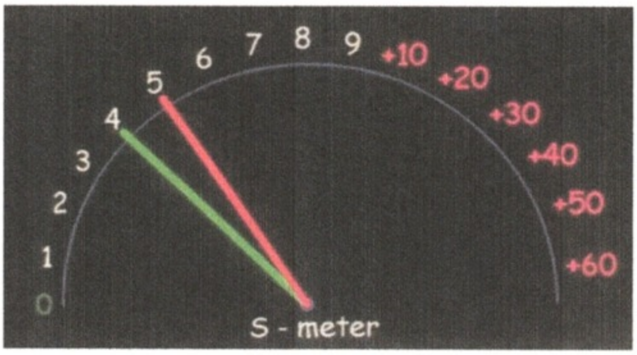
Nano EVERY board (99 x 99mm)



Also, by increasing the board size to 99 x 99mm, this was also an opportunity to consolidate several hardware enhancements, notably the Resistor Ladder (to detect Buttons pressed), INPUTS to read the state of 8 "boot configuration" header pins, and OUTPUTS (via N-channel FETs) for a number of internal relays if required.

Finally, connections were provided to drive the separate RGB LED "S-meter Board". Through 2020/21 there were several more tweaks of the component layout – and I got to "Type5b" ☺.

S - meter



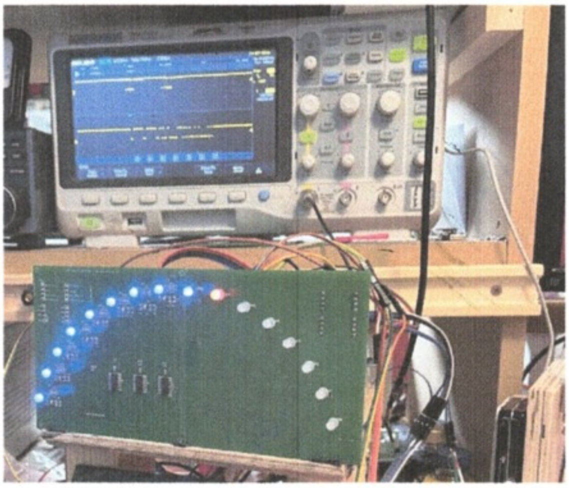
A few weeks after Jeff's request, I had a demo where I could get the data from the radio and display it on a graphical S-meter on the screen of a PC or Mac.

However, I imagined that my Controller would not have a screen, but instead have a few buttons and LEDs.

This was the turning point in the design. Most S-meters have a scale indication S1, S3, S5 etc up to S9, with "pips" for the in between values (S2, S4, S6 etc). I decided to construct an ARC of LEDs, on a new PCB, actually to get from S1 through to S9+60 would require two adjacent PCBs.

I also use colour to imply a value, the S4 LED is made Yellow for meter values S4 to S6, and the S7 LED is made Violet (actually Magenta) for S7 to S9. Inspired by the Resistor Colour Code, these colours help to indicate a meter value without requiring an adjacent scale.

RGB LED S-meter



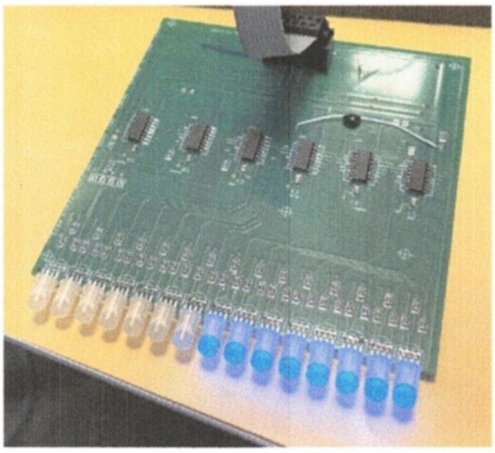
The lower expanded trace on the 'scope is showing the S-meter data being received from the radio. This is converted by the software in the Controller into an "index" of 0 to 15 which then lights the appropriate number of LEDs, in this case "10" to show S9+10.

Sadly Jeff became silent key and never saw this come to fruition.

Linear S-meter

Yet another LED PCB design... this one permitted either 16 in an arc or 16 in a line near the board edge. Then I realised that I didn't need 16 LEDs, only 15 were required to show S1 through to S9+60.

Comact Linear S-meter



Squeezing all the 45 tracks to drive the LEDs meant that this PCB has 4 layers.

Coincidently, Serendipity played a hand. I had bought an enclosure, which happened to have a groove on either side to hold a PCB, my 99mm x 99m boards fitted perfectly. I ditched the four buttons and added a rotary encoder. I also found a button which had an illuminated (RGB !) collar.... Throw in a Laser cut front panel, and somehow everything came together !

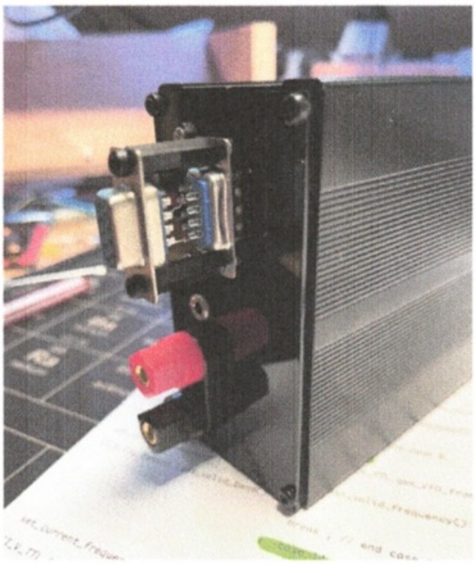
Early 2021: Single Control Button (with an RGB collar) and Rotary Encoder in a shell type case and compact 15 RGB LED layout is current Prototype.

Compact Controller

This is an earlier version of the Compact Controller, being tested with an optical rotary encoder. It was later found that the less expensive mechanical types are preferred.

Yaesu too

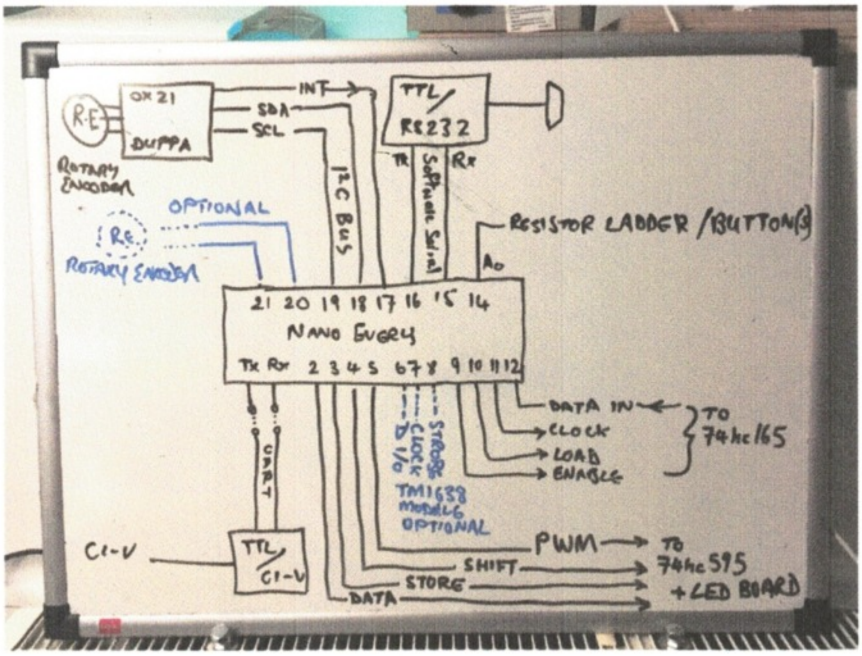
Early 2022: Dual independent Icom / Yaesu control.

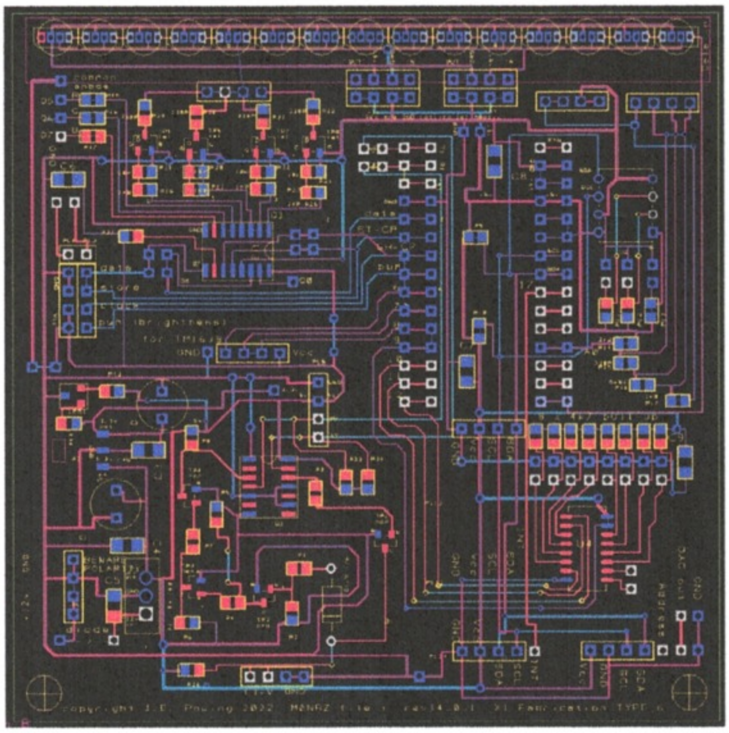


By adding an RS 232 port, and additional software, the Compact Controller can also use CAT protocols with YAESU radios.

The Current Hardware Roundup :

The 2 layer Control PCB hosts an Arduino Nano EVERY micro-controller. Almost all of it's pins are in use (or allocated).





This is almost the end of the story of the hardware development, except that a new controller board is being tested, it is capable of driving Neopixel type RGB LEDs, and they sit directly on the board.

The CI-V interface connected to Pins 0 and 1 (UART pins) has remained virtually unchanged. A second Software Serial method via Pins 15 & 16 connect to an RS232 module allowing Yaesu CAT control too.

Pin A0 connects to a resistor ladder capable of resolving up to 4 physical button inputs, although currently only 1 button is used. This is the method used to select the Controller device "State" – or more precisely it determines the function initiated by rotation of the Rotary Encoder. Each "short press" of the Control button moves the device control state through : 1Hz VFO (Yellow), 500Hz VFO (White), Mode (Magenta), Band (Blue), Squelch (Cyan). The (colour) is presented on the outer collar of the Control button to indicate the current "State". A long press of the Control button places the device in (Black) State, where a kind of direct frequency entry is possible – MHz, 100kHz, 10kHz and 1kHz adjustments are selected by pressing the Rotary Encoder's integral push button. Finally, a double press of the Rotary Encoder's push button will invoke (Green) State where control via the CI-V (Icom) port, or the RS232 (Yaesu) port can be selected by turning the encoders knob.

Pins 2, 3, 4 and5 feed the RGB LED PCB with PWM (brightness control) Clock, Strobe and Serial data.
Pins 6, 7 and 8 are allocated to an optional TM1638 Keyboard / LED display module.
Pins 9, 10, 11 and 12 are connected to an input expander shift register which read the condition of 8 configuration header pins.
Pins 18 and 19 are connected to the I2C bus which (via an interface module) read the Mechanical Rotary Encoder rotation rate and direction and push button events via a DUPPA interface. Pin 17 returns a rotary encoder event interrupt.
Pins 20 and 21 are unallocated but available to connect to an optical Rotary Encoder.

The Software :
I'm an amateur (hobby) programmer, familiar with 'C' principles and new to "C++", however the latter (in particular OOP) has been very beneficial as I incorporated many logical objects with in the code. Here's an object list, the function of some are obvious :

Controller Object : glues together the interaction of other objects

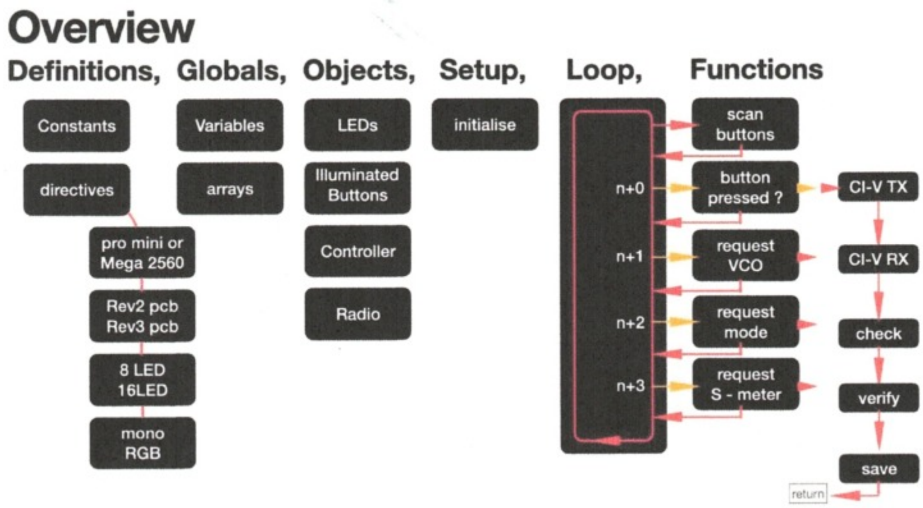
CI_V Object : assembles commands to instruct the Icom radios, request data, checks message integrity, extracts payload data, converts to / from message tokens and internal variables.

RS232 Object : similar to the CI_V object but for Yaesu radios, overall less complex functionality.

Radio Object : a virtual representation of the connected physical radios, mostly a container for the current valid VFO, Mode, Band, Squelch data.

LEDmeter Object : logical interface to the physical RGB LEDs

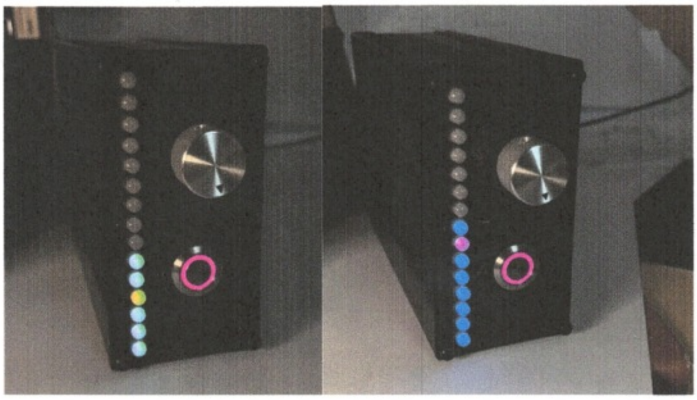
The Main Loop requests from the connected radio the current value of specific attributes, these are : S-meter, VFO, Mode and Squelch. The Request Messages sent to the radio are interleaved and some are contextual to the current State of the Controller, i.e. if in the Magenta Mode change State, the Mode info request message is sent more frequently than the request for VFO data.



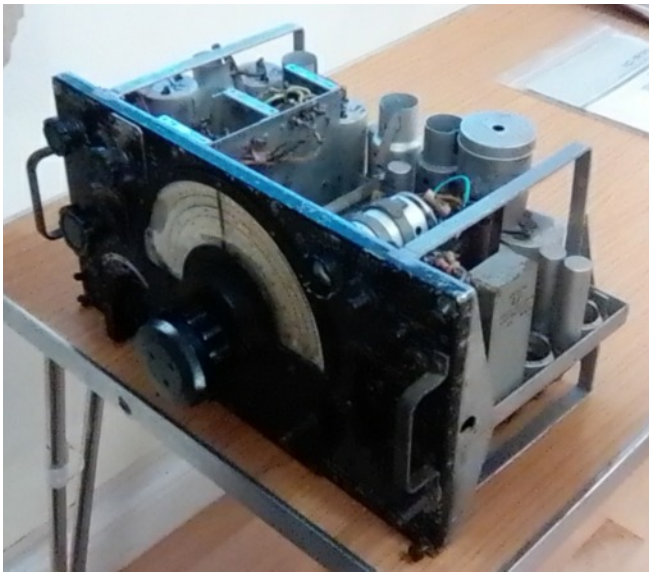
The requested data in the Reply Message is received as Tokens, i.e. Icom sends VFO data as 5 bytes, where each byte contains two Binary Coded Decimal numbers which are also arranged in an unusual pattern. The CI_V object converts these tokens into an internal variable (of type Unsigned Long) and sends it to the Icom Radio Object. Now stored as 'real numbers', the value can be manipulated. E.g. in (White state) rotating the Encoder will add or subtract 500Hz, and a new value for the VFO can be re-tokenised, and sent back to the radio to "SET" the physical radio's VFO. Yaesu sends VFO data as 10 ASCII tokens.

The S-meter data is requested most often (about every 300ms), the value (0 to 241 for Icom, 0 to 255 for Yaesu) is converted into an internal "index" (0 to 15) to then illuminate up to 15 RGB LEDs. In Icom mode the first 9 LEDs are blue for S1 to S9, the 6 LEDs for S9+10 and beyond are red. Yaesu mode uses white for S1 to S9, and blue for S9+10 through to S9+60.

The "scale-less" display uses colour to help interpret the displayed S-meter value. Inspired by the Resistor Colour Code : the S4 LED is Yellow for S4 to S6 and then the S7 LED is Violet (actually magenta!) for S7, S8 and S9.



There is nothing magical within the software, although there is a lot of it. Including "white space" it's over 11,000 lines, about 120 pages of A4. At least half of it is the code which defines the Objects. Both the hardware and Software are quite mature. The PCB's would benefit from one more iteration in order to make assembly of the device a little simpler and add dedicated pads for the RS232 module. In my limited experience, software is rarely 'done', but it's stable and probably close to the point of diminishing returns.



R1155 receiver - one of Jim 2E0GKN's collection of vintage receivers.

10GHz wideband FM transceivers by G4CIB

Homebrew Software Defined Radio (SDR) by Ken G3LP

John M0NRZ's extensive collection of hardware related to his C-IV project as featured in his article in this issue

Spring Time Operating at Crickley Hill

Anne 2E1GKY enjoying 2m