GLOUCESTER AMATEUR RADIO & ELECTRONICS SOCIETY

G4AYM
G2HX

WWW.G4AYM.ORG.UK

RAGCHEW

JANUARY 2023

From the Editor

A very Happy New Year to all GARES members and families. On the run-up to Christmas a good gathering enjoyed the social at Down Hatherley Village Hall and many thanks to Rita M6RYL for organising the buffet.

In this month's issue **Mike G6OTP,** after dabbling with digital TV has gone back in time and is now investigating crystal sets. As he explains in his article, the purchase of a vintage Lissen coil at a flea market set him on the quest to re-visit his youth. He has set the members a challenge to build a crystal set and hopefully at some stage there will be a "Crystal Set Bring-and-Show" event at club. I guess you can approach it from two directions, either using original vintage components such as a "cats whisker" detector or modern devices. Either way, it will make an interesting project!

Cliff G8CQZ examines the pros and cons of the **Raspberry Pi, the PIC, Pico, Nano and Uno.** So far I've only ventured as far as using a Raspberry Pi as a general shack computer mainly with Amateur Radio programs. Are there any other GARES members using any of these devices? If so, how about submitting an article!

Malcolm G6UGW has written to say that it helps to be plane crazy when thinking about radio aerials. The top section of the full-size G5RV measures 102 feet. The G5RV is the same size as the wingspan of an Avro 683 type Lancaster, the World War II heavy bomber. Another aircraft which can be used is the Aerospatiale/BAC Concorde which at 204 feet long is twice the length of the G5RV. The Russians built a similar type of supersonic aircraft, the Tupolev TU144, nicknamed Concordski. Think of the 2m band in megahertz and you have the right type number. If you want to compare Concorde F-BVFB and the TU144, they are on show at the Technik Museum of Transport at Sinsheim, Southern Germany. Here is the link to their web site: https://sinsheim.technik-museum.de/en/aviation

That's all for now, and hope you will take up Mike G6OTP's challenge and build a crystal set. Useful when the power goes off and the batteries in your portable radio are flat!

73 Brian G4CIB

The next issue of Ragchew will be early April. Submissions to g4cib@outlook.com by mid-March.

Book Review

"Chip War" by Chris Miller

Having spent some of my working life during the late 1970's and 1980's employed by a Californian-based semiconductor manufacturer, I looked forward very much to the publication of this book. The history of the semiconductor industry makes for fascinating reading and this book doesn't fail in this respect. The author seeks to establish how we have arrived at the situation we find ourselves in post-Covid with global shortages of semiconductors and sets out the chain of events. In the early chapters Miller presents the previously well-documented immediate post-WWII history of the invention of the transistor by Shockley, Bardeen and Brattain at Bell Labs also the saga of the "rebellious eight" who left to set up Fairchild Semiconductor and are credited with the founding of Silicon Valley. Through the 1960's and into the mid-1970's the area exploded as new companies were set up to satisfy not only the military market but also the growing consumer market. It was still essentially an analogue world where at the high-end of the market, precision was the key so there was no pressure to increase the number of transistors on a chip. With the rapid growth of digital technology in the early computers, it was recognised that computing speed was everything and the pressure was on to increase the number of transistors on a silicon chip. Indeed in 1965, Gordon Moore of Fairchild predicted that component count would double every year such that by 1975, integrated circuits would have some sixty-five thousand transistors on a single device. It was realised that to finance wafer fabrication plants which were able to process the ever decreasing device geometry sizes would be a huge financial burden on the many Silicon Valley companies. And so the concept of fab-less companies designing integrated circuits and sub-contracting the wafer processing to offshore foundries such as TSMC in Taiwan was born. One notable rebel was Jerry Sanders "the Rolex-clad, Rolls Royce-driving brawler who founded AMD, who liked to compare owning a semiconductor fab with putting a pet shark in your swimming pool. Sharks cost a lot of money to feed, took time and energy to maintain, and could end up killing you." Jerry believed in the saying "real men have wafer fabs". But even for him, the writing was on the wall. The last half of the book looks in great detail at the explosive growth of the far East as a major global power-base in the manufacture of semiconductors, the impact of technology transfer, the rise of Huawei in China and the strategic importance of Taiwan to Western economies. In summary an invaluable addition to my library.

A Winter Project

by Mike Rainbow G6OTP

My first venture into radio was to build a crystal set to a design in the junior edition of one of the national newspapers. For some reason, sometime in the mid 1950's, some of the daily papers brought out weekly junior editions which ran for about six months or so. In those days, you could buy all sorts of things in the High Street including DCC* copper wire and a cat's whisker or failing that from the small ads at the back of the papers on a Saturday. I can still see ads for mysterious things called rotary converters, easily converted to a workshop motor- they said. Well, the crystal set was duly assembled, aerial and earth connected and headphones donned. Nothing happened, not a peep. Nothing could bring it to life but it did serve one useful purpose as the start of my junk box.

I will get to the point soon - honest.

To cut a long story short, just recently I found a vintage Lissen coil, in a flea market and thought, I will make that crystal set after all, even though it will be nearly 70 years late.

So here is the challenge.

To make a crystal set from whatever we can find these days. There are plenty of circuits on the internet and quite a lot of books available. It will make a change from high tech stuff like FT8, SDR and other stuff like DMR. Back to basics and an introduction to construction for some. Going from back to front, you will need a pair of high resistance headphones (don't even think of 8 ohm.), or a crystal earpiece, or even a 'speaker' insert from a GPO* handset. A fixed capacitor and maybe a resistor depending on which of the above you are using and of course, the detector. The easy option is to use a germanium diode but it could be anything: galena crystal, a rusty razor blade or even a 'real' cat's whisker. A variable capacitor of about 500pF max but whatever value you can get and last of all, the coil. This will have to improvised. Toilet rolls are not what they were but they can be stiffened up with a wrap of masking tape. You can also use plastic piping from such emporia as B&Q. The classic antenna was a long wire with an earth or counterpoise but a ferrite aerial could also be tried.

There is life life after Maplin. A very good source of small components is Harding's Electronics in Cheltenham's Lower High Street and I have a small number of spare diodes and resistors and a few variable capacitors.

Over to you. This is not a competition but there might be a small prize.

*DCC - double cotton covered.

*GPO - General Post Office (forerunner of British Telecom)

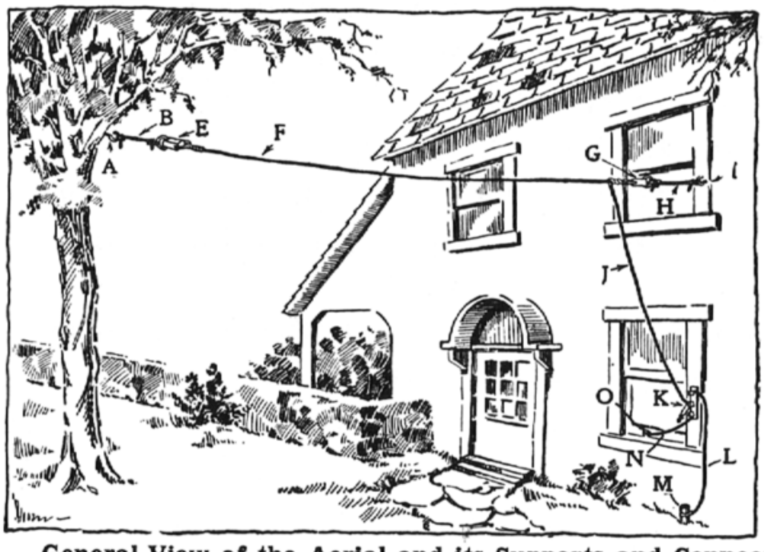


Fig. 1.—General View of the Aerial and its Supports and Connections.

Illustration of a typical early radio receiving aerial from "Crystal Receiving Sets and How to Make Them" by Bernard E Jones ("Amateur Wireless" editor) and published by Cassell, c1923.

This book was re-printed in 2000 by Lindsay Publications Inc.

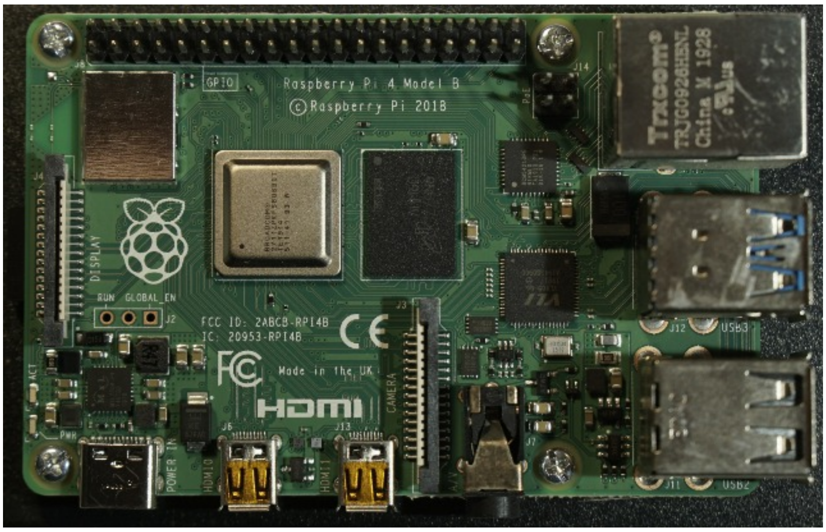
PI, PIC, PICO, NANO, UNO - A Personal View

by Cliff G8CQZ

I am a constructor rather than an operator of Amateur Radio equipment and I have seen, for example, Variable Frequency Oscillators (VFO's) go from coils and variable capacitors to Direct Digital Synthesiser chips and computerised rigs. Over the years, I have tried a variety of ways to control my projects. This is an outline of some of my more recent experiences.

Raspberry Pi

When I first saw the Raspberry Pi, the credit card sized computer, I thought that it looked ideal as a controller for some of my projects. All those pins that could be individually controlled with both digital and analogue inputs and outputs, built in protocols like I2C and SPI that could be used to control synthesiser and other chips plus USB, video and audio ports made it seem like a dream.



As a basis for experimentation the Pi was great but I only found one project where the Pi was suitable for running the project continuously: that was as the music controller for the model fairground at Tewkesbury Museum. For all other projects, the Pi really needed either a keyboard and screen or a link to another computer as a minimum. Indeed, it was as a computer that the Pi came into its own. Simply add a screen, keyboard, mouse and a portable hard disk and I had a fully functioning system

I have one that I use as a development web server and one that I keep configured as a software development and office application (Word, excel etc.) machine plus at least two others in the cupboard that I can pull out whenever an experimental machine is needed. I particularly like the fact that it runs on Linux and I can program the Pi in "C", which is my favourite computer language. Add-on's such as "GTK" can also be used to give programs that windows-like graphical interface. In addition to the wide range of software add-ons that are available, there are also a number of hardware "hats" (interface cards for linking the Pi to touch screens, relays etc.) and things like cameras that are available ready built so the Pi is a great machine for experimentation. I am also aware that other club members have done a lot more with their Raspberry Pi's than I have.

PIC

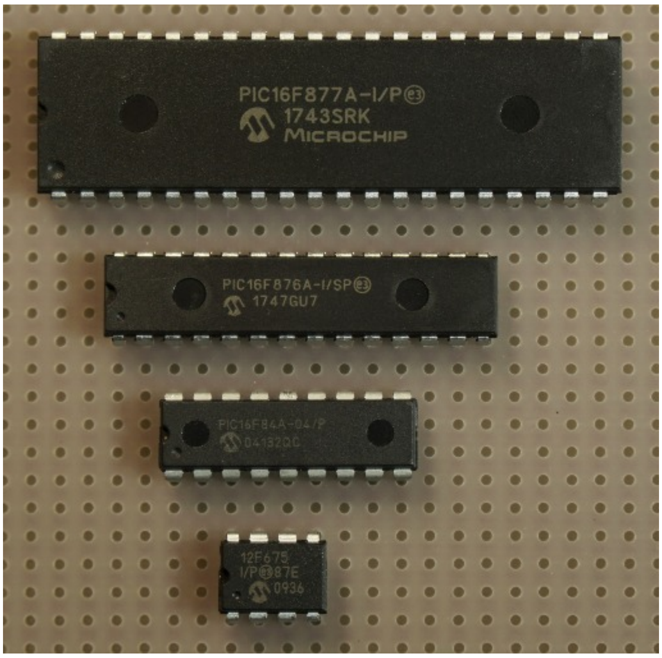
The PIC is the antithesis of the Pi: no screen connections, no keyboard connections and no operating system. It's just a chip and it even needs a separate interface card to program it. Although "C", and other language, compilers are available, I prefer to write programs in assembler. This is because the PIC only has a few kilobytes of memory and also because, in many of my projects, timing is critical. PIC chips are available with 8 through to 40 pins and the larger ones have built in I2C and SPI protocols although they have to be programmed - nothing is "ready to use" with a PIC. Unlike the Raspberry Pi, which can be programmed directly, the programs for PIC chips are written and compiled on a PC, laptop or Mac (using software that is available from the manufacturer of the PIC or other sources). The resulting code is then loaded onto the chip using the interface card. This can be done either by unplugging the chip from the project and into the interface card (which is the method I prefer) or by adding programming connections to the project that allow a link to the interface card. As there is no display or other monitor, it can be very difficult to find out why a project is not working and so I often need to edit, recompile and re-program the chips many times before success is achieved.

I have used PIC chips on a variety of projects including VFO controllers (in conjunction with DDS chips, rotary encoders and alphanumeric displays), aerial switches and PWM (Pulse Width Modulated) motor speed controllers.

The main disadvantage of using PIC chips is the speed (max 20MHz), small memory and the difficulty of learning to program them.

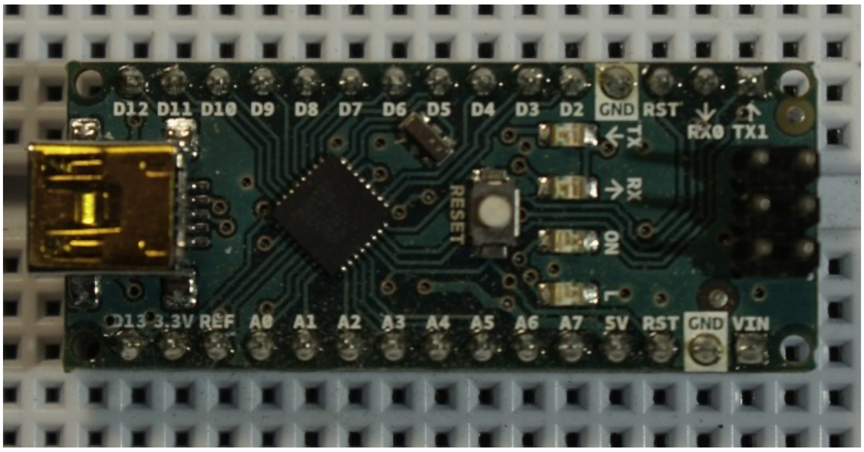
The photo (right) shows a selection of PIC ICs ranging from 8-pin through to 40-pin devices.

(Editor's note - Cliff is being very modest as he is the author of "PIC Basics" published by the RSGB in 2006)



Arduino UNO and NANO

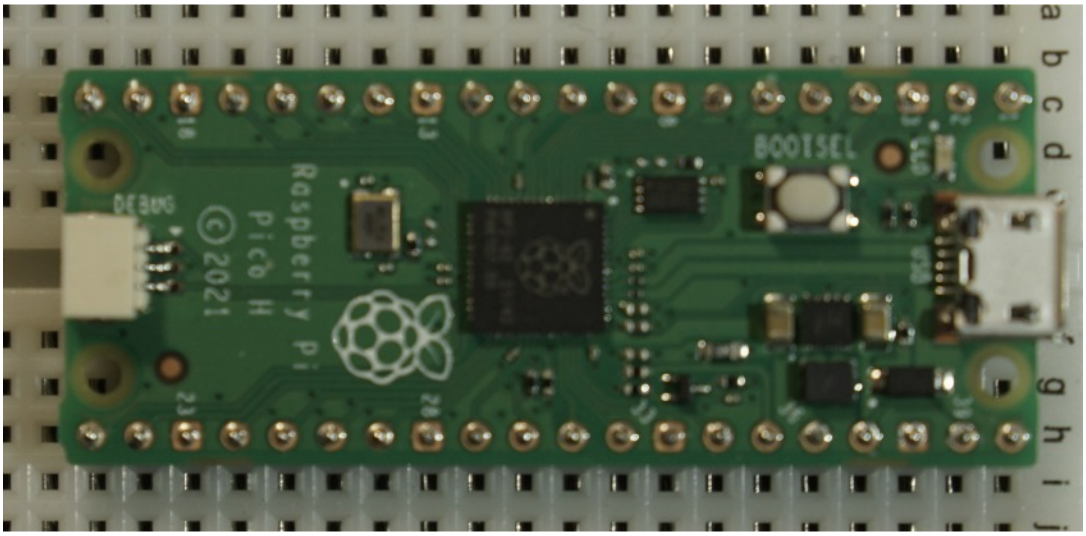
Arduino were, I think, the first company to take a PIC-like chip, add a USB interface and an operating system to make an experimenter friendly board. Various 'flavours' of board are available including a number of compatible boards from other manufacturers. I think that the two most popular boards are the Uno and the Nano. The Uno is 69 x 54mm and has connectors that rise above the board. It is ideal for use with "hats" that sit above the Uno board or with breadboard type cables.



The Nano is only 45 x 18mm with pins that are ideal for putting the board itself into a breadboard, Veroboard or a printed circuit board. In addition to the one Uno that I have, which is driving my CNC milling machine, I have a couple of Nano boards. Although I have experimented with these I have yet to find a project that really needs one.

Like the PIC, Arduinos have analogue and digital input and output pins plus inter-chip protocols like I2C and SPI. Software for the Arduino is developed on a PC, laptop or Mac using Arduino's free IDE (Integrated Development Environment): a single program to write, compile and download the finished program onto the Arduino. Although the programming language is basically "C", it has been simplified in the IDE to make it easier to learn. For example, instructions are placed either into the 'setup' routine or the 'loop' routine. As the name implies, setup is where things are initialised and the loop automatically loops without the programmer needing to do anything. There is no need to use the supplied routines and the board can be programmed directly in "C". The major advantage of an Arduino, over a PIC chip, is the USB connection and inbuilt operating system which, whilst transparent to the user, allows the board to be programmed and the program to be monitored from the computer. This makes software development so much quicker as there are many logic faults can be seen as they happen. The USB port can also be used as a data interface to pass information in both directions between the computer and the board.

This allows, for example, a frequency to be passed to the board (or read from it) as a decimal value whilst letting the board pass the appropriate hexadecimal code to the DDS chip. Although I have never done it, completed Arduino programs could also be programmed into chips (like the PIC) for building directly into projects. Unfortunately, like the PIC, the main problems with the Arduino are speed and memory. Most Arduino boards run at just 16MHz and both the Uno and the Nano have just 32KB of program memory



Pico

Currently, this seems to be the answer to my needs although I have not been working with it very long. The Pico is definitely a controller board like the Arduino and not a computer like the Pi but it runs at 125MHz and has 2MB of memory. In addition, the processor is dual core. I only currently have the Raspberry Pi Pico although I am aware that Arduino also have a Pico equivalent - the Nano RP2040. Like the Arduino, the PI Pico has a USB port for programming, monitoring and communications. Unlike the Raspberry Pi computer, the Pico only has one USB port and nothing else apart from the interface pins. There is a version with wi-fi but I do not currently have any of these. A Pico board is slightly larger than an Arduino Nano at 52 x 21mm.

The Raspberry Pi people would like you to program the Pico in Python using their ‘Thonny’ IDE but a “C” compiler is available (which I use) and it can also be programmed from the Arduino IDE by downloading a small add-on. All of the software is, of course, free. In addition to the high speed dual core processor, large memory, digital and analogue input and output pins, I2C and SPI etc., the Pico also has several smaller processors that can offload some of the more repetitive work of the processor. For example, it can take the load of driving an alphanumeric display (dividing the characters into nibbles, creating strobe pulses and waiting for acknowledgements).

At the moment, I am quite enthused about the Pico although, so far, I have only experimented with it. I wait to see how it performs when I try to build one into a real project.

The Future

I am well aware that newer chips and newer boards are becoming available. Examples include the STM32 and the ESP32. Some users say that these are even more powerful than the Pico but, whilst the Pi, PIC and Pico continue to fulfil my needs I will stick with them.