In this issue

The CQ-DATV editors gratefully acknowledge all those authors that have contributed articles for this free magazine.

Production Team

Ian Pawson G8IQU
Trevor Brown G8CJS
Terry Mowles VK5TM
Jim Andrews KH6HTV

Contributing Authors

Jim Andrews KH6HTV
Trevor Brown G8CJS
Roger Paskavan WA0IUJ
Mike Stevens G7GTN

Editorial

Welcome to CQ-DATV issue 81 of our electronic ATV magazine.

Another rabbit out of the hat as Ian always says and what a rabbit. Micro Corner is back with Mike G7GTN looking at BASIC for the Ardunio and the ESP 8266. Seems this programming language will never go away, instead it just keeps evolving, even though it was born in the 60's. It is the resident language of all the home computers that evolved in the 70 and 80's. Some of us are showing our age and mis-spent youth and never stopped using it, but lets not mention any names.

While on the subject of the 1970's Trevor has produced another of his looking back at this decade to mark 50 years. It's interesting to look at some of the idea's which did not really become household names, but only because they were perhaps a little too far ahead of the available technology E.G. the electric car and the electric bike which actually appeared a little bit later than the 70's and now e-bikes have reappeared and a whole range of electric cars are under development and may well be what we will all be driving by the middle of this new decade, lets wait and see.

Roger Paskavan WA0IUJ has written about everybody's favourite subject "mathematical equations". Perhaps we all now wish we had paid more attention to these lessons in school.

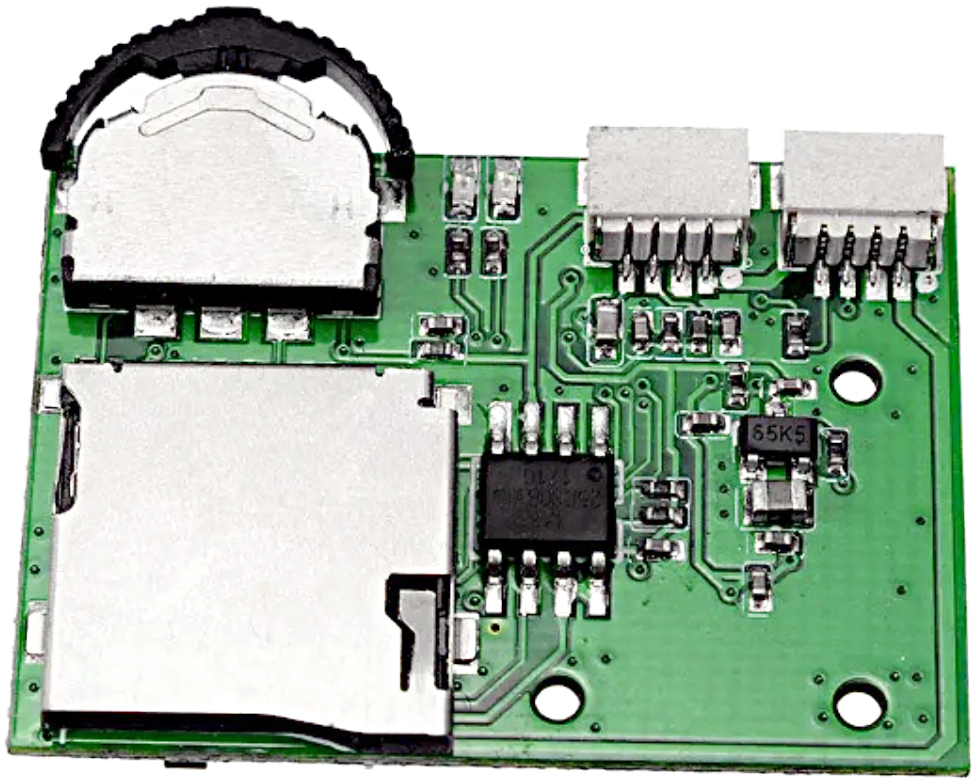
Roger points out that most amateurs dread the thought of doing logarithmic calculations. Roger goes on to say "it is so easy that you don't even need a calculator, it can be done all in your head"!! Pleased somebody kept awake during these lessons, now he will help all the rest of us catch up.

Trevor has written part 14 of his GVG mixer and joined forces with Mike G7GTN. Together they have downloaded Vmix, the most popular ATV video switching programme and together are determined to interface the GVG mixer panel to this software. For all of you that are using this software in point and shoot mode, they have investigated the short cut key menu and the MIDI interface link and have this talking to a Ardunio Pro Micro which is I2C controlled. The next stage will be connecting the GVG I2C control to the Ardunio, if anybody can do it they can, so watch this space as we progress the GVG panel to Vmix control.

What happens when two transmitters need to share the same aerial? Well a T-piece might be a start but if you want the RF to go up to the aerial and not to the other TX, this needs a little more work, but fortunately Jim Andrews KH6HTV has all the answers. Its all down to feeder lengths and is helped by the two transmitters being at different ends of the 70cm's band.

One from the Vault looks back at CQ-DATV 5 where Richard Carden explains some free software that allows you to create your own testcard. There has been a lot of discussion on social media about how to generate a Jpeg file for use as an electronic test-card. Richard's solution is far more elegant than the Power Point recommendations that were banded about. The problem was the link he provided no longer works, but a search through Richards hard drive has provided the software and its now available on the *CQ-DATV (TCM.zip)* downloads page.

I know we are all watching the new DVR recorder and hoping we can programme it from a .jpg file and this might be the answer. Several of us have ordered this unit, but due to China's export delays at the time of going to press none have arrived so we will have to wait for the next issue to see if it can be used as a programmable test card generator.



Inexpensive DVR recorder (watch this space)

https://bit.ly/2I1BL0H

CQ-DATV Production team

Please note: articles in this magazine are provided with absolutely no warranty whatsoever; neither the contributors nor CQ-DATV accept any responsibility or liability for loss or damage resulting from readers choosing to apply this content to theirs or others computers and equipment.

TV Amateur is a German Language ATV Magazine.
It is published 4 times a year.
Please note the website is currently off-line.

Tech giants are lined up behind the 'royalty-free' 4K codec AV1

Usually, when a new video compression scheme arrives it takes so long before there's a combination of compatible hardware and content that its arrival is anticlimactic. Google, Apple, Netflix, Microsoft, Intel, Amazon and a host of other companies have taken steps to make sure AOMedia Video Codec 1.0 (AV1) moves a bit faster, and for several good reasons.

Today they're teaming up to announce the release of a 1.0 specification for AV1, and get the ball rolling on support for video creation, distribution and playback with the new format. With AV1, the group promises it can deliver 4K UHD video using 30 percent less data. That's great for high-res streams and should provide plenty of savings even for those on older hardware – expect faster video startup time, less buffering and better picture quality.

As we move into the era of 4K video, HDR, VR and so many other forms of content, new compression technology will be required to make the most of each connection and keep every pixel looking its best. On the business end, these tech giants are trying to get around licensing fees and patents that have added to their costs and complexity of rolling out the tech currently needed to move HD and even 4K video around, like h.264 or the new HEVC.

With AV1, they claim all the relevant patent holders are already involved and have agreed to make its use royalty-free, so there won't be the licensing pool problems seen last time around.

Google got the ball rolling with its VP9 codec years ago, then turned planning for VP10 into this team effort. Because it's been in the works for over three years already, and it has support from both the hardware side (Intel, Apple, ARM, NVIDIA) and the software side (Microsoft, Mozilla, Google), the hope is to get this technology rolled out quickly, with support in web browsers popping up first. As Microsoft exec Gabe Frost put it: "From browsers and operating systems to devices, apps, and gaming, AV1 will drive the transformation of higher-quality 4K UHD video and beyond for more screens and devices."

Then, content makers and distributors can jump in, with YouTube, Netflix and Amazon all ready to push the new format and make sure it has hardware support in next-gen devices like game consoles, TVs and set-top boxes. Netflix's director of streaming standards Mark Watson said: "We're pleased to see the broad interest in decoder support and excited to bring the benefit of better streaming quality to our members in the coming year."

Netflix today (6th Feb 2020) announced that it has started to stream video using the royalty-free AV1 codec on its Android app.

YouTube, which started testing the AV1 codec in September 2018, currently allows users to stream in the royalty-free codec at up to 480p resolution only. This has to do with the fact that while AV1 is more efficient than VP9, it also requires higher computational power.

The free AV1 Codec 1.0 - compiled by Chocobo1 includes x64 executables for Windows: aomdec.exe and aomenc.exe (AV1 decoder and respectively encoder) and can be downloaded from *https://www.free-codecs.com/download/av1_codec.htm*

Source: Alliance for Open Media

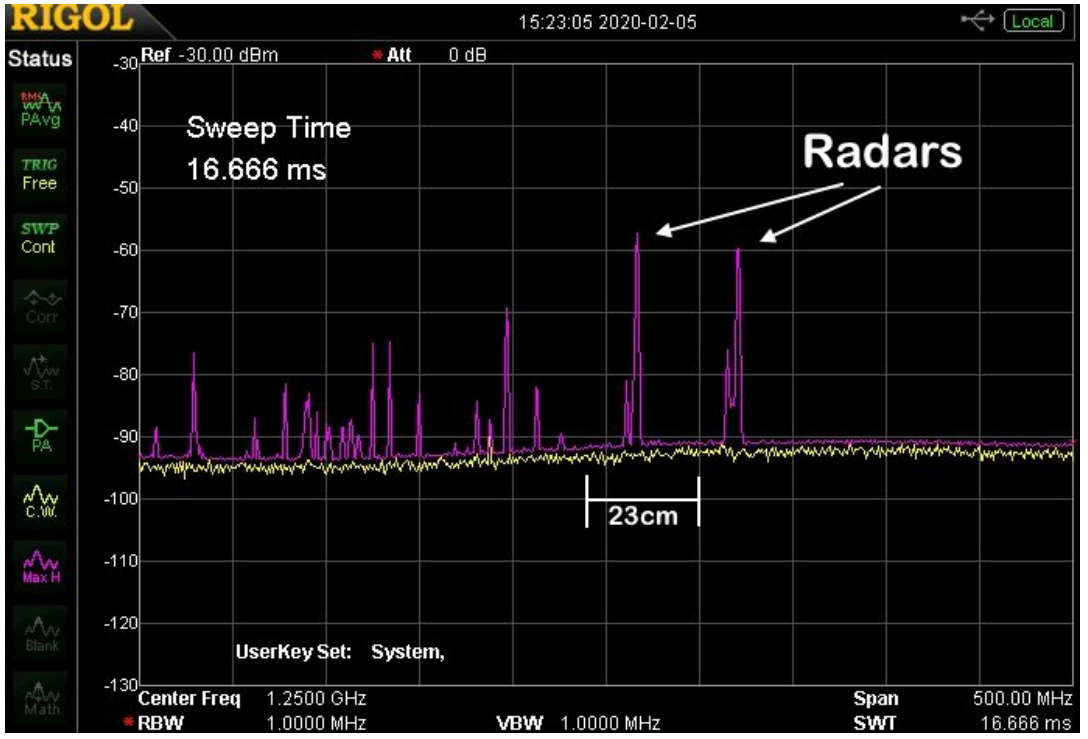
ATSC 3.0 Update:

The new standard for over-the-air digital television in the U.S. continues to advance. Sixty television stations in forty markets are expected to be on the air with the new standard by the end of 2020. Twenty television sets with ATSC 3.0 tuners were shown at the recent 2020 Consumer Electronics Show. Several stand-alone tuners (set top boxes) were also shown. Stand by for more updates and announcements after the National Association of Broadcasters convention in Las Vegas coming up in April.

If you haven't been thinking about the ATSC 3.0 digital television format, you may want to give it some consideration. This is going to be the new digital broadcast standard in the U.S. It utilizes COFDM and it is FAR superior to ATSC 1.0 in terms of signal coverage and capabilities, including I.P. protocol, 4K video and enhanced emergency broadcast features. Further, if you operate an ATV repeater you would be wise to retain your 6 MHz channel allotments. Although European DVB can be operated at less than 6 MHz channel widths, I am not aware of any provision in the ATSC 3.0 standard to do so.

Source: Dan Rapak - WA3ATV

Boulder ATV Repeater Site Spectrum:



On the 5th of Feb. in the mid-afternoon, Don, N0YE, Colin, WA2YUN, & Jim, KH6HTV set up a couple of spectrum analyzers in the parking lot of our ATV repeater's site on the mesa south-west of the city of Boulder. We were interested to see what, if anything we found in the 23cm band. In general, the amateur band is very quiet, except for the FAA radar. The above plot shows what we found sweeping from 1 to 1.5 GHz.. The yellow trace is a "live" single sweep. The magenta trace was with the spectrum analyzer in "peak hold" mode for several minutes. The analyzer's bandwidth was set to 1 MHz.

Source: Boulder Amateur Television Club TV Repeater's REPEATER

This is your free ATV magazine.
Please consider contributing an article!

A NEW Wide-Band LNA:

I have just developed a new, improved LNA covering from 6m to 13cm (50MHz - 2.4GHz) bands.

I want to share the good news with you. It has particularly outstanding low noise performance on 2m thru 23cm with << 0.8dB Noise Figure. It is based upon an MMIC I recently discovered. The RFMD SPF-5189Z. It is a GaAs, pHEMT, LNA, MMIC.

Unlike some of the other amplifiers I have designed in the past, this one does not have a flat, S21 gain frequency response. It does however have high linearity of OIP3 > +38dBm. It also has a high P(-1dB) gain compression of >+20dBm over it's useful frequency range.

I have designated this as my model WB-LNA-3. It far outperforms my previous LNAs the -1 & -2 and I have thus discontinued offering them.

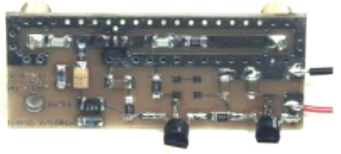
See the spec sheet for further details on this new LNA.

It includes photos of the pc board and detailed performance data. I am electing to sell this new LNA, either as a bare pc board only (no components) for $11, as an assembled pc board, including SMA connectors for $36, or as a complete amplifier in an all metal enclosure, tested with calibration report for $65.

Website: *http://www.kh6htv.com* **email:** *kh6htv@arrl.net*

Jim Andrews kh6htv Boulder, Colorado, USA

Model WB-LNA-3
Wide-Band, Low Noise AMPLIFIER



The KH6HTV VIDEO Model WB-LNA-3 is a Wide-Band , Low Noise Amplifier with useful frequency response covering from 50 MHz to 2.4 GHz. It has very low NF < 0.8dB from 2m through 23cm. It is also capable of handling high input signals with Pout(-1dB) > +20dBm. The amplifier is available for sale, as a simple pc board, as an assembled pc board, or complete in a metal enclosure with test report.

Typical Performance

Parameter	50MHz	144MHz	430MHz	915MHz	1.27GHz	2.4GHz
Noise Figure	1.7dB	0.6dB	0.5dB	0.8dB	0.8dB	1.6dB
S21 Gain	23dB	25dB	22dB	16dB	15dB	10dB
S11	-7dB	-8dB	-11dB	-14dB	-13dB	-10dB
S22	-17dB	-25dB	-23dB	-18dB	-24dB	-18dB
Pout(-1dB)	19 dBm	22 dBm	22 dBm	20 dBm	22 dBm	21 dBm

DC Supply Voltage	12 Vdc @ 100mA, +11 to +15 V range, internal voltage regulator
RF & DC connectors	SMA jacks (female) DC = solder terminal on feed-thru capacitor
Dimensions	2.2"x0.9" (pc board) - or - 1.5"x3.6"x1.25" die-cast enclosure

KH6HTV-VIDEO www.kh6htv.com e-mail: kh6htv@arrl.net Boulder, Colorado, USA

Micro Corner
Basic for Ardunio – ESP8266 Quick Test

Written by Mike Stevens G7GTN

Another option for programming our Ardunio or ESP8266 modules we can make use of a modern version of the Basic programming language. It is a free application that runs under the Windows Operating system to provide a comprehensive coding IDE.

You can make use of a large selection of additional libraries as you would do within the Ardunio environment. These libraries are wrapped from the original C code that we might be already familiar with. This allows us to later on create our own libraries by wrapping them, or by using inline C code to call very specific core functions.

Download and installation

Follow the instructions on the linked page to get your installation *https://www.b4x.com/b4r.html* since I currently use Ardunio IDE Version 1.8.11 I needed to update rCore Library as indicated.

The ZIP file needs to be added to the Anywhere Software Library installation location, the XML descriptor file should be placed outside of this location as depicted in Figure 1

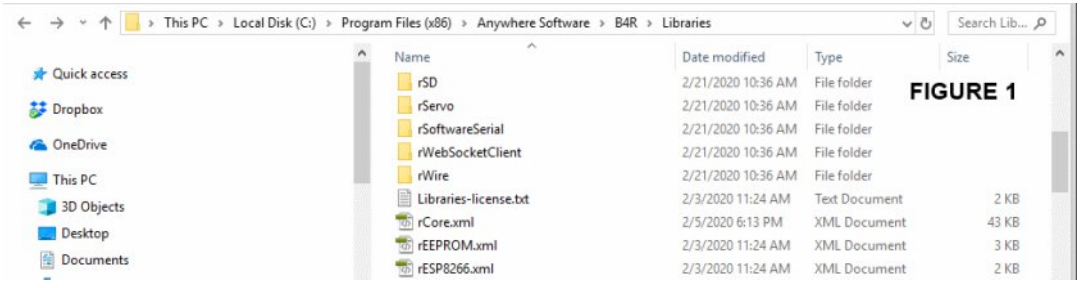
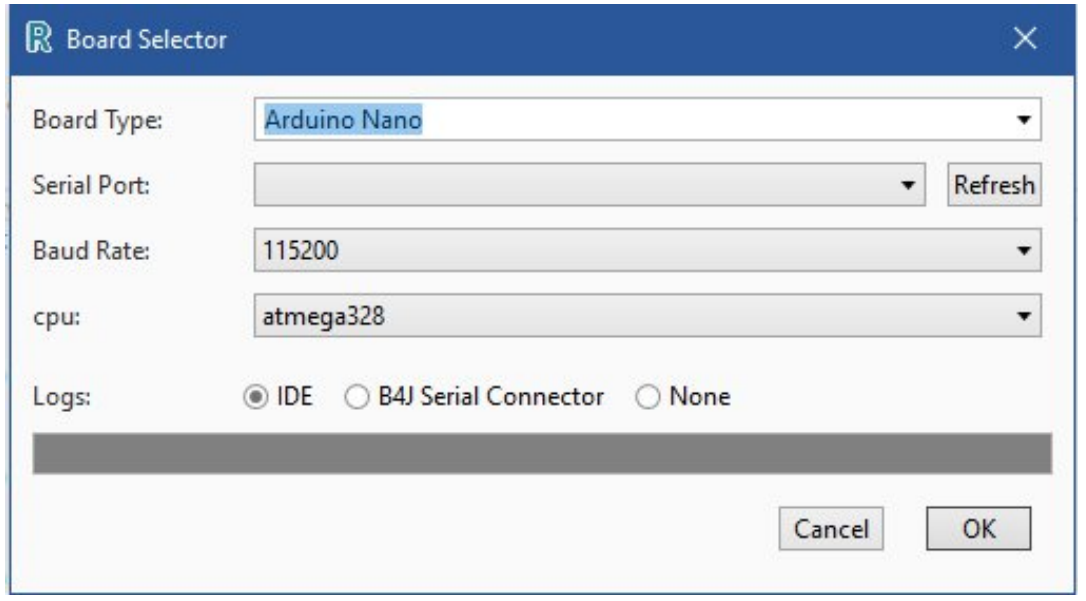


FIGURE 1



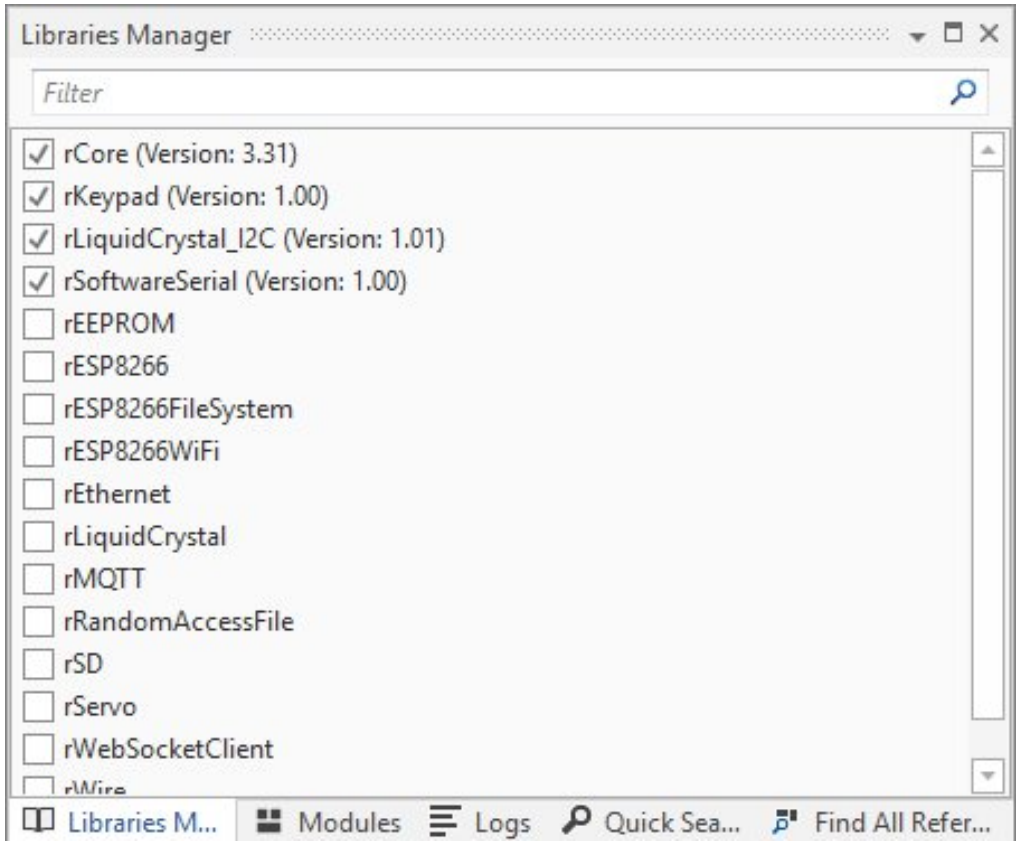
With the installation complete you need to select the type of board you wish to use from the Tools Board Selector option. A drop down selection box will be populated – this comes from the Ardunio IDE installed boards file so if you have not previously installed them there that option will not be available to you.

IDE

As we would expect the Development environment provides such features as code folding and a syntax example if you place the cursor in front of a command. There is a separate window that can display Log messages as your code is running which helps while debugging sections of your project.

A Practical Test Project

Now we have the installation details sorted out a simple test project will help to get more familiar with the IDE options. As opposed to Ardunio where we have to include required files within the code here we download the libraries and these are then shown in a manager pane for us to select from.



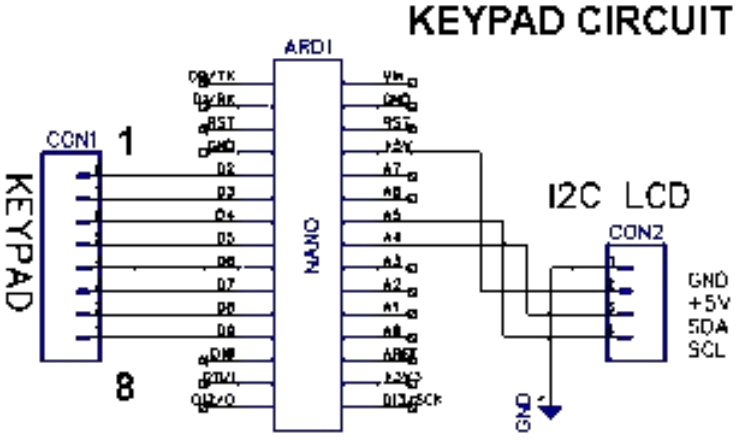
The test code I wrote reads a (4X4) Telephone Matrix keypad and displays this as a Log message in the IDE with I2C LCD optionally available. Using an Ardunio Nano the keypad fits directly on to the header pins starting at D9, the keypad has a small number 8 printed on the plastic shell to indicate this as the first pin.

If you wish to connect an I2C LCD the SDA connection goes to A4 with SCL on A5 you might need to make a change to the address if your PCF8574 controlled display is not set at (0X3F)

```
48      lcd.Initialize(0x3F, 16, 2)
49      lcd.Backlight = True
50      lcd.Write("Press Button")
```

If you need help to determine the address of your display, use the I2C Scanning function by removing the comment. This will discover the address of your display for you to make the change.

The simple KEYPAD project source code can be downloaded from the software page as keypad.zip the two required libraries are in the links and also noted within the sourcecode for you to download and install..



Conclusion

If you prefer to code in the Basic programming language on Ardunio or EPS8266 modules this is a powerful environment to do so. Instead of having a limited subset of commands available you have the ability to add in additional libraries developed by the Ardunio community.

The documentation is of a high standard for a freely available product. One downside might be that it is only available for the Windows Operating system, if you can work with that then this provides a powerful development system.

Where the Basic that Trevor G8CJS has been coding all his GVG switching experiments in is just web browser based and covers the ESP8266 and ESP32 devices we can use this different solution for lower end Ardunio boards as well.

You can of course also code in your favourite text editor for ESP Basic and again using a Windows only utility upload and download your code to your ESP processor. I feel happier and more accustomed to using a tethered USB IDE type system.

At least we have two very solid solutions we can use freely to develop our micro controller projects. I can see that both are fully capable of developing some quite sophisticated systems albeit slightly differently.

What I personally see as an advantage is that I can wrap a library from Ardunio that I need without having the ESP Basic author having to do this or even want the functionality added in the first place.

Both systems are working off the community backend with using libraries employing some of my own endeavours I can get to determine extra functionality without community Forum debate on use cases I might actually have.

I can easily live without having a Web Based development environment just for this single gain. I do commend both different versions of Basic for further project investigations.

Code download *https://cq-datv.mobi/downloads.php#latest*

References

https://www.b4x.com/b4r.html

https://www.b4x.com/android/forum/threads/updates-to-internal-libraries.76644/#post-709411

https://www.b4x.com/b4r/documentation.html

https://www.b4x.com/b4r/help/rkeypad.html

https://www.b4x.com/android/forum/threads/liquidcrystal_i2c.66460/

https://www.b4x.com/android/forum/forums/b4r-libraries.78/

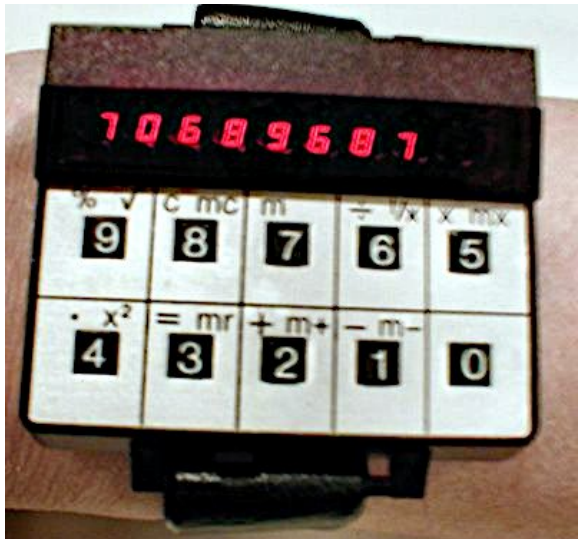
https://www.arduino.cc/en/Main/Software

50 Years ago

Written by Trevor Brown, G8CJS

THE 1970s

There were one or two innovations in the 70's that appeared and disappeared without trace. The first one that comes to mind was a wrist calculator, sounds good, but did not look quite as cool as an apple watch. It was launched in February 1977 by Sinclair Instrument's. The word Sinclair became synonymous with cutting edge projects of the time. The Sinclair versions were just a little different from what we use today. Inventions that we have become used to don't look quite like the Sinclair version of the 70's. Electric bikes are not anything like his Zike, or electric cars don't have much in common with the C5.



The wrist calculator cost around £11 in the UK and was only available via mail order in kit form. It was a difficult build due to part size variation. Having completed a few for colleagues who ran into difficulties with components, which clearly would not fit the PCB.

The design used 10 keys, with a three-position switchto select the correct function. The display was LED and yes, it had a micro, the Mostek MK 50321N. I have no idea of how long the battery lasted.
This was not Sinclair's first calculator; it was preceded by the Sinclair Cambridge in 1973. This was available in both kit formand ready built. Several models were available including the more popular Cambridge Scientific.

Clive Sinclair's problem was he was a little too far ahead of his time and the technology was not available to support his ideas. None of which had much impact on Television. We had to wait for his Sinclair Spectrum, which was in the 80's.

TheCommodore PET was with us in the 70's and as Ian pointed out in issue 79 was based on the 6502 processor. It was delayed until 1977 with production problems and by that time we also had the TRS 80 and the Apple II. These three were often referred to as the 1977 trinity.

The Pet keyboard was not popular, but those that complained had yet to see the Spectrum keyboard

Commodore was the first company to license Microsoft's 6502 BASIC, followed by Apple.

In 1979, Commodore replaced the original PET 2001 with 2001-N (the N was short for "New"). It now had a conventional full-sized keyboard and Commodore's newly introduced disk drive. It was offered in 8KB, 16KB, or 32KB models as the 2001-N8, 2001-N16, and 2001-N32 (the 8KB models were dropped soon after introduction). The cassette deck being replaced with the floppy disc. CP/M was here and 6502 machines could not run CP/M (sorry Ian).

CP/M,originally (Control Program for Microcomputers) was all about portability of software. It was created in 1974 forIntel 8080/85-basedmicrocomputers. The CPU'S that could run CP/M were the Intel 8080,Intel 8085,Zilog Z80 (one restriction on portability was that certain programs written for CP/M had to exclude the use of the extended instruction set of the Z80), Zilog Z8000, Intel 8086, Motorola 68000.

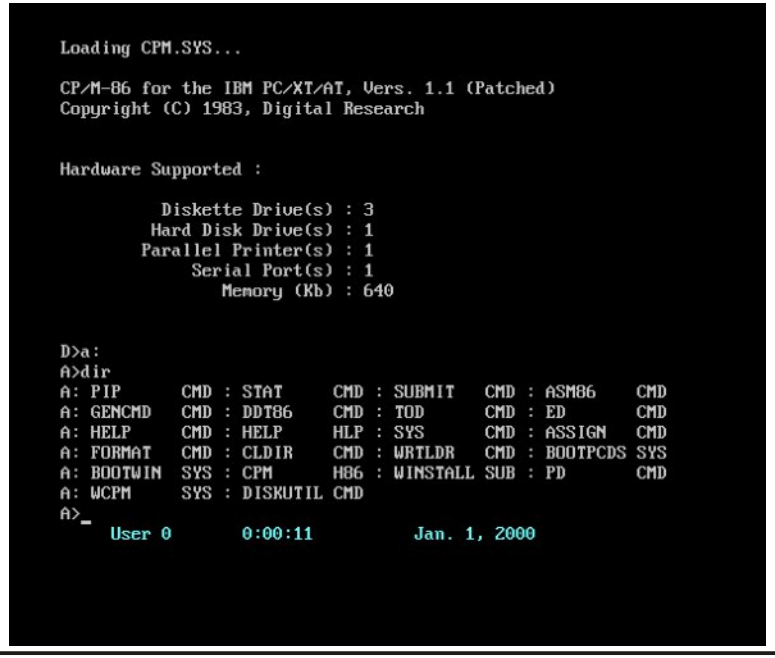
It was initially confined to single tasking using 8-bit processors with no more than 64K of memory. This computer platformwas widely used in business through the late 1970s and into the mid-1980s. CP/M increased the market size for both hardware and software by greatly reducing the amount of programming required to install an application on a new manufacturers computer. CP/M was displaced by DOS soon after the 1981 introduction of the IBM PC.

CP/M was revolutionary. The problem was all the home computers used fixed hardware to cope with data save and restore (usually cassette-based tape), keyboard and screen. These were all different so even if the code was the same as another brand running the same CPU there was no way of running the programme.

CP/M found a way around this in that the floppy disc was here and the outer track was reserved for the system of your computer. Take a Disc from another computer change the data in the system track to that of your own and the code cold be run as it vectored all the keyboard, screen and disc

commands via this track so thanks to Gary Kildall we had the first transportable operating system for 8080 and Z80 CPU computers.

Then it gets a little foggy and there is a lot of rumours around what happened next. IBM brought out the PC and needed an operating system. They approached Gary for an operating system but wanted him to sign a non-disclosure agreement before they had discussions. This did not happen, (was he around or did he refuse there are various versions) so they approached Bill Gates who was happy to sign, but allegedly did not have an operating system, but allegedly said he did or would write one. Again, rumours that he allegedly adapted CP/M and called it PC Dos. MS Dos was born, but was its CP/M? There are many arguments that still rage today. Gary Kildall is no longer with us and if his work allegedly lives on MS dos or windows we will never know. However it makes a good story; it has often been said that Gary embedded a copyright notice in CP/M but if this is truth or fiction we may never know. What we do know is it has never popped up on a PC, but it makes a brilliant story.



As home computers blossomed so did BASIC as a resident programming language it was not new. Invented byJohn G. Kemeny and Thomas E. Kurtzof Dartmouth College in Hanover, New Hampshire, back in 1964. Just the hardware had not been available to run it in the home. What was good about BASIC was the text file was the actual source code, it did not need compiling to create computer code before it could be used. It did this at run time and was called interpreted.

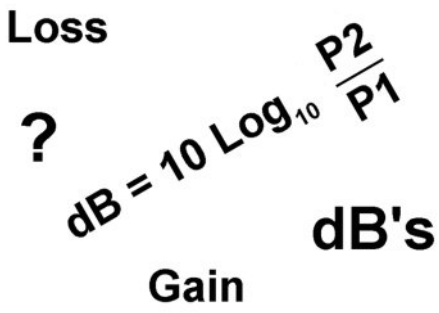
This made it easy to understand and edit as everyone had the source code. This made it hard for anyone trying to earn a living as the secrets were in plain view. The norm was to write a source code compile it, sell the executable file, and keep the source code. BASIC got a reputation for being slow because it interpreted the code. The other problem was dialects. Every computer had its own version and so if transported to a different computer it would need adapting to run. Enter Microsoft who tried to write a Universally accepted BASIC so every computer was on the same dialect.

Some used Microsoft BASIC, some avoided it EG Spectrum BASIC, BBC BASIC and so on. BASIC still refuses to die and now we have ANNEX BASIC which I have used with the ESP Micro to investigate the workings of the GVG panel, yes I have adapted what I learnt in the 70's when I was a little more studious than I am today, but a day spent learning is never a day wasted.

The Zike bicycle, like the Sinclair C5 Car, was battery powered

Decibel Math – Throw-Out the Calculator

Written by Roger Paskavan, WA0IUJ
Reprinted from Boulder Amateur Television Club TV Repeater's REPEATER February, 2020



We all want to know how many dB of gain that antenna modification made. Yet, most amateur operators dread the thought of doing logarithmic calculations to try and get to the bottom of a gain or loss problem. There is another way of looking at decibels and only requires knowledge of simple arithmetic. In fact it is so easy that you don't even need a calculator. It can be done all in your head. So let's look into this not so new form of Ham magic. There is a non-calculator way to solve decibel-log problems in seconds without using charts, tables or calculators. There is no need to involve algebra, just some addition and multiplication of simple numbers. First, here is some background on the "dB".

The word Decibel dates to the late 1800's of Alexander Graham Bell and his research on the famous telephone. Alexander had a lot to do with measurements of audio and signal intensities in his work so he created a term called the "Bel." (One L intentionally missing). The problem with the Bel was its numerical size. It was a huge unit probably derived from the large signals needed in those days to operate the crude telephone equipment. With the modernization and efficiency of components, the same telephone services are now achieved with much lower signal levels. In honor of its inventor, the logical thing to do was to keep the unit of measure and apply the metric system to the Bel.

So, the Bel was divided by 10 and prefixed with the word deci (meaning 1/10 in metric). This became the root of the word, deci-bel, later termed dB. We now had a common way of expressing gain or loss in the form of a ratio between power, voltage or current. The B is only capitalized since it refers to Alexander's last name.

A decibel expresses the ratio between two values with the same dimensions. The quantities compared may be two power levels, two voltages, two sound pressure levels, and so on. Since the quantities in the ratio always have the common same dimension units, the dimensions cancel out; the decibel itself is dimensionless. But, dB measurements must be made between identical impedances to be accurate. For example, two power measurements must be on two 50 Ohm lines or 600 ohm line, etc. for the numbers to work.

Decibels can be used to express gain or loss between any number of antennas, amplifiers or two measurements. They can be utilized to show the input/output differences between two amplifiers or two antennas thus providing meaningful gain figures, etc. This article will deal with power measurement.

So now that I've defined some basics, let's look at a method of finding dB's without doing those painful logarithmic calculations. To begin understanding this method, you need to memorize three key numbers and their associated dB figures with a multiplier. 10 dB = 10 x, 6 dB = 4 x and 3 dB = 2x. (x= times)

The Key number is telling you what to do with the power value when its related dB figure is used.
For 10 dB, the key number is 10. This means that for an increase or a +10 dB change, our power level would be multiplied by 10. For a decrease or a -10 dB change, we would divide by 10.

For 6 dB, the key number is 4. This means that for an increase or a +6 dB change, our power level would be multiplied by 4. For a decrease or a -6 dB change, we would divide by 4.

For 3 dB, the key number is 2. This means that for an increase or a +3 dB change, our power level would be multiplied by 2. For a decrease or a -3 dB change, we would divide by 2.

While it is quite easy to visualize 3 dBs as doubling the power, larger numbers follow the same logic but it may not seem that way. It is more difficult to realize that an LNB with 57 dB of gain will have doubled the power of a 54 dB gain LNB. Anyway, you cut it, 3 dB is double your power. Remember, the power is doubled for every 3 dB and halved for every 3-dB loss. In my original example, we had to double the power 18 times to get to 54 dB's and then double once more to get to 57 dB. Remember, if you increase a 500 Watt linear amplifier to 1000 Watts that is still only a 3 dB gain even though the numbers are big.

By way of another example, a dish antenna with a 30 dB gain delivers one-tenth (1/10) the signal power of a larger dish having a 40 dB gain. The signal power at the dish focal point is multiplied by 10, four times to obtain +40 dB of gain. In 30 dB example, the dish antenna signal power was multiplied by 10, three times to get to a gain of 30 dB. (The difference is 10 dB or 1/10 the gain for the smaller dish.

A resistive power splitter loses 6 dB per port. What does this mean? With 6 dB gain or loss, the multiplier/ divider is four. This means that for a given signal power at the input, each output port will have a 6-dB loss because of splitting the signal. Since 6 dB is four times and this is a loss situation, the input power is divided by four at each port. Referring to signal power, if the RF drive signal input is 10 milliwatts, each port would measure 2.5 mW.

So, let's go through some worked examples of combining the key numbers, 3, 6 a 10 to figure dB's in your head and throw away the calculator.

Example 1: A UHF amplifier boasts a gain of 13 dB. If you put 4 watts of drive into this amplifier, what will be the power output? First, increase the level by 10 dB of the 13 dB (to 40 Watts). That's 4 watts times 10, then increase the 40 watts by the remaining 3 dB, which is 2x (double the 40 W). The answer becomes 80 watts output and no calculator.

Example 2: Your new UHF beam antenna has a gain of 7 dB. If we feed it with 4 Watts, what will be the radiated power? First, increase the level by 10 dB which is 40 W, then subtract 3 dB by dividing 40 by 2. You are now at 7 dBs gain and the answer is 20 Watts ERP.

Example 3: If you feed 10 Watts drive into an amplifier and you measure 25 watts output, how many dB gain is that? In your mind, run through combinations of dB equivalents. Utilizing the 10 W input power, figure the permutations that make 25 watts. The combination that worked was add 10 dB which is (10x) then subtract 6 dB (which is divide by 4) and the answer is 4 dB. (10 dB – 6 dB = 4 dB of gain.)

Example 4: You replace your old antenna with a new 9 dB VHF beam. If you feed in 1 KW, what will be your ERP output? Figure combinations that make 9 dB. Use 6 dB and 3 dB to make this work. (6dB + 3 dB =9dB) Six dB is a 4x multiplier and then add 3 dB which is a 2x multiplier. Your astonishing answer is 8000 Watts ERP. (1000 x 4 = 4000 then times 2 = 8000W) Scary!

Example 5: This one is wild, an LNB for an 13 cm satellite dish is stamped 43 dB gain. How many times is the input signal from space multiplied before it arrives at your receiver? First, use 10dB and 3 dB to figure this problem. Make the initial calculation 40 dB, then add 3 dB to get your

final answer. 40 dB is 10db + 10dB +10dB +10dB. Since its key number is 10, we multiply 10 x 10 x 10 x10 = 10,000 times, then you add 3 dB by multiplying the 10,000 x 2 = 20,000 times, which is your answer.

That is a whole lot of gain in any device. (43 dB is a gain of 20,000 times)

As you can see 3, 6 and 10 dB are fixed given ratios of power. By combining these three numbers in any number of combinations, most dB problems can be resolved without logs or a calculator. Keep in mind that for positive dB gain you multiply the ratios. For negative gain (loss) you divide the ratios.

If a fixed numerical reference is used in place of one of the ratios, then the Decibel becomes a measure of signal against some referenced standard which is called 0 dB. (Similar to par in a golf game)

For electronic calculations of power, the commonly used 0 dB references are: 0 dBm = 1 milliwatt (0.001 Watts) and 0 dBW = 1 Watt

This means that all gains or losses are above or below the given reference value. In this context, 20 dBm is 100mW (1mW x 10 x10) and 50 dBm would be 100 watts. (1mW x 10 x 10 x 10 x 10 x10)

To make sense of this measurement, the reference must always be given. In this system, gain or loss utilizes the reference (0 dB) as one of the ratios. So your answers are numbers above or below that given reference.

dB measurements can also be used for voltages, sound pressure and current measurement but that is a topic for a future article. Hope the 2020 New Year brings you a new way of thinking about the age-old mystery, the decibel.

Roger Paskvan WA0IUJ was licensed at age 15 in 1961 and has been a Ham for 58 years under the same call sign. He holds a B.S. degree in Industrial electronics, A Masters degree in Education and is employed as a 42 year professor at Bemidji State University, Bemidji, MN. He has taught engineering classes in broadcast radio and television for the past 42 years. Roger is a broadcast maintenance engineer and owns a number of AM and FM radio stations. Working in broadcast television, he also enjoys 20m SSTV and dabbles in fast scan 13 cm Ham TV. Roger enjoys reading about DVB-S2 transmissions and would like to build a DVB repeater. Waiting for a USA version of QO-100 that would really build DVB interest. He is a VE and offers ham classes, being responsible for hundreds of new licensed hams over the years. Roger can be contacted at: *rogerp@paulbunyan.net*

Grass Valley Mixer Conversions - Part 14

Written by Trevor Brown, G8CJS and Mike Stevens, G7GTN

The GVG mixer although ahead of its time dates to the mid-80's and we have to realise the limitations of kit in this era.

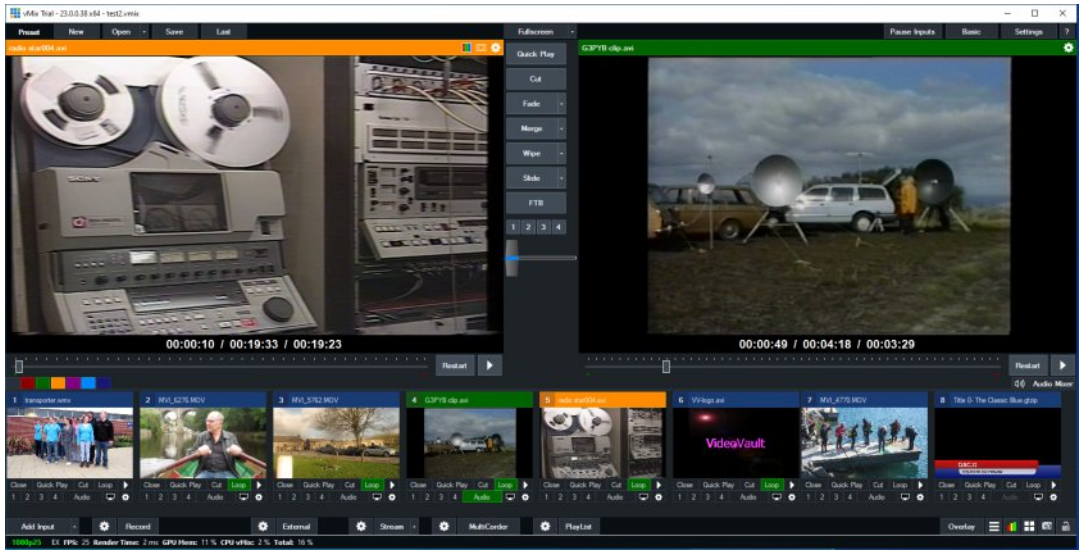
It could only mix synchronous sources, which is sources all locked to single SPG. Colour black had to be fed out to the camera's and they would synchronise their outputs to this signal.

Adjustment would then have to be made for line timing and colour phase, due to cable lengths. Subcarrier moves 1° for every 8" of cable. So, all the vision mixer sources would be viewed on a waveform monitor and vector scope and adjustments made so the mixer would have the same phase sources on all its inputs.

Fortunately, we are now in 2020 and things have improved no end, hence the GVG panel has been changed to work with improved technology (or is about to be). All these limitations are long gone in a modern mixer such as the software package called VMIX.

It's not free but there are two no cost options when you download the package.

Demo which has limitations but is usable and full and has everything, but only runs for 60 days. I chose the full as I needed to explore all the functions and not walk into any barriers.



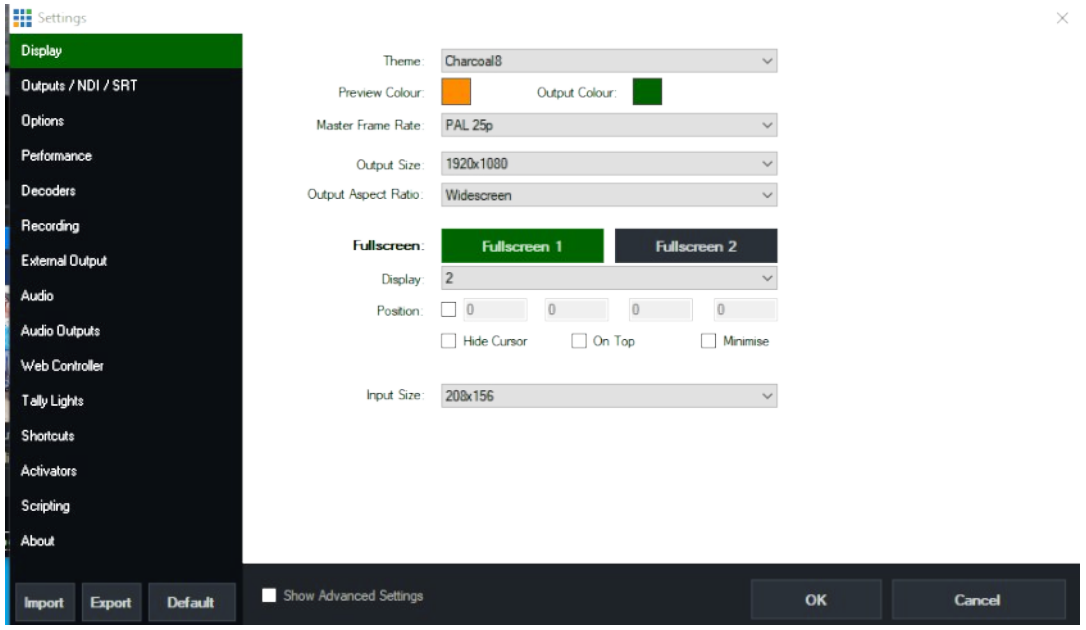
Note the demo version of VMIX will have a limitation on the number of inputs

I don't have any camera's available as they need video cards to allow interfacing to the software, but I do have a hard drive full of video clips and by using the bottom left icon (Add Input) they can be added as sources. If you have a video card or a webcam it can be brought up in just the same way. No need to worry about synchronous sources or phasing up sources, thats long gone with turret cameras and other long in the tooth hardware.

Just point and shoot at any source with your mouse and it will move to the preview monitor. The cut button will switch the preview and on-air monitors over and animate the video clip at the same time. You can also set up digital effect's transitions using the bank of buttons between the monitors.

This software is very intuitive and you will be in control of it in no time.

Once you are on top of controlling it go for the settings button (top right) this will produce the menu.



Far left, (select short-cuts) and you will be presented with

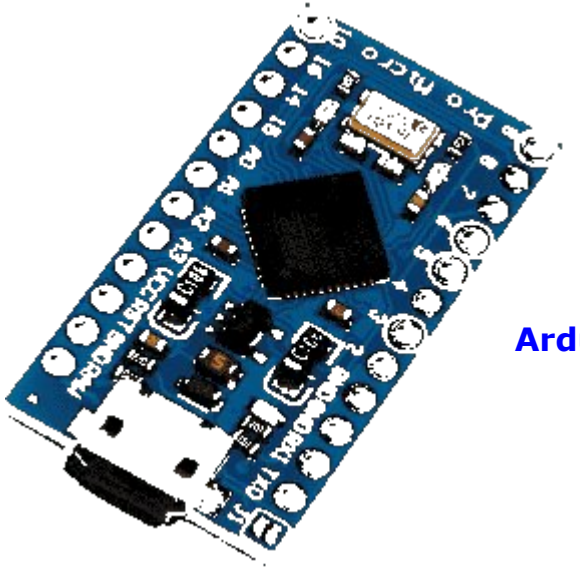
[1] PreviewInput 2 MVI_6276.MOV
[2] PreviewInput 2 MVI_6276.MOV
[3] PreviewInput 3 MVI_5762.MOV
[4] PreviewInput 4 G3PYB clip.avi
[5] PreviewInput 5 radio star004.avi
[6] PreviewInput 6 VV-logo.avi
[7] PreviewInput 7 MVI_4770.MOV
[8] Cut Preview (0s)
[9] OverlayInput4In 8 Title 0- The Classic Blue.gtzip
[10] OverlayInput4Off

Here you can set up keyboard functions to act as hot keys. I have set some as examples. By selecting Template you can see your choices on a keyboard graphic.

Choose your hot keys and practice using them instead of the mouse

The next stage is to remote these key presses out to our GVG panel. We could opt for mapping in a whole bunch of PCF 8574's and butchering an old keyboard for the interface or we could go with a more elegant solution and use the standard MIDI protocol of VMIX. This is a 31250 baud rate serial based system originally developed for the control of musical instruments.

Since the GVG Panel is being controlled via I2C it makes sense to have a custom MIDI controller also communicate using this protocol. We opted for a custom-built interface, the Ardunio Pro Micro-controller.

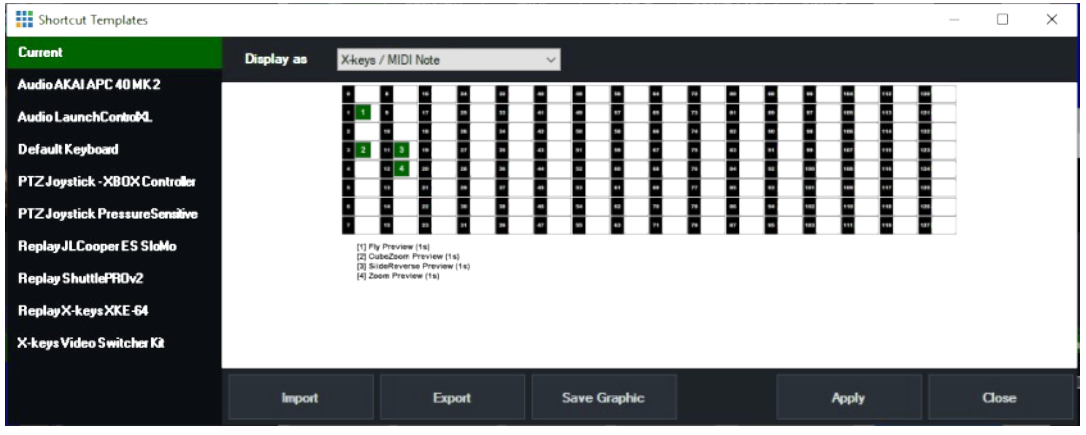


Ardunio Pro Micro-controller

Using the small Ardunio Pro Micro we can create a USB to MIDI control element since we can programme the ATMEL 32U4 processor to be a Human Interface Device (HID). This has a custom I2C address and becomes a Slave unit that ESP Basic will send control numbers to. We sit in a loop waiting to receive any of these numbers.

When one is received a custom Midi Note is sent which we later find and map from within VMIX – in a similar fashion to the keyboard control already mentioned. We have defined a provisional KEYMAP for each of the buttons on the physical panel to enable us to know which button has been pressed. We can have three different types of messages transmitted; the first two are what is known as a Note On and Note Off command.

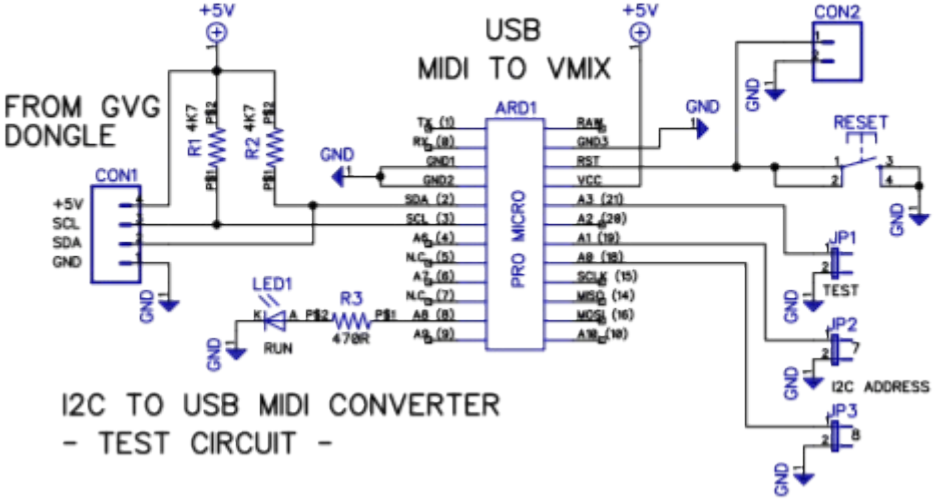
The last is called a Control Change that unlike the first two can vary in value. These particular messages would be used for Potentiometers, such as the T-Bar fader. A note worth adding is that MIDI messages have what is termed a Channel Number. These can be up to 16, after this and the actual note we also have a velocity number. This relates to the force applied to a keyboard key press. Since we do not anticipate the GVG broadcast panel having a function within music we have set this to a value of 64 which is broadly termed as being "Normal Velocity". Using the ability of saving and then importing templates it is quick and simple to create separate operator profiles that map specific keys to used functions.



The communication to the GVG is only one way so the GVG software needs to present the same information on its keys as interpreted by VMIX software so yes, we can have Preview controlled by the GVG pre-set Bank, On Air direct by the PGM bank and so on, but we need to address the first problem and that is on VMIX any source that is put on the TX monitor either direct of via the Preview monitor causes the Preview monitor to show the source it replaced.

This is not the case with GVG 14 software. It will need to be updated to GVG 15 (in the pipeline) which will keep the two banks of the GVG mixer in step with the Preview and On-Air monitors of the VMIX display. We can also assign some of the yellow buttons to effects, again GVG 15 will do this.

This I hope completes the hardware. Mike is almost there with the code for the Ardunio Pro Micro. I have made a start on GVG 15 which I will post when I have it working, so watch this space!



A Not So Simple Combiner

Written by Jim Andrews, KH6HTV

When we received permission to install our ATV repeater in it's new home on the mesa, it was with the restriction that we add no more coaxial cables from the radio room. There was only one spare, LMR-400 coax available for us to use. Also there was no more space available on the tower to add our DB-411, 4 element, co-linear, 70cm transmit antenna. If we wanted to move in, we would have to share BARC's 70cm FM voice repeater, DB-411 antenna. How to do that without screwing up each other's transmitter? The solution we came up with used what looks like a very simple, type N tee adapter to join each transmitter to the antenna. But wouldn't that send an awful lot of RF power not just into the antenna, but also the other transmitter? Yes, but No.

Let's put some transmission line theory to work.

What happens to an impedance when it is separated from the point of connection by a 1/4 λ piece of coax cable ? The Z takes a trip 180 degrees around the Smith Chart. So if I have a short circuit on one end of this 1/4 λ coax, on the other end – I see an open circuit, or in other words an infinitely high impedance (∞ Ω). Now this same phenomena repeats for all odd values of 1/4 λ, i.e. 3/4 λ, 5/4 λ, etc.

So, fortunately, our two transmitters are at opposite ends of the 70cm band. The BARC FM transmitter is at 448.9 MHz. Our BATVC transmitter is at 423 MHz. Now if we carefully (emphasis on VERY CAREFULLY ! !) prepare two coaxial cables of exactly the correct lengths we can have two very nice n * 1/4 λ, Z transformers.

The antenna connectors on both transmitters are actually dc short circuits. The inputs to the BARC duplexer and our BPF are DC short coupling loops.

By cutting each coax going to the type N tee correctly, we can transform each of these shorts to ∞ Ω. The coax connecting the BARC transmitter to the tee needs to do this for our frequency of 423 MHz. The coax connecting our transmitter to the tee needs to do this for the BARC frequency of 448.9 MHz.

A complicating factor was the fact where the tee attached to the 7/8" Heliax was near the ceiling in the radio, thus each cable to the transmitters had to be quite long at 7 to 8 feet in length. Don, N0YE, was able to accomplish this heroic feat with the help of Colin, WA2YUN.

The resultant combiner worked and we measured -1.8dB of insertion loss for our 423 MHz DTV signal.

Creating your own Test Card
First published in issue 5
Written by Richard Carden, VK4XRL

Yes it's a Philips PM5544 test card but with CQ-DATV added. It is one of the preset test cards available at *http://www.oodletuz.fsnet.co.uk/soft/tcmaker.htm* ***(link no longer functional. ED)*** but this site is so much more than preset test cards, there are all the elements available to quickly put together a test card of your own. The Test card maker software will enable you to design and create your own test card, by putting together premade patterns such as colour bars, grey scales, castellations and text. The preset items can be scaled and arranged in layers and then viewed, and re edited until you have your very own test card which can then be saved as a jpeg.

Here is a run-down of some of the help file for your information;

The individual parts from which you can create a test-card are called Objects and the completed test card is a tcd file which can be turned into a JPG or BMP file.

- *Greyscales*
- *Solid blocks*
- *Alternating colour blocks (e.g. castellations, low-frequency gratings)*
- *High frequency sine grating*
- *Colour gradients - Circles / ellipses - Boxes / frames*
- *Single straight lines*
- *Crossbars (corner stripes as seen on Test Cards C, D, E, F)*
- *Bulls eyes (sets of concentric rings as seen on many optical test cards)*
- *Sweeps (gratings of smoothly increasing frequency)*
- *Frequency Bursts (an more versatile type of grating)*
- *Star Bursts (radial frequency grating)*
- *Triangles*
- *Crosses (horizontal and vertical lines with optional solid background)*
- *Grid crosses (extra styling on grid lines as seen on Test Card F)*
- *Light spots (three optionally overlapping spots with colour bar colours)*
- *Text - Clocks (time, date or countdown)*
- *Pictures*

For each of these, a position and size are specified, plus several other properties e.g. colour bar intensity, grating frequency, line thickness.

Position and size are relative to the logical coordinate grid. There can be any number of each type of object in each layer of a TCD.

Each TCD has several properties which can be amended, and which apply to every layer defined within the TCD:

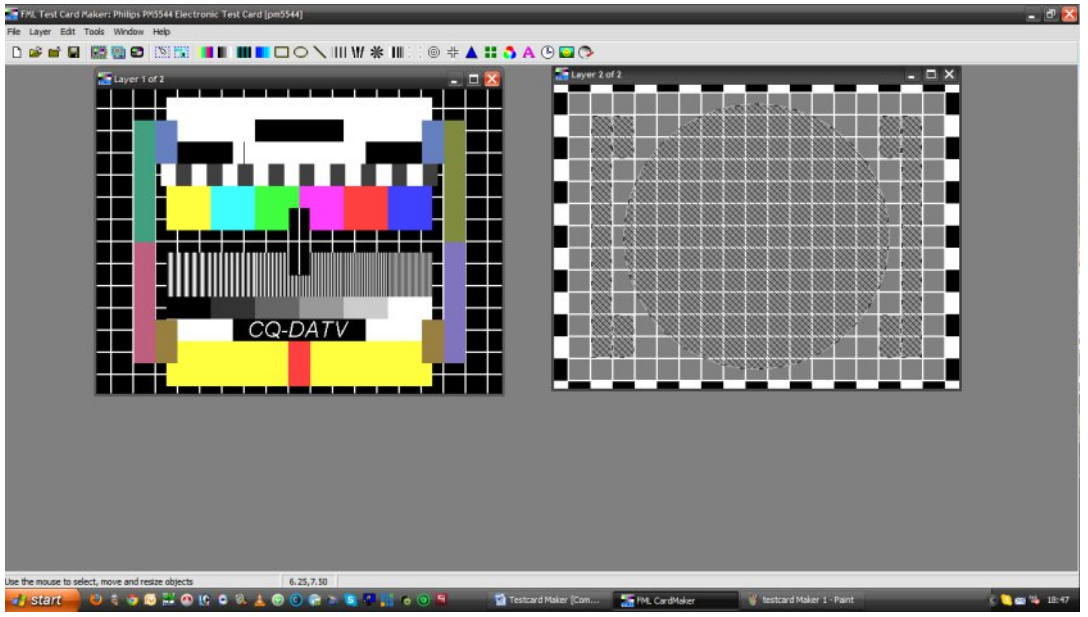
- *Title*
- *Aspect ratio*
- *Layer editor size*
- *Coordinate grid size*
- *Border size*
- *Forcing of square grid cells*
- *Alignment of grid to pixels (i.e. forcing each cell to be exactly the same size)*
- *Effective TV line frequency (for defining frequency bursts in MHz)*

The following properties can also be amended separately for each individual layer:

- *Transparent or opaque background*
- *Background colour (if not transparent)*
- *Grid thickness and colour*
- *Position of physical grid relative to the card's logical grid*

Basic Use

When you start up Test Card Maker or select File-New from the menu, you are presented with a default TCD consisting of single, blank layer with a mid-grey background. The aspect ratio is 4:3, editor size 512x384, with a 16x12 logical coordinate grid, and no physical grid. This automatic default can be switched off - see "Options" section.
(img,, alt: testcard2 src: ../Images/testcard2.png)



The menu contains four sub-menus: File, Edit, Layer and Tools (plus the 'About...' option which gives version information - currently there is also a HTML on-line help).

File menu: the usual New, Open, Save, Save As, Close, and Exit, plus Close All, Properties and Viewer.

Edit menu: Undo/Redo, Cut/Copy/Paste/Delete, Select All and Shift Set.

Layer menu: access to layer and grid properties, and manipulation of layers themselves (see "Multi-Layer Test Cards").

Tools menu: Colour Cruncher tool, ToneBox tool, and Options.

To save a completed TCD (or indeed a work in progress) select File/Save from the menu or press Ctrl+S. To save it with a new name select File/Save As or press Ctrl+Shift+S. These options save the required information to a .TCD file - note that this does *not* create a bitmap.

If you omit the '.tcd' extension it is added automatically. For new TCDs (indicated by "" in the title bar) File/Save behaves like File/Save As.

To open a previously saved TCD select File/Open from the menu or press Ctrl+O.

The File menu will also show the file names of the most-recently opened or saved TCDs (including from previous sessions). Choosing one of these options re-opens that TCD. Below the menu are a set of three buttons, followed by a row of nineteen.

The first two buttons each pop up control panels, allowing access to the properties of the whole card (as per File/Properties) or the current layer (Layer/Properties). The next button (arrow in box) allows the definition of a selection area on the current layer, by clicking and dragging with the mouse. If you then click an 'add object' button (see below) the new object's position and size default to that of the selection area. Alternatively if you click Edit/Select All or press Ctrl+A, only objects lying within the selection area are selected, rather than all the objects in the layer. The Select Area button stays down until one of these actions is taken or you click it again.

The button after this (box with illuminated rectangles) is Select All. If you have defined a selection area, only objects in that area are selected, otherwise all objects in the current layer are selected.

The remaining buttons allow objects to be added to the current layer. Each button has a schematic version of the object type; the status bar and tool-tip also indicate the object types when you move the mouse over the buttons. Clicking any of these buttons brings up a 'properties' control panel appropriate to the object type. Once an object has been added, its properties can still be altered, either by double-clicking on the object's 'active area' (see below), or by editing it from the Layer Properties panel's object list. In either case this brings back the Object Properties panel. This panel also has a 'Delete' button allowing the removal of objects. Note that if more than one object is selected, only the object you double-clicked is deleted, not the whole set.

Viewer

Displays the options specific to the Viewer.

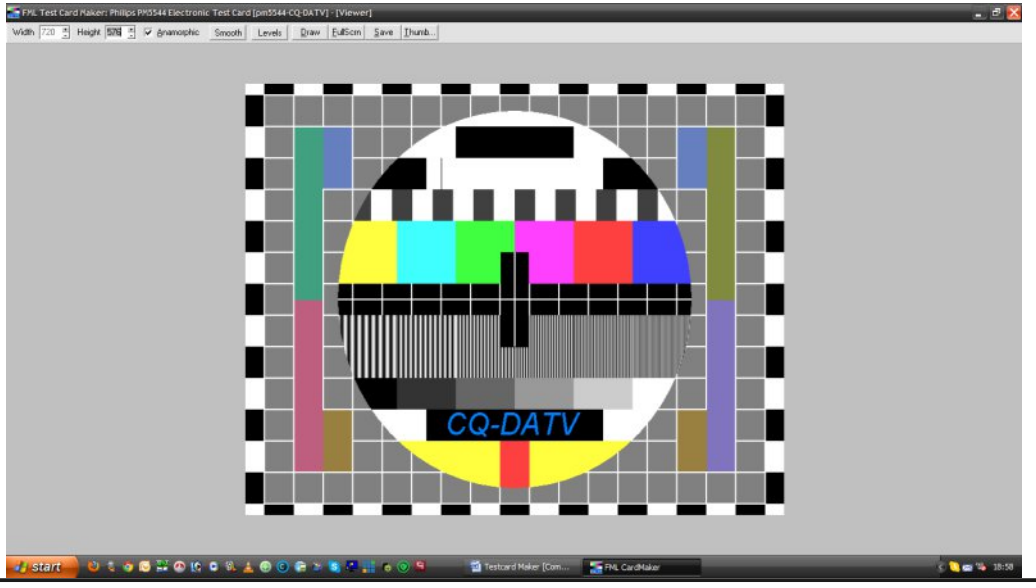
Start-up Mode

Click a radio-button to determine what happens when you invoke the Viewer:

Blank: Viewer does not draw the card automatically.

Drawn: Viewer draws the card automatically but doesn't go into full-screen mode.

Full: Viewer draws the card automatically and then goes into full-screen mode.

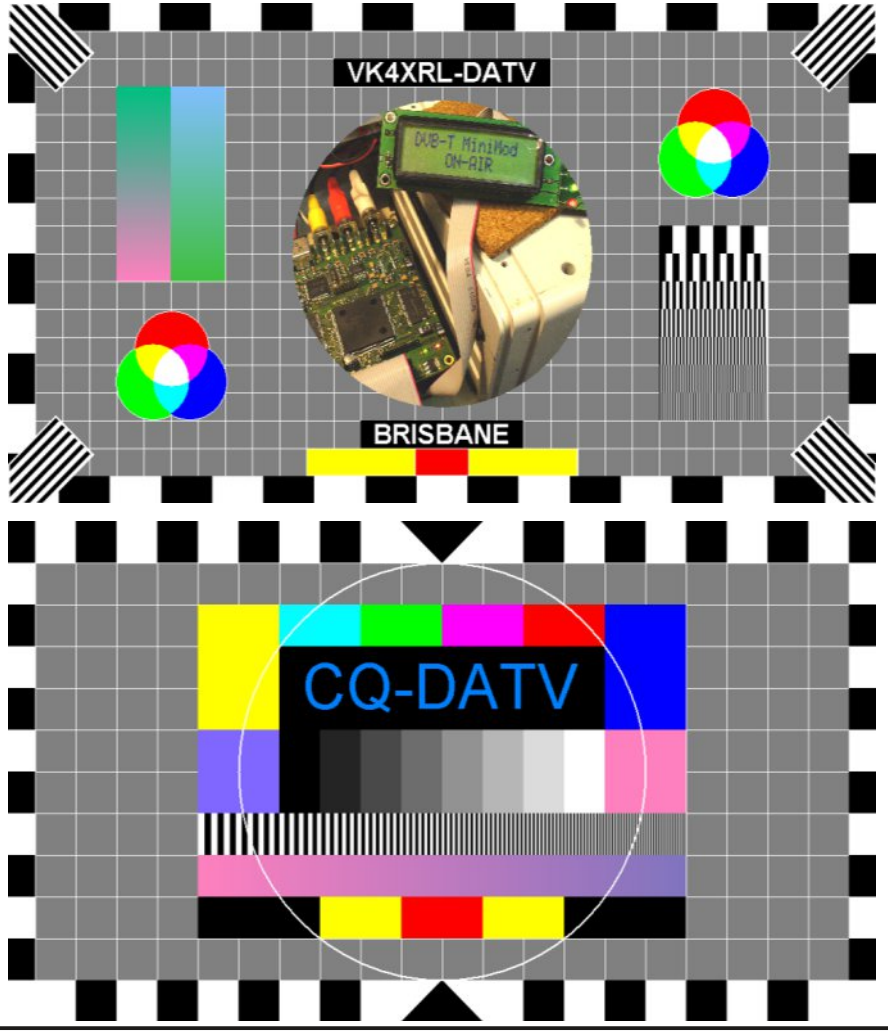


I could go on and place the whole Help file here but I won't, go and download it and give it a try! You'll find it at:
FML Home Page: *www.oodletuz.fsnet.co.uk* ***(link no longer functional. ED)***

The above section of the help file was originally created 23/02/2001 by Steve Heap.

Here are some Test cards that have been made using this software.

Testcards from Richard Carden Vk4XRL



External links

If you have an eBook reader that does not have WiFi then you will not be able to use the hyper-links in this publication. If you have an eBook reader that has WiFi then you will be able to providing you are in a WiFi zone.

But if you have a Kindle 3G then yes, but only to Amazon, and there is not a lot of ATV material on their site.
Smart phone reading apps are ok providing that you have a 3G data connection.

Note: These links will fire up your devices browser and if you are using 3G/4G then you will incur data usages charges.

Legal Niceties (the small print)

E&OE. Whilst every care is taken in the production of this publication, dotMOBI accepts no legal responsibility for the advice, data and opinions expressed. dotMOBI neither endorses nor is it responsible for the content of advertisements or the activities of those advertisers. No guarantee of accuracy is implied or given for the material herein. dotMOBI expressly disclaims all liability to any person in respect of anything and in respect of the consequences of anything done or omitted to be done wholly or partly in reliance upon the whole or any part of this publication.
As the regulations for the operation of radio frequency equipment vary in different countries, readers are advised to check that building or operating any piece of equipment described in dotMOBI will not contravene the rules that apply in their own country.
All copyrights and trademarks mentioned in this publication are acknowledged and no infringement of the intellectual copyright of others is intended.

Copyright

Notice to Contributors

Authors are alone responsible for the content of their articles, including factual and legal accuracy, and opinions expressed by them may not reflect the editorial stance of the publication. Material submitted to dotMOBI should not infringe the copyright of other writers or bodies.
Contributions are accepted for publication on this basis alone.
dotMOBI publications - *https://cq-datv.mobi*

Author Guidelines

CQ-DATV welcomes contributions from our readers. It does not necessarily have to be on ATV, as long as it is of interest to our readers.

Although a formatted article showing the layout can be sent, we prefer an unformatted text file of the script, along with annotations of where important images should be placed. All images should be identified as Fig 1 etc and sent seperately.

Images should be in PNG format if possible and the best quality available. Do not resize or compress images, we will do all the rework necessary to publish them.

If you are sending a construction project, please include the dimensions of any pcb's and make the pcb image black and white, not greyscale.

CQ-DATV reserves the right to redraw any schematics and pcb layouts to meet our standards.

Coming up in CQ-DATV

Is this the latest issue of CQ-DATV? *Click here* to go to our web site to check to see if there is a later edition available.

CQ-DATV is published on the last day of the month. The cut-off day for submissions/corrections/alterations is 5 days before the day of publication.

Please note that this mailing list is only used to advise interested people about the availability of new issues of CQ-DATV magazine. The list is not, and never will be, shared with any other organisations.

Want to be notified when issues of CQ-DATV are published? Join our *mailing list*.