In this issue

CQ-DATV dotMOBI

Like us on Facebook

Production Team

Ian Pawson - G8IQU Trevor Brown - G8CJS
Terry Mowles - VK5TM

Contributing Authors

Jim Andrews - KH6HTV Trevor Brown - G8CJS
Richard Carden - VK4XRL John Hudson - G3RFL
Terry Mowles - VK5TM Mike Stevens - G7GTN
Rudi - S58RU & Mauro - IV3WSJ

Editorial

Well we published CQ-DATV 29 on October the 31st which as everyone knows is Halloween or all Hallows Eve. Traditional activities, include Trick or Treating, well we did consider knocking on CQ-TV's door and shouting trick or treat, but were unsure what constitutes a trick if we were unhappy with the treat.

Costume Parties are also traditional, but Ian would not dress up, also traditional is visiting haunted houses, but the editorial team bottled out.

We looked at some special ATV related projects and came up with Johns Hubble bubble PCB etching tank.

Richard had a look at ATV repeater improvements, but never mentioned removing Ghosts.

Trevor dug into H265 compression and how to use it in post production, but not a mention of any artefacts that we could use in a good ghost story..So all in all we had to settle for a spooky cover and a few bats on selected pages.

We did an Internet search to find some spooky facts and came up with a NASA prediction that an asteroid is going to be flying closer to Earth than any has since 2006, and will miss us by a mere 499,000km. The asteroid was spotted by NASA a fortnight ago and is set to fly past the Earth on Halloween at over 78,000mph. It will cross by Earth at the closest it's been since 2006.

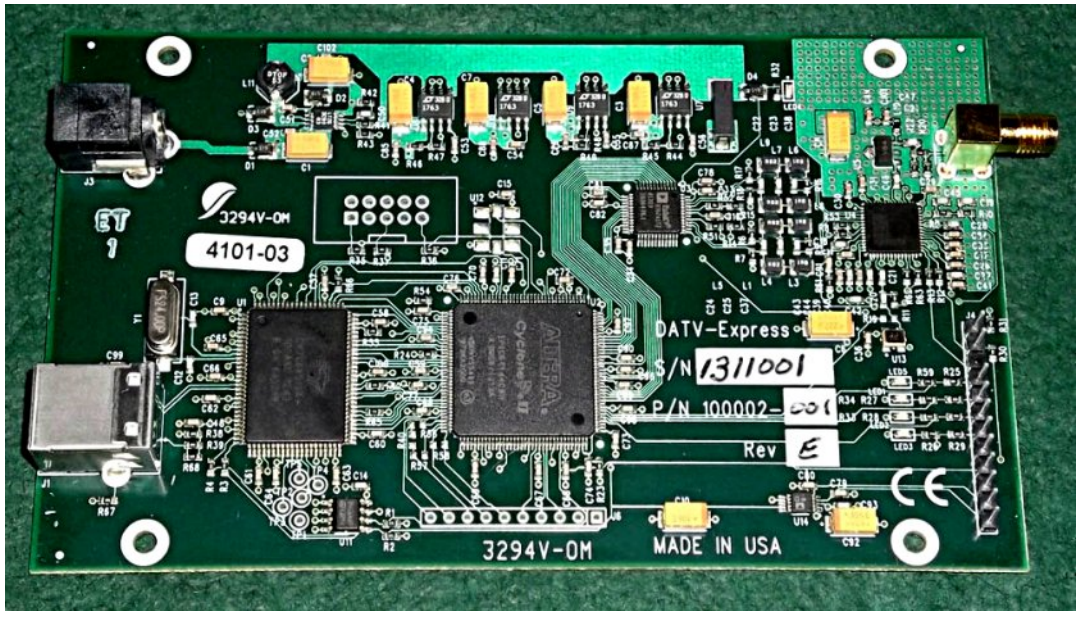
We don't expect any EAE (Earth Asteroid Earth) contacts at that distance and speed

So I think we all might be safe from this near miss and can safely settle down and enjoy CQ-DATV 29, you can always read it on an eBook with the lights out so the trick or treaters will think nobody is home.

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.

DATV-Express boards



The DATV-Express Project team has decided to order a third production lot of the DATV exciter boards. The last remaining board in inventory was shipped to Japan this week. Art WA8RMC is now in the process of ordering an additional 25 units to replenish the team's inventory. If the ordering of components and assembly goes as planned, tested boards should be back in inventory by November 15th.

Visit *http://www.datv-express.com/Purchase/ShowBoardInfo*

73…de Ken W6HHC

Ofcom to revoke unvalidated ham radio licences

The UK communications regulator proposes to revoke amateur radio licences which have not been revalidated

UK Amateur Radio Licences remain valid unless surrendered by the licensee or revoked by Ofcom. It is over 9 years since Ofcom last revoked an amateur licence for any reason.

Ofcom require holders to revalidate their licence every 5 years but until Ofcom goes through the formal process of Revocation such licences remain valid.

It is suspected there might be tens of thousands of amateur licences which have not been revalidated. Ofcom proposes to start with those that should have been revalidated in 2012.

If Ofcom wishes to revoke a wireless telegraphy licence, it must:

- *notify the licensee of the reasons for which it proposes to revoke the licence;*
- *specify a period in which the licensee may make representations;*
- *decide whether or not to revoke the licence within one month of the end of that period;*

and

- *notify you of its decision within a week of making it.*

Ofcom announcement
http://licensing.ofcom.org.uk/radiocommunication-licences/amateur-radio/revalidation/

How to revalidate your licence
http://www.essexham.co.uk/validate-your-licence

CQ-DATV dotMOBI

G4CQM Yagi Antenna Projects 2015

Derek Hilleard G4CQM has just concluded work on his 2015 yagi projects website with the addition of a 13 element 4 wavelength high performance design for 144MHz.

Homebrew yagis have become very popular for many reasons and Derek has posted several pictures of successful builds.

Due to the level of interest at peak times two servers now host G4CQM designs, US (LA) and EU (Herts).

http://www.g4cqm.co.uk/

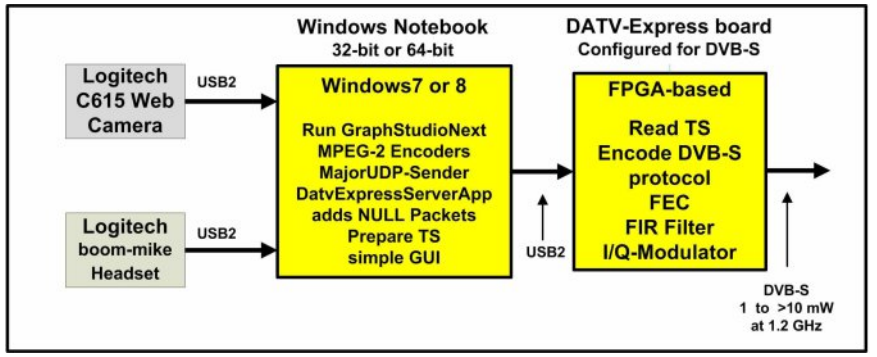
DATV-Express App Software for Windows

The DATV-Express Project team has uploaded a zip file called DATV-Express-Utils_v1.2 with the two apps up to the DATV-Express Project web site at *www.DATV-Express.com* . Look for the zip file on the DOWNLOADS page.

The two apps created by Charles G4GUO for the DATV-Express board are:

- *DatvExpressServerApp is an alpha-release program to transmit DVB-S DATV running only on a windows PC with the hardware board. (no linux)*
- *DatvExpressSdrApp is an alpha-release program running only on a windows PC to transmit FM or SSB using the hardware board (also no linux).*

PLEASE NOTE - these two applications are experimental and are likely to be full of bugs. They are provided as a service to the DATV-Express community. There are three README files included in the bundle...and a brief User Installation Guide is planned as time permits.

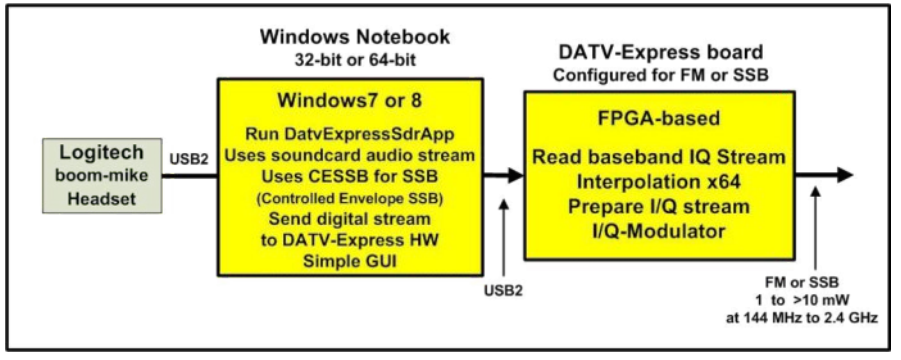


Typical Block Diagram for DatvExpressServerApp running DVB-S

There is a little more information about both of these two apps that can be found in my ARRL/TAPR DCC2015 presentation that is downloadable at *www.W6ZE.org/DATV* OCARC website (click on the PRESENTATIONS link).

Finally, there are more details about the DatvExpressServerApp for DVB-S that can be found in my DATVtalk13 article in the free eMagazine CQ-DATV issue 27, go to *http://cq-datv.mobi* The DATVtalk13 article is titled "Webcams and UDP and DatvExpressServerApp on Windows". But, you are essentially on your own...so please do not expect much in the way of support.

73...de Ken W6HHC

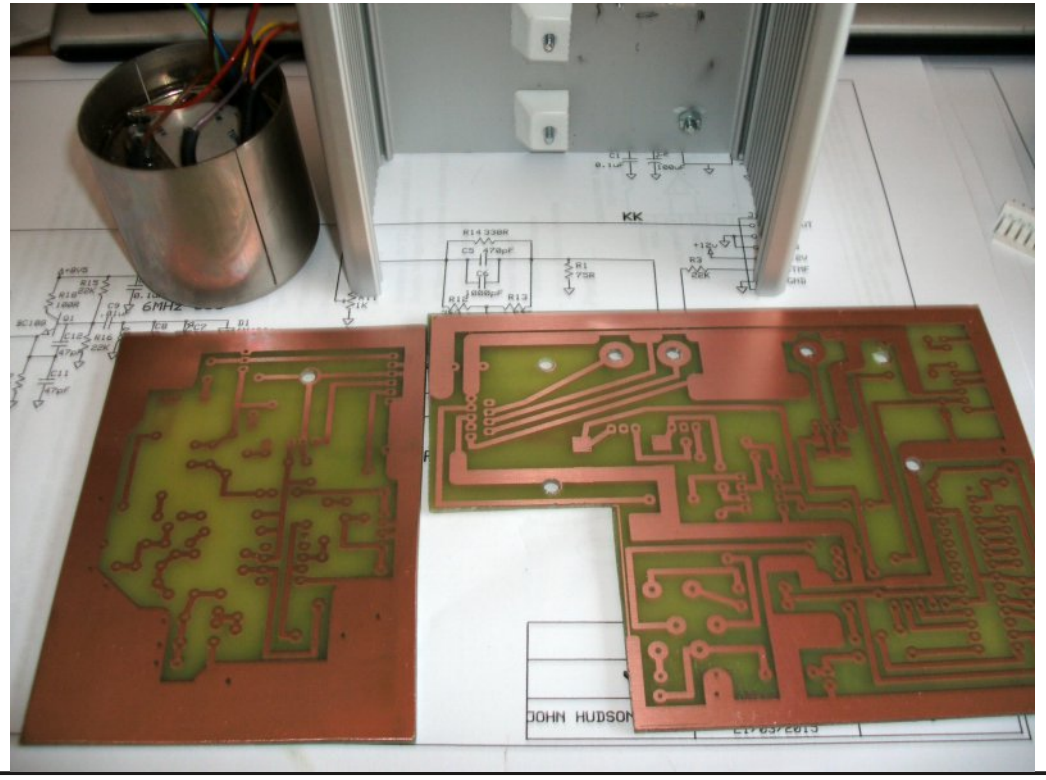


Typical Block Diagram for DatvExpressSdrApp running FM or SSB

Hubble Bubble Etch Tank

By John Hudson G3RFL

As you will have gathered from all the constructional projects I have produced for CQ-DATV I am great fan of homemade PCB's. They are worth the trouble and effort, that you inevitable encounter and are perhaps for RF the only way to go. I photo etch the PCB's using pre-coated PCB material, the artwork is contact printed into place using a UV source and then the PCB is etched in Ferric Chloride, and the holes drilled and the board populated.



WARNING NOTICE: the fumes are dangerous and so need a well ventilated room and they will also cause tools to go rusty. My system is in a shed and I leave the door open before and after for some time.

To develop the photo board I just use 2/3 teaspoons (plastic) per pint of caustic soda. When the copper looks clear I wash off with water. Develop time is affected by room temperature. Note: must be caustic soda and not drain cleaner with caustic soda.

The weak link has been the ferric etching and I thought it was about time to throw some serious money at a professional etching tank so as to speed up the process and remove some of the hassle factor my old system put into this production chain.

First stop was a browse on line at the professional products in the AC Farnell/CPC catalogue and very nice they are too, until you come to the price £610....Just to warm up a little Ferric Chloride.

I might just pass on this purchase or do a little shopping around for a less expensive alternative.

My first stop on the less expensive trail was one of my favourite stores "Pound Land". The prices were considerably better, but let's not kid ourselves, we are not buying like for like and there may be one or two items required that the price competitive emporium does not stock.

The first component was the tank and yes it does look a little more basic, but at a £609 saving.

If only it came with a heating element and all the sexy controls! The heating element was my next investment and unfortunately did not come from an establishment that operated the same pricing policy.

This was a shop selling tanks for tropical fish. The heater cost £15 for this rather excellent 50watt quartz heater unit, which is brilliant at warming up the etching fluid .

The suckers on the tank base were another £1 and yes, you have guessed it Pound Land, along with the Araldite to secure them in place and glued into the bottom of the tank so as to tie-wrap the air pipes so they did not float to the top of the tank.

All that is now required is some form of agitation (no I said agitation, not aggravation, there is never a shortage of that!). This is the process of blowing bubbles into the tank to move the solution around and to aerate it for a quicker, more even etch.

The air pump used was salvaged from the original etch tank. It again came from the tropical fish dealer along with some plastic tubing. I sealed the end and drilled 1mm holes in the tube at about 10mm spacing. The tube is snaked across the tank bottom and creates movement in the etching solution.

The first PCB I etched took about 10mins (I let the tank get well and truly hot first). The etch was much more even than the old tank, with all the parts of the PCB etching at the same time. I tend not to produce a lot of double sided PCB's as I am really an RF man and the reverse side of the PCB is usually reserved as a ground plane.

Am I pleased? Yes. It does not look as sexy as the CPC commercial product and perhaps has a slightly extended etch time. Often as long as 30mins if I don't allow the tank to reach optimum temperature before I start the process. But then I am probably only etching 1 or 2 PCB's a week. I am not in a rush and I assume people who purchase the CPC tank have considerably more throughput and time is an important consideration.

The most important factor was an even etch, which is vital for preserving finer 1mm artwork mixed with sections of the same PCB where large areas of copper need to be removed.

This was achieved and it was not all about cost saving it was about home construction not only for the projects, but for the kit used to produce them. I get a kick out of designing and building and I try to bring that to others. I think it would be a dull hobby if all we do is buy kit, plug it in and see what comes down the aerial. To this end I thank CQ-DATV for all their help and support in spreading my ideas around the world.

There are numerous approaches to etchant tanks and I have included a couple of useful links but the Internet is full of inexpensive designs. Just let your fingers do the walking and don't forget to feed anything you build back to the editor. I am not holding my tank up as the ultimate design, just something that works for me.

https://www.youtube.com/watch?v=DXjWZ9gJ4XE
http://www.embeddedtronics.com/etchant.html

ONLY YOU CAN TAKE HIS PLACE

We were saddened to learn of the death this week of one of our most valued members — Someone Else

Someone's passing creates a vacancy that will be difficult to fill. Else has been with us since the very beginning.

He did far more than a normal person's share of the work. Whenever there was a job to do, a function to attend, orders to be filled or a meeting to cover, one name was on everyone's lips, "Let Someone Else do it".

It was common knowledge that Someone Else was among the largest contributors of his time. Whenever there was a need for volunteers, everyone just assumed Someone Else would volunteer.

Someone Else was a wonderful person — sometimes appearing superhuman but a person can only do so much. Were the truth known, everybody expected too much of Someone Else.

Now Someone Else is gone! We wonder what we are going to do. Someone Else left a wonderful example to follow, but who is going to do the things Someone Else did? When you are asked to help, remember — WE CAN'T DEPEND ON SOMEONE ELSE ANYMORE.

Original Source Unknown.

Quad DVB-T Receiver

Jim Andrews, KH6HTV

This is an edited version of an application note on http://kh6htv.com/ and is reproduced here by kind permission.

This article gives the details on the design of a Digital TV (DTV) receiver to receive and display simultaneously the DVB-T transmissions from four separate, 70cm, DTV transmitters. It is used by BCARES to provide TV support for local police and fire public safety agencies.

The Boulder County, Colorado ARES group, BCARES, has been providing TV coverage since 1995 of the University of Colorado football games for the CU Police Dept. The TV images are not of the game on the field, but of the crowd of up to 50,000 spectators for public safety purposes. BCARES also provides TV coverage for other public safety agencies, including the Sheriff, and various police and fire departments in the county. Other events covered have included: large forest fires, floods, major foot races (50,000 runners), public protests, student riots, and SWAT operations.

When BCARES initially started in 1995 working with the CU-PD, a single TV camera image was provided. It proved so popular with the police command staff, that they soon asked for even more cameras. This was first done on a single, 70cm, channel by switching on and off separate TV transmitters. Soon the police asked to see all the cameras at the same time. BCARES then drew the line at four cameras as we had run out of available 6 MHz channels in the popular 70cm band. We used channels 57, 58, 59 and 60. This was a technical challenge to accomplish on four adjacent channels without co-channel interference. We were using AM-TV transmitters, originally from PC Electronics and later from VideoLynx.

Fig. 1 BCARES TV camera crews at a University of Colorado football game using DVB-T, digital TV transmitters.
Left to right: George, KA0BSA, Steve, WB0NFQ, Jim, KH6HTV, Ron, K2RAS and William, KD0YYY

We installed a Spectrum International, inter-digital, 6 MHz, band-pass filter on the output of each transmitter. We also used identical channel filters on the inputs to each receiver. Initially, we provided the images on four separate TV receiver/monitors.

However, these took up too much space in the command post. The next improvement was to build a Quad TV Receiver box. We used four separate, commercial grade, cable TV head-end receivers. They were Pico-Macom model MPCD ($140, available from ATV Research). The composite, NTSC, video outputs from these receivers were then fed into a quad processor which combined the four inputs into a single NTSC, composite video signal with each image displayed in one of the four quadrants on a single video monitor screen. When desired, with the push of a button, a single image could be selected and displayed in full screen mode. This analog system was used successfully for many years by BCARES.

While the analog TV system worked, we were almost never able to provide true, broadcast quality pictures for our public safety agencies. Running low power (1 watt), portable operations, typically from a back pack transmitter with flexible whip antennas, we were always fighting the issues of low signal strength and multi-path with the attendant display issues of "snow" and "ghosting".

In the fall of 2014, Jim, KH6HTV, brought his new digital TV transmitter and receiver to the first CU home football game for a demonstration. His equipment used the European terrestrial DTV standard, DVB-T. The CU-PD police commander was extremely impressed with the performance.

Finally BCARES was able to provide absolutely perfect quality TV pictures, with no snow nor ghosting, plus we were able to provide pictures from some parts of the stadium which had never been possible previously. They also had the added bonus of being in true, 1080P High-Definition!

Additional details about this can be found in the June, 2015 issue of QST. See pages 42-44, "DVB-T: A Solution for ARES TV Operations"

The CU-PD made the decision to fund a complete switch-over from the old analog TV system to the new DVB-T system for the Fall, 2015 season. $10,000 was allocated in funds for new DVB-T transmitters and a Quad DTV Receiver. New cameras were not necessary as we were already using hi-Def Sony camcorders. The BCARES coordinator for CU-PD, Mark Huff, K0LRS, took the lead to order all necessary parts and assemble the systems. This included Mark designing and building a totally new QUAD-DTV Receiver. Mark has received a lot of technical assistance from Matt Holiday, K0DVB, our local resident DVB-T expert.

Fig. 2 The new BCARES Quad DVB-T Receiver and live, hi-definition images at a CU football game received on channels 57, 58 & 60. Note: ch. 59 is also possible

Figures 2, 3 & 4 show the new Quad DTV Receiver. Figure 4 is the block diagram of the major components. The key to making the entire system possible is the HDMI Quad Processor.

Fig. 3 Close up of the front of the new Quad DVB-T Receiver. Key items seen top to bottom are the 2m FM transceiver, the HDMI quad processor, 4 Hi-Des DVB-T receivers, and 4 Spectrum International band-pass, channel filters.

After a search on the internet, Mark was able to find one at: *http://www.ambery.com/4hddvispqupi.html* It was called a "4-Channel HDMI DVI Split-Screen Quad PIP Video Matrix Switch", Model: PIPV400H and cost $1,670.00

The four DVB-T receivers are from Hi-Des Technologies, in Taiwan, *www.hides.com.tw* . They are the model HV-110 and cost $169 each. To ensure the system works reliably in a high RF, EMI situation, such as a large football stadium, and to minimise co-channel TVI, high-quality, 6 MHz bandwidth, band-pass, channel filters are required on the input of each

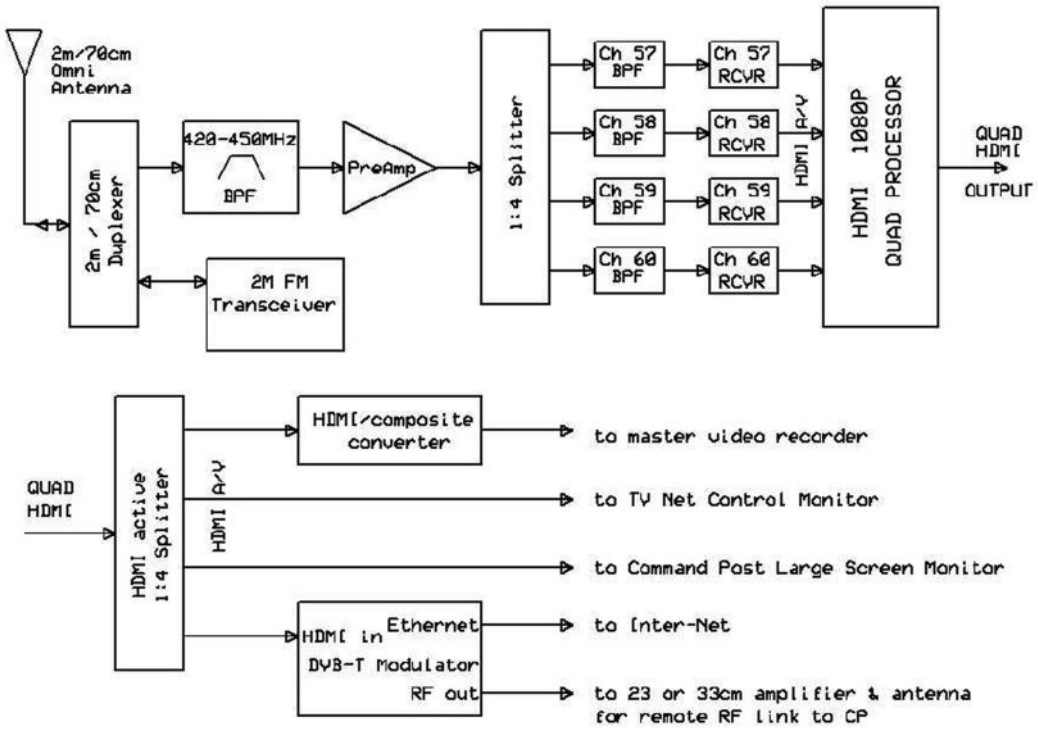


Fig. 4 Block Diagram of BCARES, Quad DVB-T Receiver

receiver. The BPFs used were salvaged from the previous BCARES analog quad TV receiver box. They were built by Spectrum International. SI is no longer in business and these filters are now only available on the surplus market. Suitable replacements are now available from Don, N0YE, and marketed by KH6HTV Video, *www.kh6htv.com* as the model 70-BPF-5P-6BW-XXX ($325).

The input DTV signals are received on a dual-band (2m/70cm) omni-directional, vertical antenna. We recommend the Diamond X-50N. They first pass through a 2m/70cm duplexer (Diamond MX-72H, $50). Additional RFI protection filtering is provided by a 70cm (420-450MHz), 30 MHz bandwidth, Spectrum International, band-pass filter. The DTV signals are then amplified in a low noise (0.5dB NF, 18dB gain), GaAsFET preamplifier (Advanced Receiver Research,

www.advancedreceiver.com , model P432VDG, $90). The amplified signals are then split four ways (-7dB loss) in an ordinary CATV, 1 : 4 splitter and sent to the four channel filters and DVB-T receivers. The A/V outputs from the receivers are HDMI. These go to the HDMI quad processor.

The quad processor only provided a single HDMI audio/video output. To provide additional HDMI outputs, a 1 in / 4 out active HDMI splitter was used. These are available on the Internet from many sources at low cost (< $50). An NTSC, composite, quad video signal was provided using an inexpensive HDMI to composite converter. An additional feature of this receiver was the ability to send it's quad video image to other remote locations, either via an RF link and/or the Internet. This was provided by using a Hi-Des, model HV-100EH, DVB-T modulator ($570).

A 2 meter, FM transceiver was included in this quad DTV receiver box. It is used to provide intercom communications between the TV net controller and the remote TV camera crews. It shares the 2m/70cm omni antenna via the duplexer. Other miscellaneous items needed included +12Vdc and +5Vdc power supplies, cables, etc. The entire receiver was packaged in a rugged, portable, 19" rack mount cabinet, Gator model GRR-8L, available for about $250 from *Amazon.com* . The overall cost of the entire receiver, excluding labour costs and antenna(s), is approximately $5,500.

Check out the DKARS website at:-
http://www.dkars.nl/

Understanding Repeater Housekeeping

By Richard Carden - VK4XRL

What is repeater housekeeping you may ask?

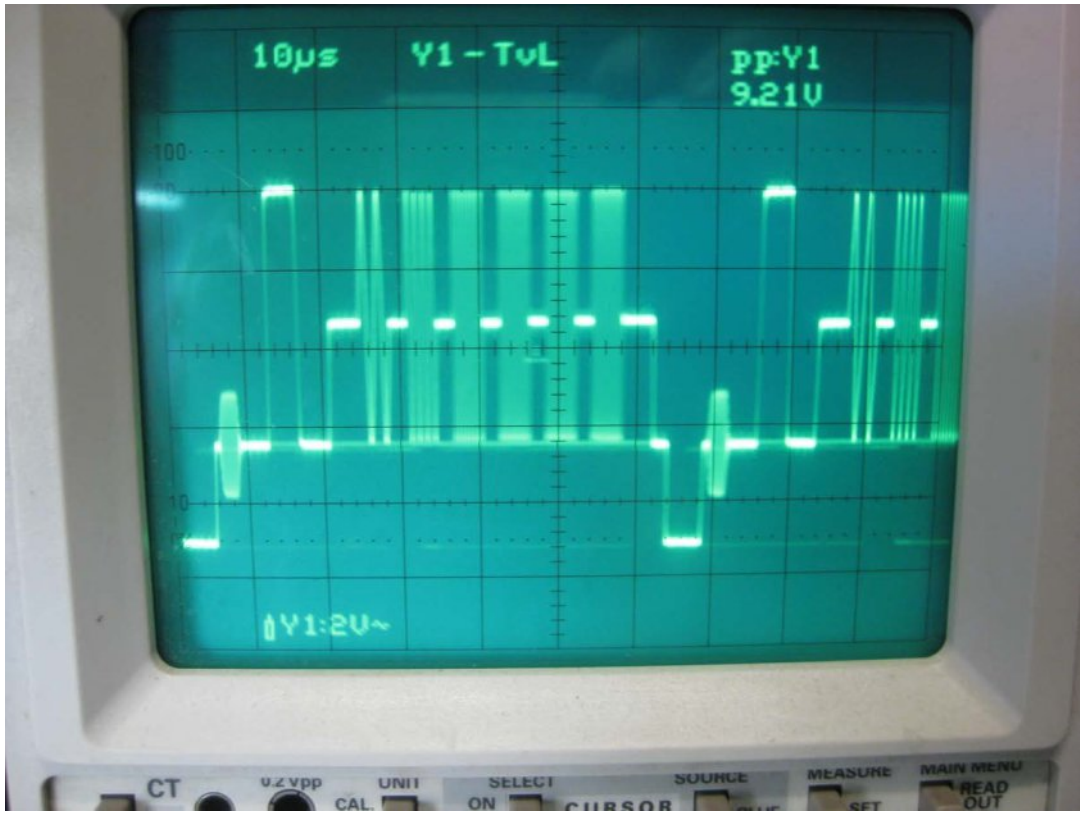
Well it's maintaining system specifications in regards to audio and vision parameters. It's interesting to note that when you look at different controllers that have been presented that that not all have a reference setup especially regarding audio.

Vision in most cases is easy with a 1V p/p test signal. Having spent many years been responsible in maintaining procedures to provide high quality audio and vision passing through a station system form inputs to outputs it makes for an easy task when switching etc. between sources.

This can be utilised in a repeater system as well especially now most repeaters are digital. In fact it is an advantage as it allows for people that haven't all the right gear to set levels to within reasonable levels.

As a practical example let's look at the changes for VK4RMG over the past few months. The initial controller was provided by me in a nominal 3 RU setup. This consisted of controller, WD Testcard Generator, Sony switcher and associated VDA and SDA including LED level monitoring plus calibration. This system had been working for years giving good service with the required audio/video levels. The input levels for FM and DATV where then set via the normal audio controls.

I was then asked to look at a 1 RU system for future use in repeater sites with limited space. We looked at a Sanyo video/audio IC and checked out the normal problems like crosstalk which was found to be ok. I looked up LOS IC's and this also worked fine. A drawing was produced by another member and I started to make up a prototype using a board made for the proposed circuit.

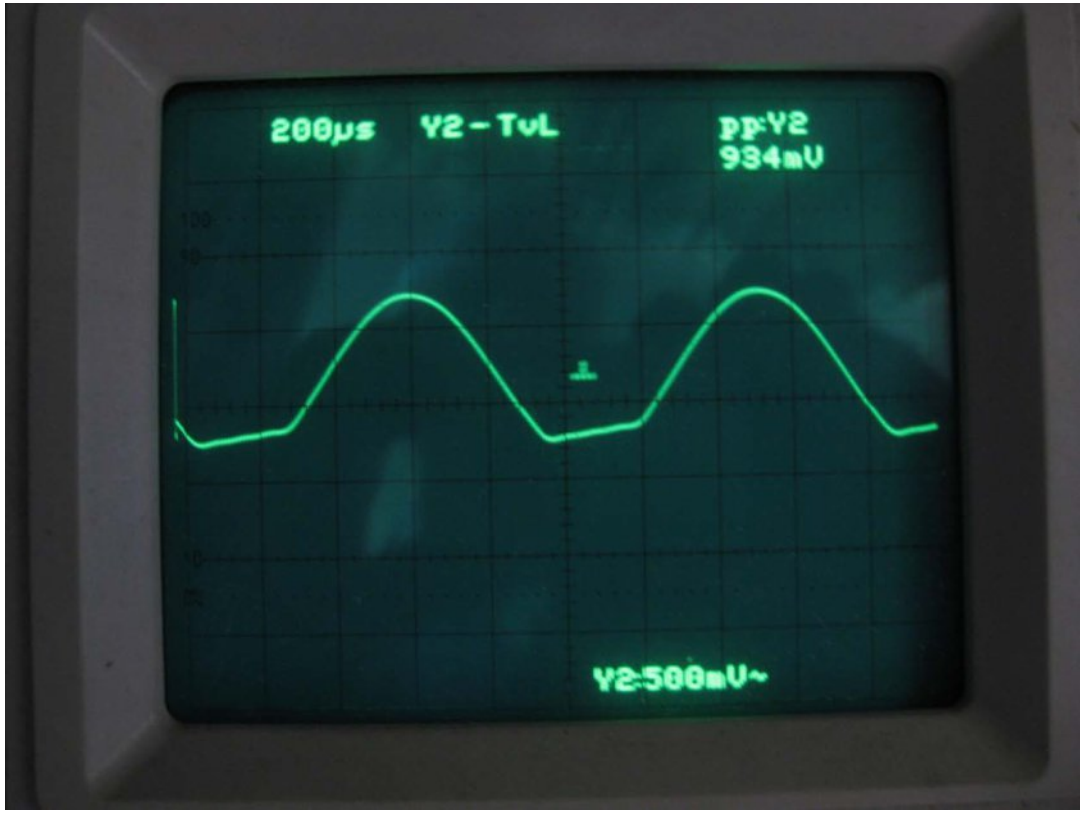


I started with one input which fed testcard and also audio levels from the WD media player. I'll work through the vision circuit first noting that the internal amplifier of the switcher IC had a gain of 6db so that with the circuit presented to me I had a problem as the output MAX497 also had a gain of 6db.

The original circuit I used from the switcher IC was an emitter follower 75 ohm output to a 75 ohm termination resistor. This arrangement did cause some sync clipping however but I remembered that Mike Cox had an article in CQ-TV on his sort of problem so I lowered the emitter resister value. While this worked to a degree we still had a repeater with sync crushing which was a no no. Keyed clamping would have solved that problem however the solution I found was to provide a resistor divider that fed 1V p/p to the base of the emitter follower from the switcher IC.

The output from the emitter follower was then fed via a capacitor and a diode providing sync tip clamping and added bonus. From the photo you can see that the mutiburst is passed through unaltered.

Audio Problems.

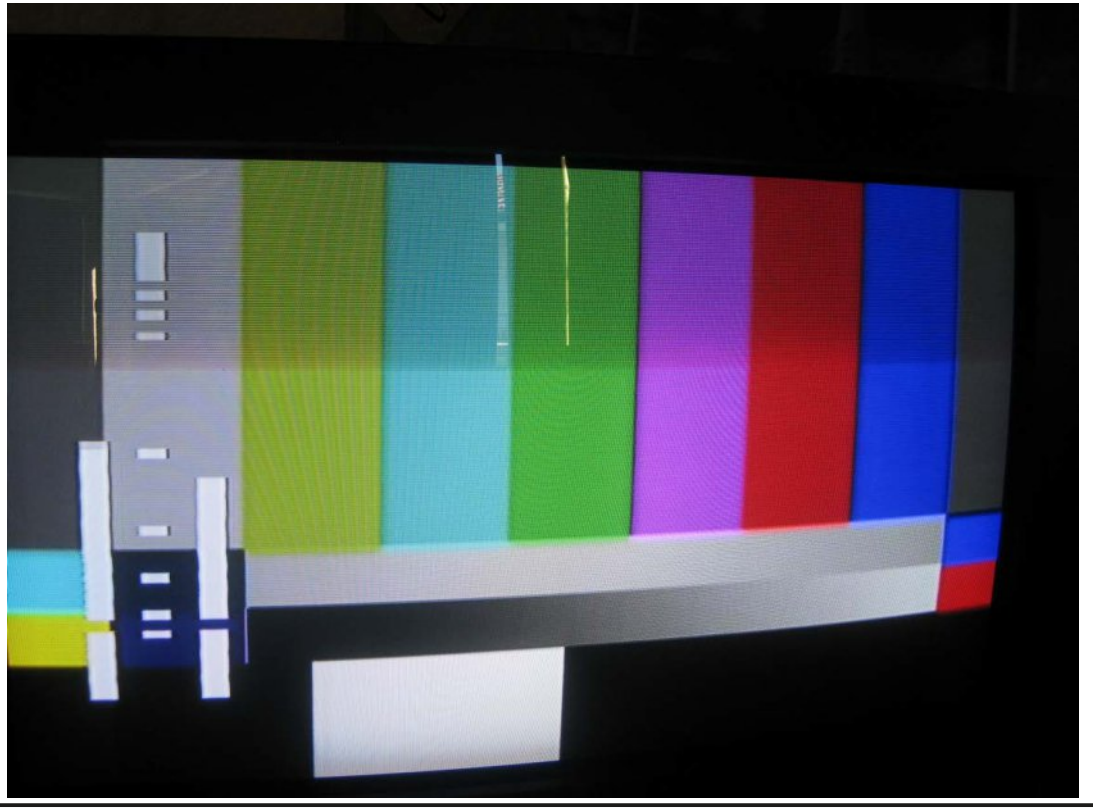


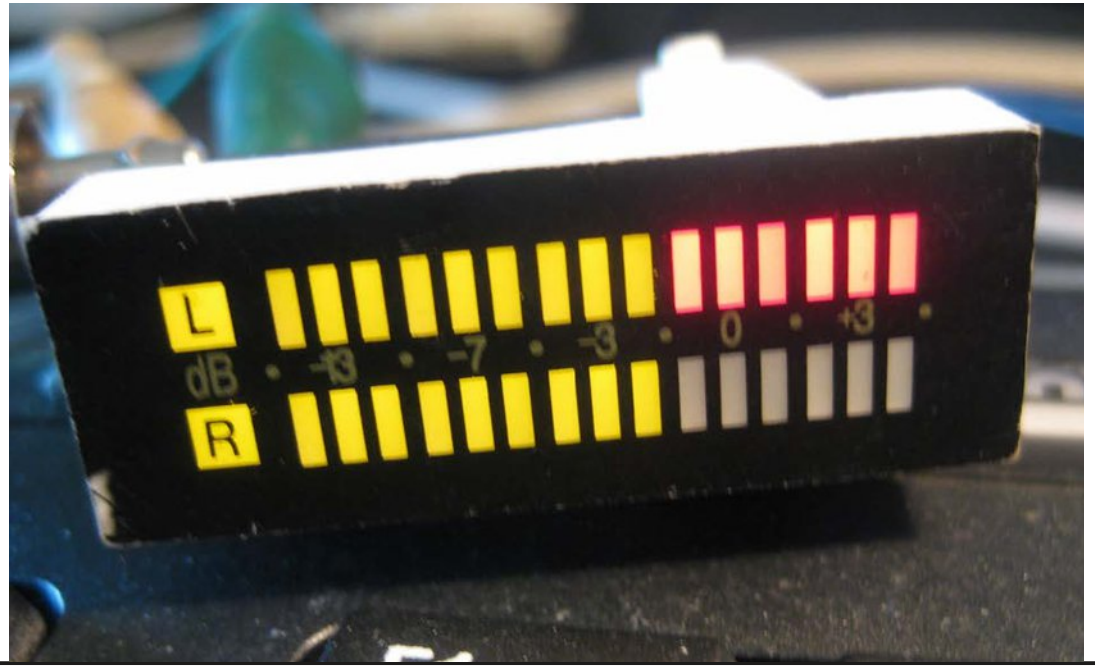
Now when I setup the prototype it wasn't intended to slot into the existing system. The overall gain of the old system was different to this newer 1 RU unit. For repeater operation two things were missing (1) a reference audio tone that inputs and outputs could be set to and (2) a VU meter to set and adjust them to after calibration. In the older system that was stereo a tone was provided from the WD media player and once all had been calibrated could be switched off. Again the audio output from the Sanyo IC had a gain of 6db (x2) and also the output buffer amplifiers had a gain of 6db (x2) a total gain of x4. The output buffer stages are run from +/-5v giving around 8 to 9 v p/p. The photo below shows the output from my STB looking at the RMG repeater output from receiving my DATV signal. The other problem is that the left and right levels don't match as shown by the OSD. It was just a simple modification to change the audio buffers to a gain of one.

The FM audio levels where far worse.

It is far better to use a peak reading meter when working with digital signals, and using the VU meter is ok on the analogue FM signals only.

The prototype switcher now feeds my SD channel on my 446.5 MHz digital transmitter. The following levels are what I have in regard to that system.





- Tone from WD 1.95v p/p

- Tone from controller 3.78v p/p

As there is no audio level control within the controller I still use an external SDA where the output is set to around 2.28v p/p these are normally run from +/-12. The controller reference indicator is set to zero looking at the controller output. It is therefore a simple matter to receive the DATV and FM receiver outputs from a correctly setup transmitter and then adjust the audio output levels for reference as shown on the meter.

A simple digital meter can be made using a BA6137 while an analogue unit can be build using a bridge with BAT 46/48 schottky diodes with a variable potentiometer on the input to calibrate the unit.

4K video and H265 compression

By Trevor Brown - G8CJS

H265 (x265) compression is with us. It delivers almost 50% more compression than H264 and was developed for the American cable networks.

The result is better picture quality for a given streaming data rate, or less bandwidth for a given picture quality, depends which way you look at it.

Yes, like any new technology that comes along, we have yet to get our hands on the kit to evaluate it, from editing to DATV transmissions. I walked into it rather by accident when I purchased a new Camera, the Samsung NX500, which I have documented in previous issues.

This is a bridge camera, which means it will shoot video and stills, in the video mode it can shoot 4k resolution pictures 4096 × 2160. That's twice the horizontal resolution and twice the vertical resolution of HD TV.

It goes without saying that this is a lot of data and one of the problems is putting all this data onto a memory card. So by using H265 compression the data stream is reduced so that the memory card can cope. The cards are the weakest link, so always buy the fastest (sequential rather than random).

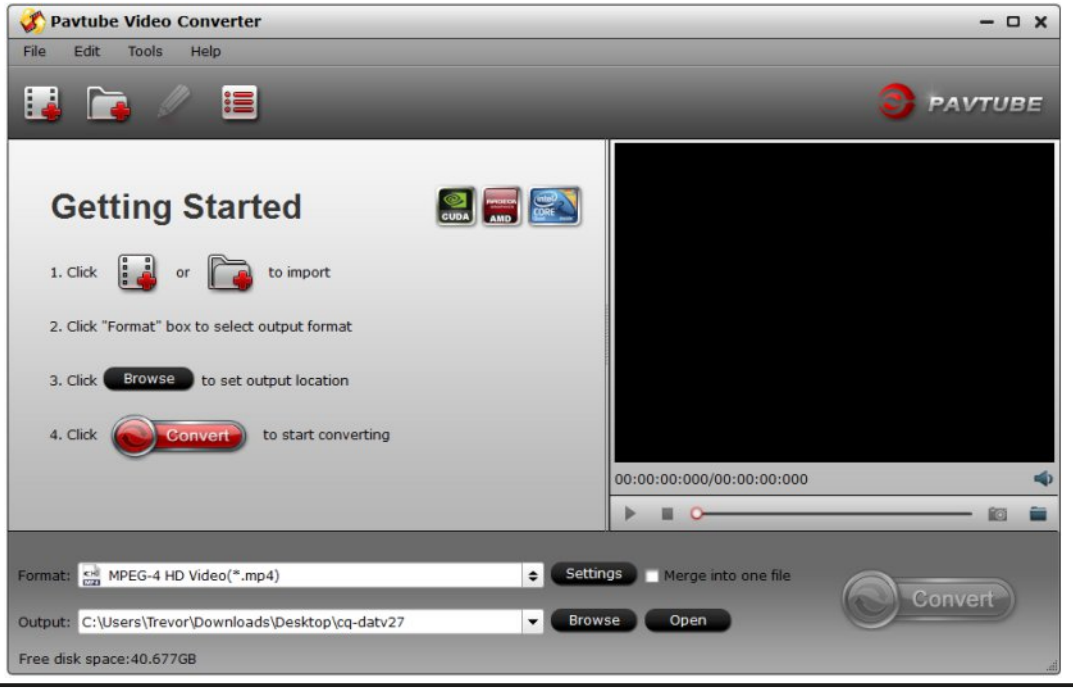
The H265 coded pictures can be viewed on the camera as normal and even on the home TV down the HDMI adaptor lead. But thinking about it, HDMI is parallel data so coding will not have been applied at this stage.

The memory card can be downloaded to your PC, but you then need an H265 viewer and editor or a converter (transcoder), to use existing software for editing, while we wait for technology to catch up.

The first transcoder software I tried was handbrake (well it is a free download) *https://handbrake.fr/* it's not the most friendly of software to use, but you can drop and drag clips into it and set up a queue, but there is a delay between adding each clip individually to the queue, before the inevitable delay before it transcodes all the clips. Yes it works, yes it is free, but there has to be a better way and yes there is and it's called Pavtube (Windows and MAC only, unlike Handbrake which also has Linux support - ED).

Pavetube is not free but there is a demo version so you can try before you buy *http://www.pavtube.com/* . The demo version does leave a very small watermark on the conversions, but if you are just using it to view clips this of course does not happen.

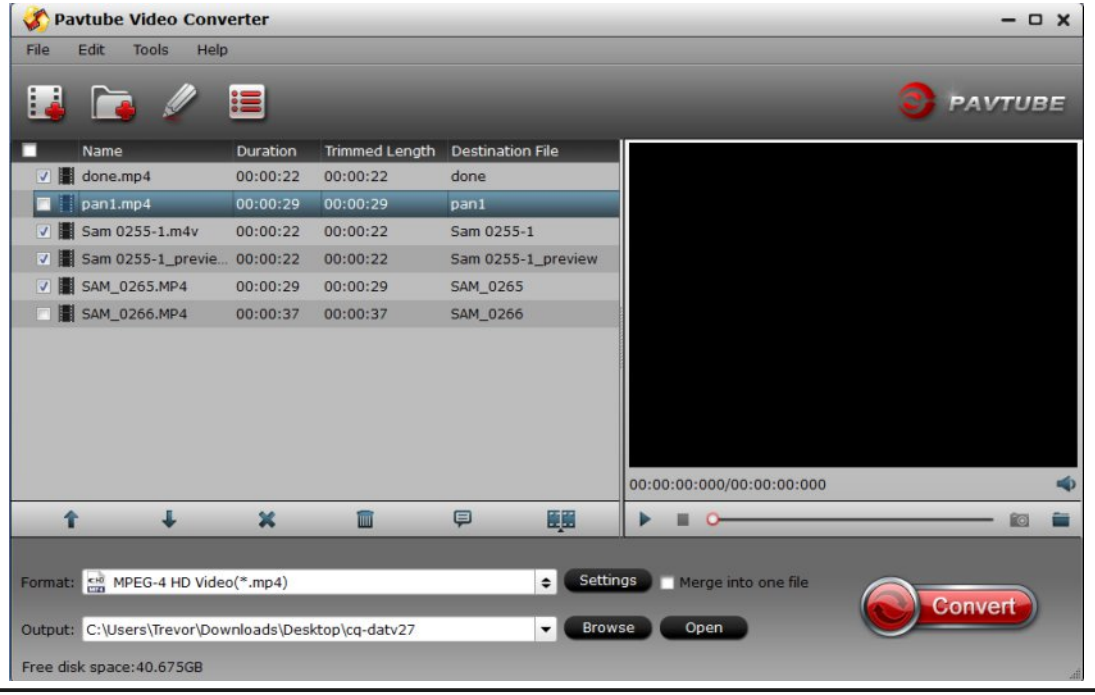
I downloaded the demo and transcoded all my clips with a watermark and then edited them together in Speed edit, my preferred editing software, that will not cope with H265 (I tried it).



This is the opening screen of Pavtube which allows you to import a single file or folder and set the conversion format and set up the output folder.

The way I organise my workflow is to set up a folder with the name of the project and a sub-folder called rushes where I copy the H265 files from my SD card. These are all on a separate hard drive within my computer called V drive. The reason is when you use hard drives for video they seem to have a short life, two to three years. I don't wait for them to grind to a halt, I replace them and usually gain from their ever increasing speed and size (my current drive it 3TB). This is so much easier than replacing a windows drive. I just connect in the new drive, copy the material across, remove the old drive and then change it from the temporary dive letter I assigned to it back to V, so the editing software does not see any difference and all the folders have the same address on my new drive.

I imported the rushes folder into Pavtube.



The clips are listed on the left and can be highlighted and played on the viewer on the right. The wanted clips can have the tick left in the box on the left and the unwanted clips have their tick removed. In this way we have carried out the first stage of editing without even opening the editing software. We have identified the clips we do not want to use.

This makes for an ideal viewer and only the wanted clips leave the rushes folder into the working folder that will become the clip bin for the edit.

This is also where I walked into my first problem. The player stutters when it plays H265 clips! They are very heavily compressed and require considerable resources to play. I am running a quad core CPU and the hard drive is a new 3TB Western digital drive. I contemplated replacing it with a solid state drive to solve the problem, but then our son turned up with his state of the art MAC laptop. Yes it's a laptop but with a 2.4GHz quad processor and a solid state drive, so we downloaded the MAC version of Pavtube (Yes it is cross platform software) and it performed about the same as my PC with its quad processor and mechanical hard drive.

There might be improvements that can be made to the graphics processor. I know mine is not top flight, but as yet I am putting up with the stutters. It's early days and an ideal solution would be to work with the H265 files without trancoding, but I suspect the problems of previewing them in Pavtube may also exist in scrubbing and manipulating the files in any editing package.

To this end we need to devolve a strategy for dealing with the complex H265 files, and I might just have the answer. I tried my solution on the watermarked clips which are really the equivalent of a cutting copy used to edit film. The video edit using these clips took place and I could check all the shots worked without investing in Pavtube, which I think was around £30, (shows how mean I am).

I then purchased a full version of Pavtube and ran the rushes through again and created a new folder. So I have two folders, watermarked clips and un-watermarked clips.

The editor makes an EDL (Edit Decision List) which has the location of the clips and the cutting instructions. The folder being the watermarked clips. If I now rename the folders in windows so the watermarked folder becomes unwanted and the un-watermaked folder becomes watermarked. I then initialise the editing software, load up the EDL list, which pulls in the un-watermarked clips and cutting instructions and I can render up a video which is now made of un-watermarked clips.

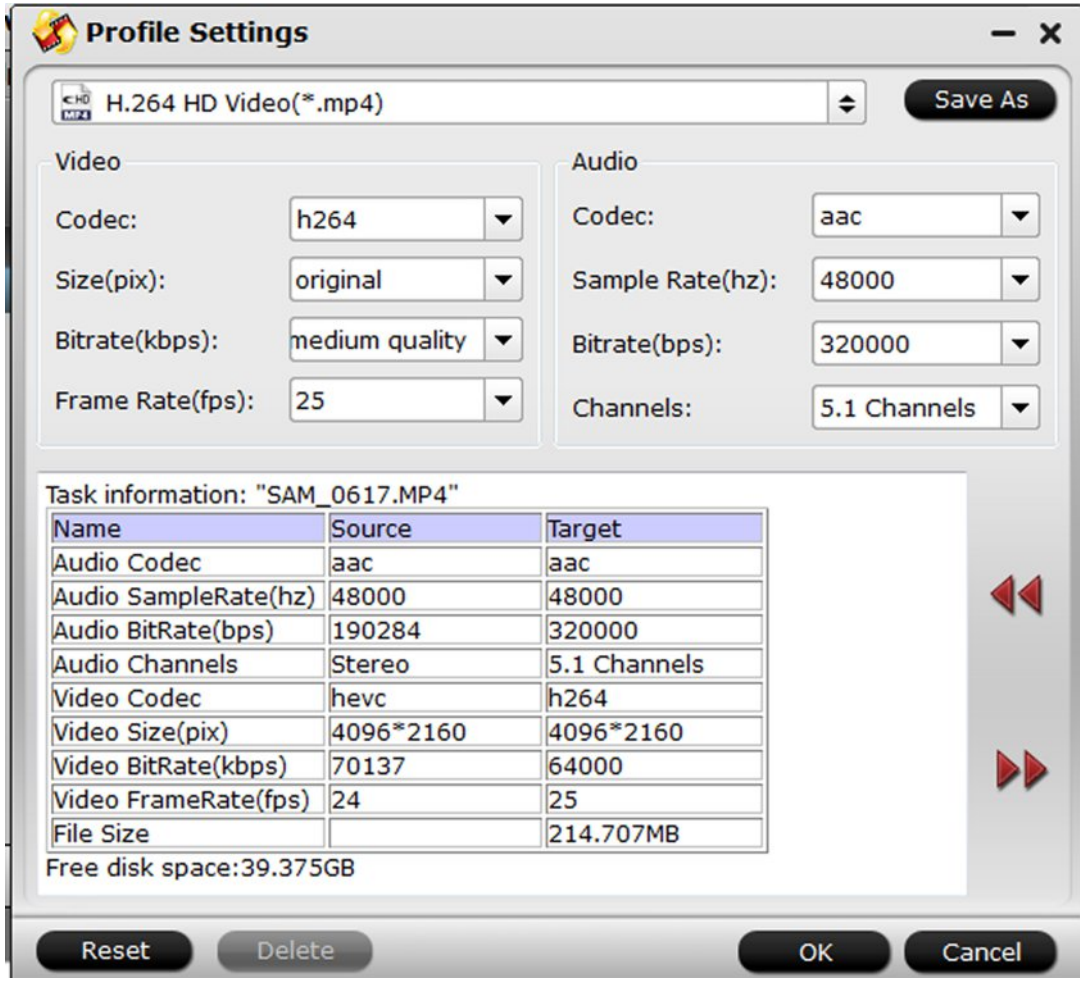
Obviously having non watermarked clips in a folder called watermarked is not ideal so we need to establish better names for the folders, Bin1 and Bin2, would be better.

There is mileage in this for 4k as the files are big and do not scrub easily in the editor so compiling low res versions, editing them and then folder swapping with a set of high resolution clips, might be the answer.

There are numerous standards behind the format settings

I recorded a 28 second 4k clip and the result was a 237MB file on the camera card. Transcoding it to an H264 MP4 in high quality resulted 357MB, about what you would expect from a less efficient codec. Medium quality resulted in 226MB and low quality resulted in 92MB which was ideal for my cutting copy and then I replaced the clips with the high quality by exchanging file names and re-rendering.

So now my old software is editing up 4k Video. The pictures look good on my 2k screen, but then, Christmas is coming and 4k screens are becoming more affordable every day. Perhaps the January sales may see another price reduction.



Si5351 Xtal Substitute Module

By Terry Mowles VK5TM

The genesis of this project was as a substitute for the increasingly harder to find 10.710MHz xtals in the Pic-a-star project. The original can be seen here:- *http://www.vk5tm.com/starlo.php*

As many would be aware, quartz crystals in custom frequencies (and even some that used to be standard frequencies) are now either getting very expensive or even unobtainable. Xtals upwards of 100MHz are also not readily available to the homebrewer.

This module, using the SI5351 can generate a frequency from as low as 8kHz, up to 160MHz. It also has 4 configurable output levels of drive - 2,4,6 or 8mA. The PIC also goes to sleep after programming the Si5351.

There are many projects on the internet using the SI5351 with two or three outputs in a VFO + BFO arrangement etc. I am not a fan of this arrangement, as the 3 output Si5351 variant (as used in this project) has a common supply rail for all output stages, resulting in a fair amount of crosstalk between outputs. This project is restricted to using just one output for this reason.

Build information

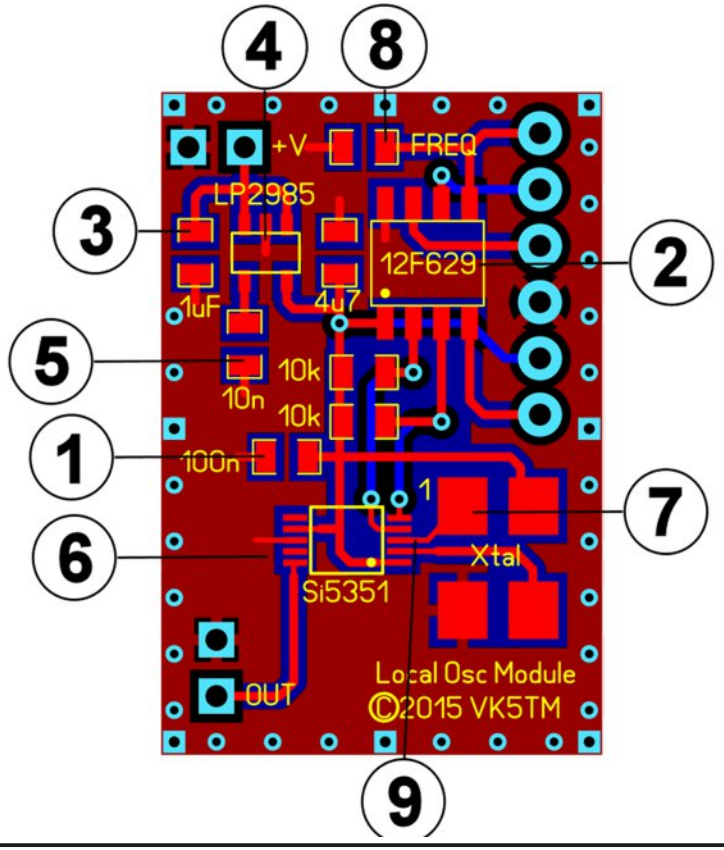
Before describing how to generate the code for the frequency of interest, I will go through the build process. This project should be an easy build for those experienced in using SMD components.

Having said that, those with average to good soldering skills and a steady hand, should also be able to do it with a little patience and perseverance.

The pcb itself is double sided, 20mm x 30mm in size and because of the narrow spacing of the Si5351 pins, is not really suited to etching at home. But if you really want to try, a pdf file of the layout is available on the *CQ-DATV site*. Otherwise, I have a quantity of pcb's available for sale.

This build description is only a suggestion of how to go about it but there are several important points to note that will be explained through the procedure.

The following numbered drawing corresponds to the numbered steps following. Please read completely through them first before commencing construction. (Please note, this is an earlier version of the pcb, but the components are still in the same relative positions. Also ignore the number 9 step, it no longer applies).



1:- FIRST IMPORTANT NOTE: The PIC is static sensitive, so safe anti-static work practices should be used.

A programmed PIC can be erased or otherwise be made non-operative from the effects of static electricity (ask me how I know this!).

Install the 100n capacitor FIRST, regardless of whether you are going to program the PIC once installed on the pcb or have already programmed the PIC.

2:- Install the PIC. If programming on board, do it now. The header pins on the right hand side are set-up to work with PicKit 1, 2 or 3 programmers (see component overlay diagram further on).

For other programmers, you will need to make up an appropriate cable.

Also, see the important smoke alert at the end of the build description.

3:- Install the 1uF and 4.7uF capacitors. These are Multi-Layer Ceramic Caps (MLCC) and are not polarity sensitive.

SECOND IMPORTANT NOTE: DO NOT use Electrolytic or Tantalum caps, they do not have the right characteristics for use in this circuit.

4:- Install the LP2985 3.3v regulator.

5:- Install the 10n cap and 2 x 10k resistors.

6:- This is the part that needs the most care in mounting - the Si5351 - the pins are small and very close together.

A good magnifier and plenty of flux are your friends. I soldered this using the "flood soldering method" - i.e. apply flux, position chip (the right way round) and solder across all the pins on both sides (don't worry about the solder bridges).

Next, with more flux and de-soldering braid, remove the excess solder.

Although it may appear that all the solder has been removed, there will be more than sufficient under the pins for a good connection to the pads.

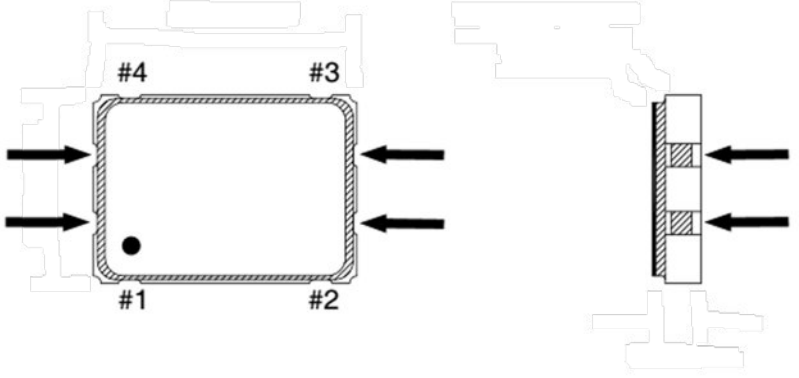
Don't be tempted to "touch up" the pins with a bit of extra solder, you are only asking for a world of trouble.

7:- Fit the oscillator module.

The BOM (bill of materials) lists a 7 x 5mm oscillator module. You may want to source a 5 x 3mm module (BOM available as part of the download on the CQ-DATV site).

THIRD IMPORTANT NOTE: (and one that caught me out) - some oscillator modules have small pads on the ends (see drawing below), which for no logical or sensible reason, are not connected to their adjacent pins.

In the oscillator module I used, that pad adjacent to pin 2, was connected to pin 1! DO NOT get solder on these pads!!!



8:- This is where you decide which frequencies to generate. If only generating a single frequency, you can skip this step. (The original software was capable of generating one of two frequencies).

An alternative programmed frequency can be generated and this is done by placing a zero ohm 0805 resistor or linking the two pads with a piece of wire at the position marked "FREQ".

You could also wire a switch across these pads if desired.

Note that this is an either/or function, you cannot switch between the two frequencies while the module is powered on.

That is basically the build information, so the next step is check and double check that everything is in the correct place, correctly oriented and soldered and that there are no solder bridges.

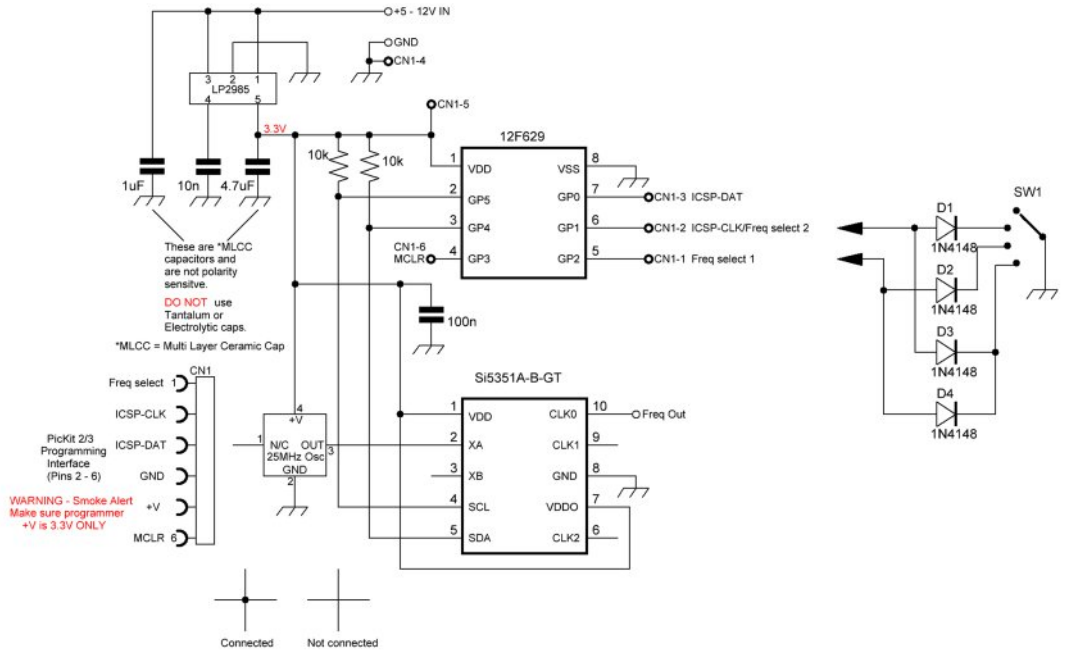
NOTE -> SMOKE ALERT: If you have a fully loaded module and are experimenting with changing the program (or forgot to program the PIC in a previous step), it is VITAL that your programmer does not output more than 3.3V on it's V+ pin.

If it does, you will let out the all important smoke from the Si5351 and the module will never work again.

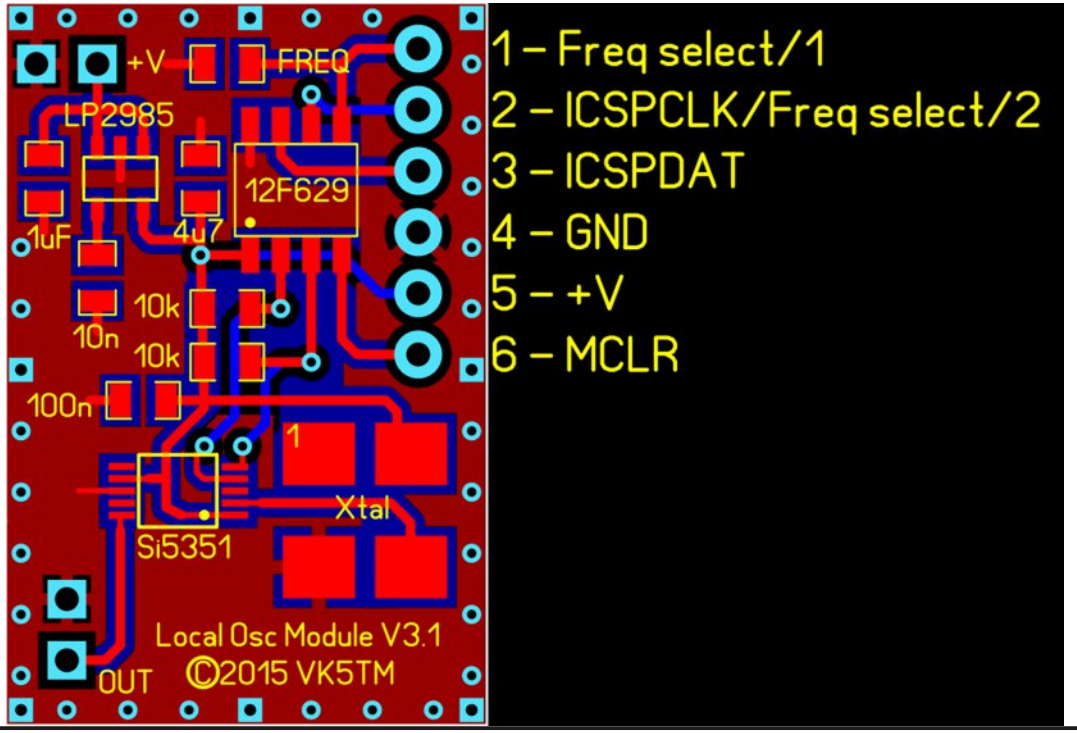
All that remains is to connect a source of power between 5V and 12V (the regulator is a low dropout type and will work satisfactorily from 5V) and check the output.

The via's around the edge of the module are all grounded, so you can use these to solder a shield to. That's all there really is to it.

Here is the schematic of the module. You will notice a diode matrix and switch to the right hand side. This enables multi frequency switching and is explained later in this article.



And component overlay and programming connector details.



A larger version of the schematic and overlay is included in the download available from the *CQ-DATV site*.

Coding the Si5351 xtal substitute frequency

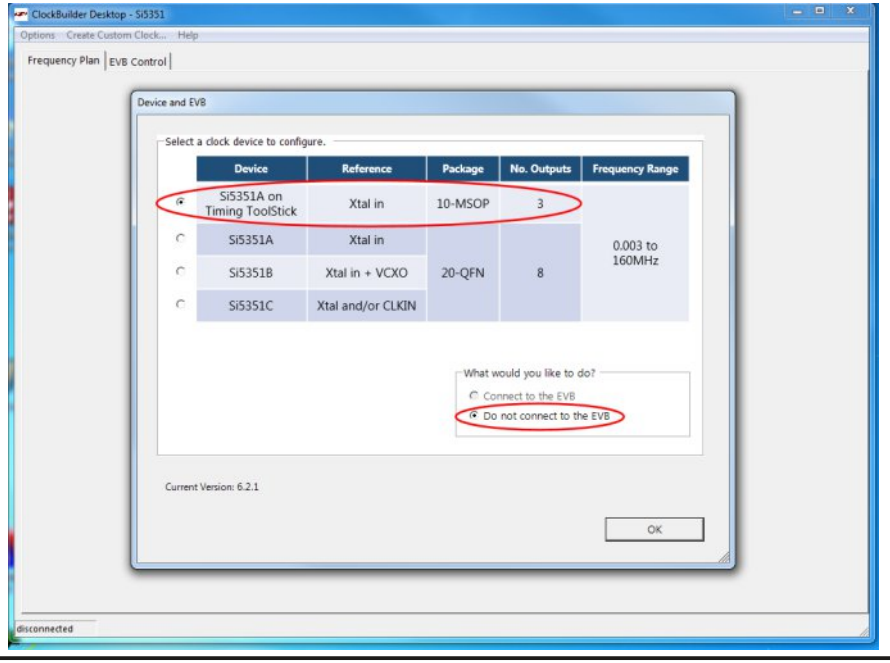
Coding the frequency of the xtal substitute module is a relatively easy task, however, you will need PIC programming skills (or somebody that can do it for you). The asm file, VK5TM_Si5351_LO.asm, is available in the download from the *CQ-DATV site*.

To start, you will need the Clockbuilder software for the Si5351. At the time of writing, it was available here:

http://www.silabs.com/products/clocksoscillators/pages/timing-software-development-tools.aspx

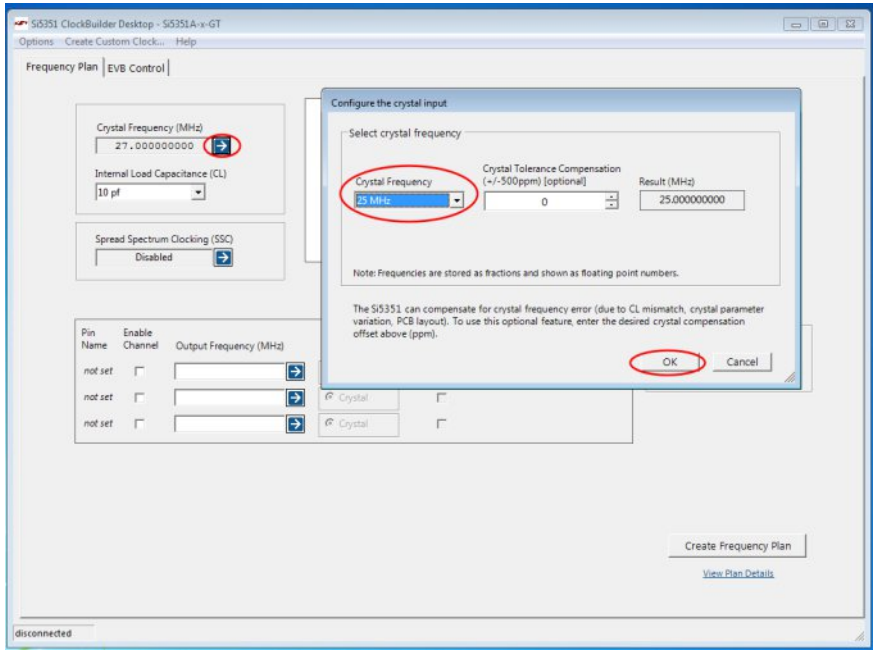
Look for"Si5351" in the list.

Once you have downloaded and installed it, start the software and you should see the following (without the red ellipses):



Select the options shown circled in red and click OK. This will take you to the main page.

Click the arrow next to "Crystal Frequency" which will pop up the box where you select "25MHz" from the drop down list. Note that there are only two options, 27 or 25MHz. Again, click OK.

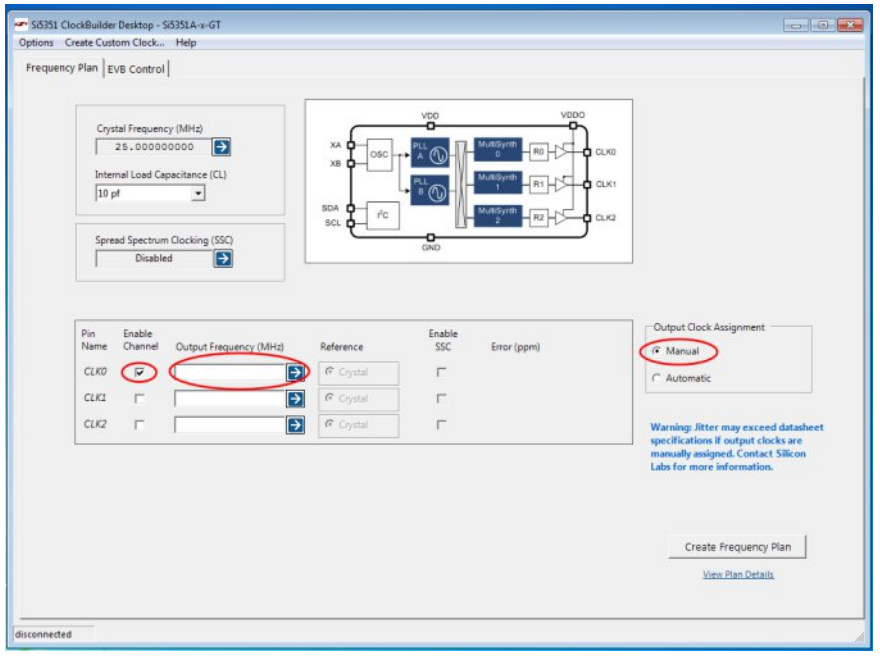


On this next page, click "Manual" first and then select the checkbox next to "CLK0". Ignore the warning that comes up, you are only using one output so there can be no jitter problems.

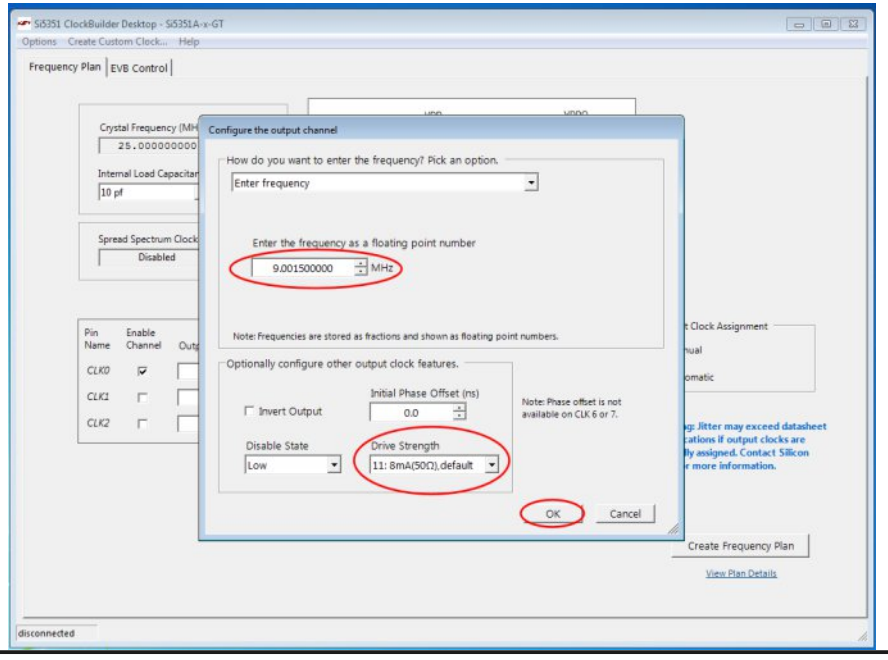
See image top of next page.

Clicking the arrow next to the blank "Output Frequency (MHz)" box will pop up a box where you can enter the frequency required and the "Drive Strength".

You can select a drive strength of 2,4,6 or 8mA.

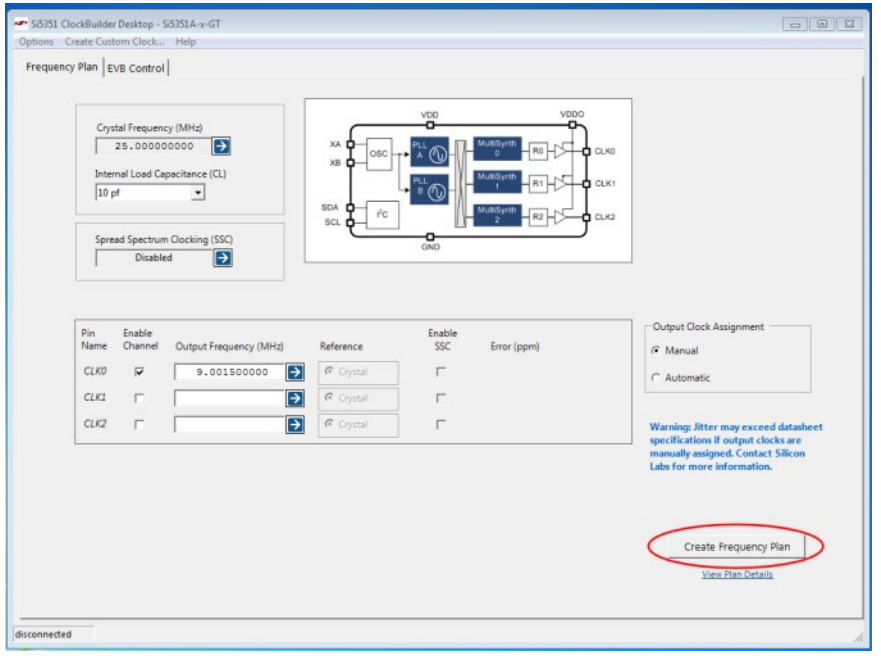


Depending on your application, you will need to experiment with this setting. The higher the current level, the higher the drive. If unsure of the level needed, select 8mA, you can always put an attenuator on the output if it is too high.



For this article, I have entered an example frequency of 9.0015MHz for use as the LSB frequency with a 9MHz filter.

Once you have made your choices, click on the "Create Frequency Plan" button. Nothing will appear to happen, but it does generate the needed file.

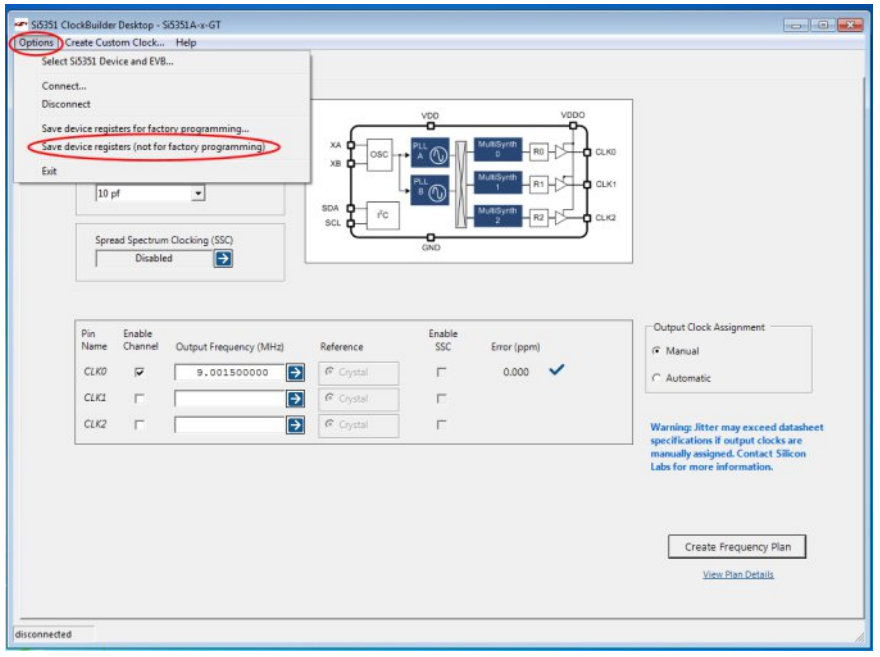


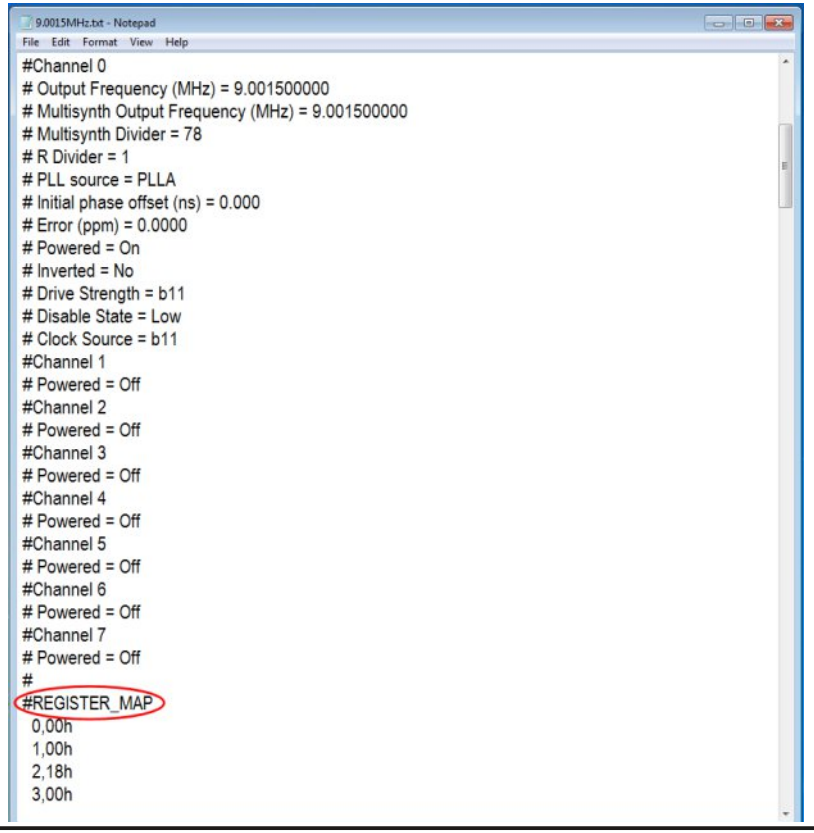
You can click the "View Plan Details link if you wish, but it will not tell you anything really useful.

Now, and this is the important bit, click on "Options" in the top menu bar and select "Save device registers (not for factory programming)"- see image top of next page.

This will bring up the normal Windows Save menu. Give the file a name and select a location to save it to.

Once you have done that, you can now open the generated text file, the picture at the bottom of the next page is a portion of that file. The important bit is from the circled "Register_MAP" tag and below.





If you only want the one frequency, then you are finished with this step, if you want a choice of frequencies, then you need to do it all again and generate another file/s, corresponding to the number of frequencies required.

Remember to save subsequent files with a different name.

What you need to do with this information is explained in the following steps.

(Larger copies of all images are available in the *download*.)

About now would probably be a good time to go and get that cup of coffee (or other beverage) you have been thinking about.

Changing the PIC program

This will require reasonable PIC programming skills. I have commented the ASM file as much as possible, but if something is unclear, contact me and I will clarify it as best I can.

Firstly, a brief explanation of the text file that is generated by Clockbuilder.

Everything with a "#" in front of it is a comment.

At the top is information on the options selected when generating the file. Then you get to the important bit, the "Register_MAP".

Parts of this are what is needed to be programmed into the PIC. The first digits on a line are the register number in decimal, followed by the command in hex.

While the list runs from 0 to 232, you do not need to add all of these.

The general sequence of programming the registers is shown in the code snippet below. First you send the address of the chip. The address of the chip is 0x60 as per the datasheet and, just to be confusing, in the program it shows as 0xC0. Just trust me that it is correct. If you really need to know why, study the I2C protocol and how the addressing works. HINT - 7bits and MSB (Most Significant Bit) first.

Following the sending of the IC address, you send the register address you want to program, 3 in the snippet below and then the command, H'FF'. At the end, you send the stop command.

This is for individual, seperated registers.

```
Sendsiregs

      call    openw           ; Open for Write, signal start, send 0xC0, ask for ACK
      movlw   d'3'            ; Address Register 3
      movwf   ebyte
      call    putbyte         ; Output byte and get ACK
      movlw   H'FF'           ; Register command - disable outputs
      movwf   ebyte
      call    putbyte         ; Output byte and get ACK
      call    stop            ; Send Stop to Si5351
```

Where the register addresses are sequential, this will get very tedious, so as per the I2C specs, you can send the chip address, the address of the first register in the list to program and then sequentially send a list of commands.

The Si5351 automatically steps to the next register in turn as you send the commands. Once you have reached the end of the list, then you send the stop command.

The following short code snippet (see image top right) is part of the sequence that programs registers 24 to 92 and the stop command is called at the end (not shown).

Now to the registers to be programmed, firstly for a single frequency. According to a post on the SiLabs forum it goes like this:

```
      call    openw           ; Open for Write, signal start, send 0xC0, ask for ACK
      movlw   d'24'           ; Address Register 24
      movwf   ebyte
      call    putbyte         ; Output byte and get ACK

      clrf    ebyte           ; Register command - CLK 0-3 disable state
      call    putbyte         ; Output byte and get ACK (24)
      clrf    ebyte           ; Register command - CLK 4-7 disable state
      call    putbyte         ; Output byte and get ACK (25)
;-------------------------------------------------------------------------------
      movlw   H'04'           ; Register command - (10.710MHZ)
      btfss   GPIO,FREQ
      movlw   H'02'           ; (10.715MHZ)
      movwf   ebyte
      call    putbyte         ; Output byte and get ACK (26)
      movlw   H'E2'           ; Register command - (10.710MHZ)
      btfss   GPIO,FREQ
      movlw   H'71'           ; (10.715MHZ)
      movwf   ebyte
      call    putbyte         ; Output byte and get ACK (27)
;-------------------------------------------------------------------------------

      clrf    ebyte
      call    putbyte         ; Output byte and get ACK (28)
```

1. Disable all outputs. reg3 = 0xFF

2. Write reg187 = 0xC0

3. Power down all output drivers

reg 16 = 0x80
reg 17 = 0x80
reg 18 = 0x80
reg 19 = 0x80
reg 20 = 0x80
reg 21 = 0x80
reg 22 = 0x80
reg 23 = 0x80

(Note that I haven't checked whether all 8 output registers are present in the 3-outout version of the Si5351, but it accepts the code, so they may be there.)

4. Set interrupt masks register (see Register 2 description in datasheet). I didn't do anything with this one.

5. Set crystal load capacitance, XTAL_CL in reg183[7:6]. See datasheet for register description.

6. Write registers 15-92 and 149-170 using the contents of the register map generated by ClockBuilder Desktop.

7. Apply PLL A and PLL B soft reset - reg177 = 0xAC

8. Enable output with OEB control in register 3. (OEB = Output Enable Bits).

So basically, you need to extract from the Clockbuilder file, all the values of registers 15 - 92 and 149 - 170 and transpose them into the appropriate places in the asm file.

A lot of the values are 0x00, so in the asm file you will see the command "clrf ebyte", which is the same as loading it with a zero value.

You may notice two or more consecutive sets of clrf ebyte, call putbyte and another clrf ebyte. This was done to make sure that the file remained at zero, as the STATUS C register is used in moving the bits out to the Si5351.

Every section in the asm file corresponding to each register is marked with the register number like so (24).

If you are doing a two frequency program, there is a preliminary step that needs to be done, note that, in the first software version, it will not switch between frequencies while powered up.

(See 'Modifying the Si5351 xtal substitute software Part 2' for multi frequency operation.)

You need to compare the two frequency plans and note which registers differ from each other. There will not be a large number of them.

The code snippet below shows the structure of how the test is done to determine which frequency to generate. It first loads the default frequency value into the 'W' register of the PIC and then tests if the "FREQ" pin is low.
If it is, it loads the new value into the 'W' register and proceeds to load that new value into the Si5351. You will find various sections delineated within the asm file where this happens with the current set of frequencies.

```
;_____________________________________________________________________
        movlw   H'0C'           ; Register command - (10.710MHZ)
        btfss   GPIO,FREQ
        movlw   H'0B'           ; (10.715MHZ)
        movwf   ebyte
        call    putbyte         ; Output byte and get ACK (29)
        movlw   H'23'           ; Register command - (10.710MHZ)
        btfss   GPIO,FREQ
        movlw   H'B7'           ; (10.715MHZ)
        movwf   ebyte
        call    putbyte         ; Output byte and get ACK (30)
;_____________________________________________________________________
```

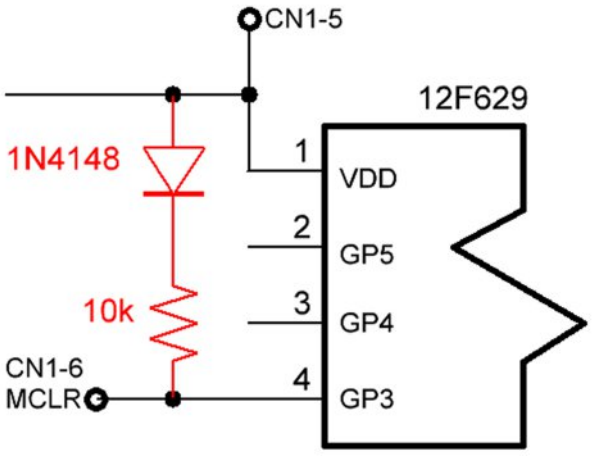
Modifying the Si5351 xtal substitute software Part 2

This is the second part of modifying the SI5351 LO module software such that it can switch between various frequencies and doesn't need to be powered down to do so.

The only hardware difference is a switch matrix using the "Frequency select" pin and one of the programming pins and modified software.

As it stands, I have only used two pins for the frequency select function out of the four available, so this will give you the choice of 4 frequencies.

The more ambitious among you may want to use the additional two available pins (ICSPDAT & MCLR) and get up to 16 frequencies. Note that internal pull-ups are used on the frequency select pins, but this function is not available on pin 4 of the 12F629 (GP3/MCLR), so you will need to add a pull-up resistor (10k) and an isolating diode to this pin.



I have not drawn a pcb for the diode switching, these can easily be mounted on the back of the frequency change switch.

Multi frequency Software

An asm file, VK5TM_Si5351_Multi_LO.asm, written at the request of another ham, can be used as an example of how to accomplish multiple, live frequency switching. It is in the download available from the *CQ-DATV site*.

It is much the same as the original software as far as programming the Si5351 is concerned, but has had major changes to accommodate this modification. It now uses interrupts. The PIC still goes to sleep when not actively reprogramming the Si5351.

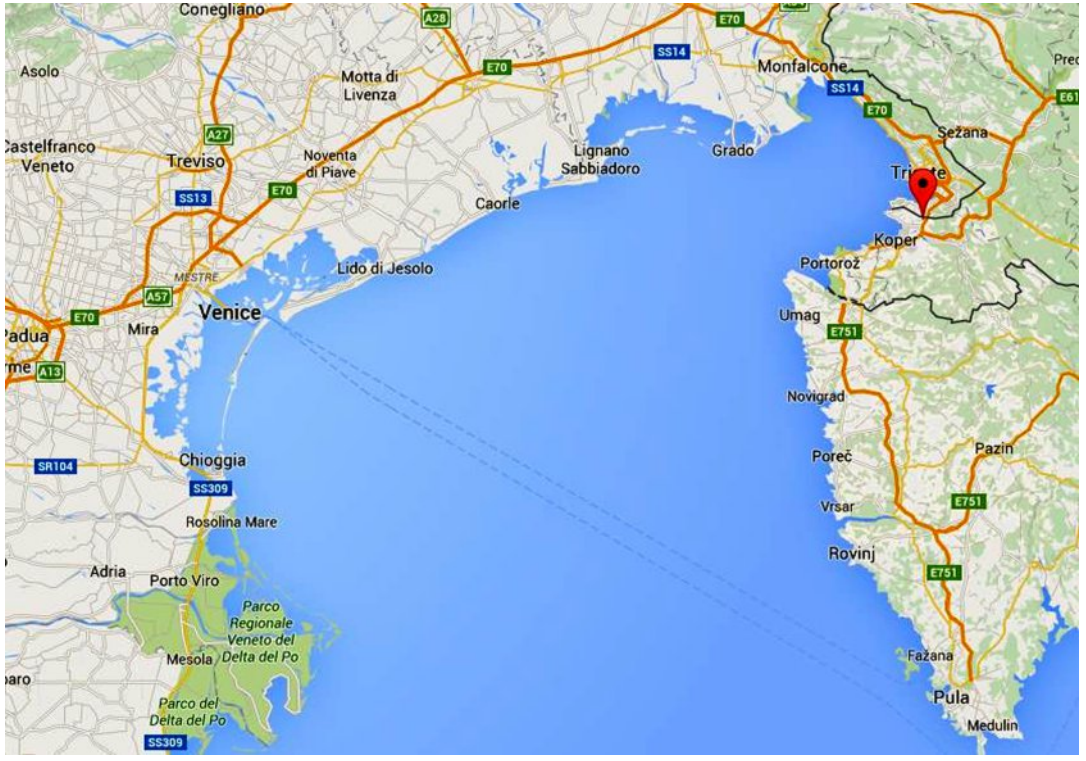
In this example, the four frequencies generated are: 54MHz, 53.9985MHz, 53.0015MHz and 53.9993MHz.

Hopefully, it is commented enough for you to work out what is going on, contact me if you need clarification on anything regarding this software.

TV Amateur is a German Language ATV Magazine It is published 4 times a year and if you would like to subscribe go to http://www.agaf.de/

SSTVJ ATV repeater

by Rudi S58RU and Mauro IV3WSJ

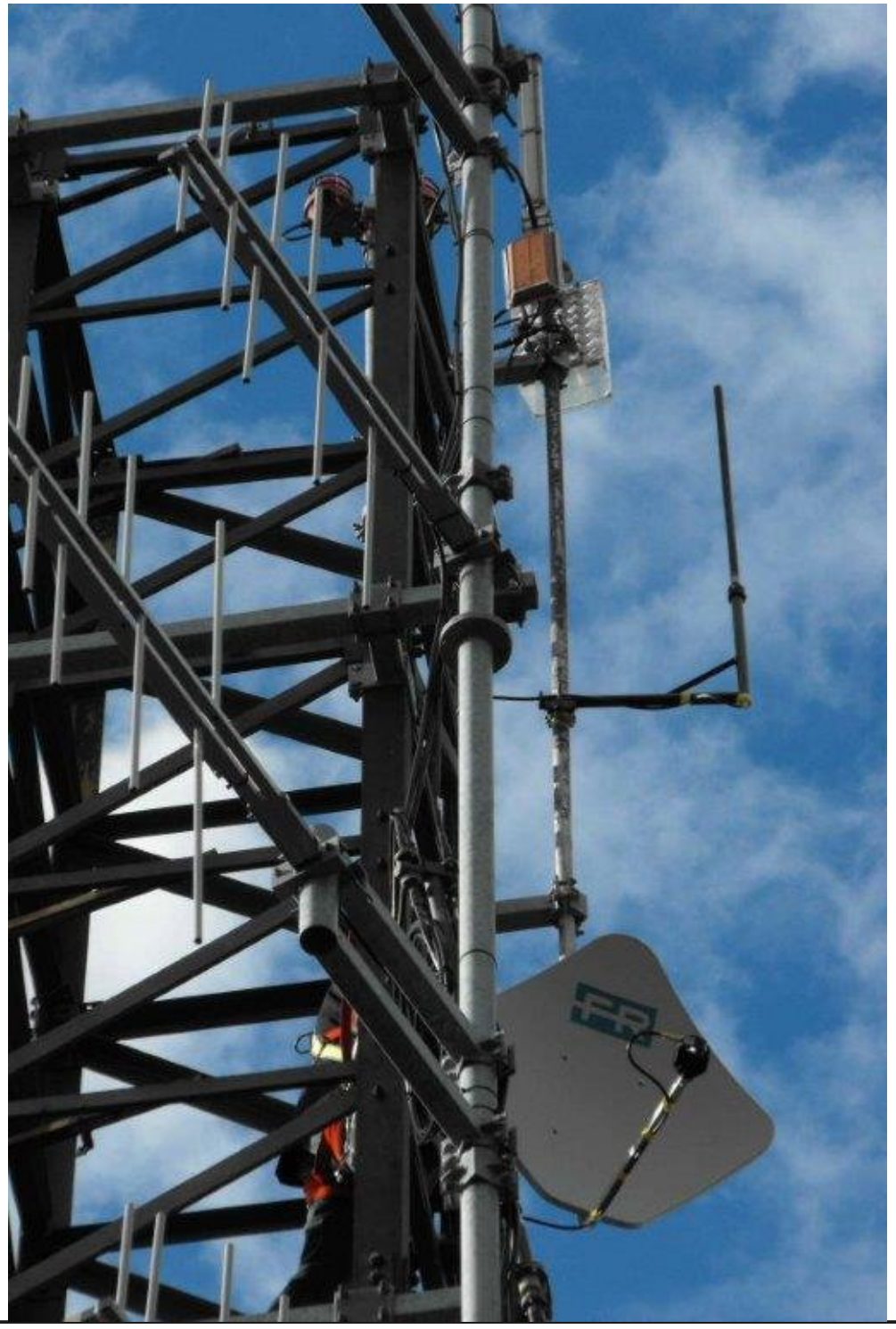


This is an ATV repeater, located in Elerji in Slovenia, and started four years ago as an Analogue ATV repeater.

Two years ago, Rudi S58RU and Mauro IV3WSJ, decided to add a DATV repeater using DVB-T, and It has now been active for several days, with the call sign of S55TVJ ATV.

This repeater has an input on 1292MHz DVB-T and an output of frequency DVB-T 10.400MHz. The output power is 50mW at 10.400MHz and the antenna is a slotted 360° with 9 slots, to have a lower angle.

The heart of the whole system is the DVB-T repeater of Hides, the BR-100. We had to add receive filter to narrow the frequency 1292MHz, a homemade down converter for 23cm.



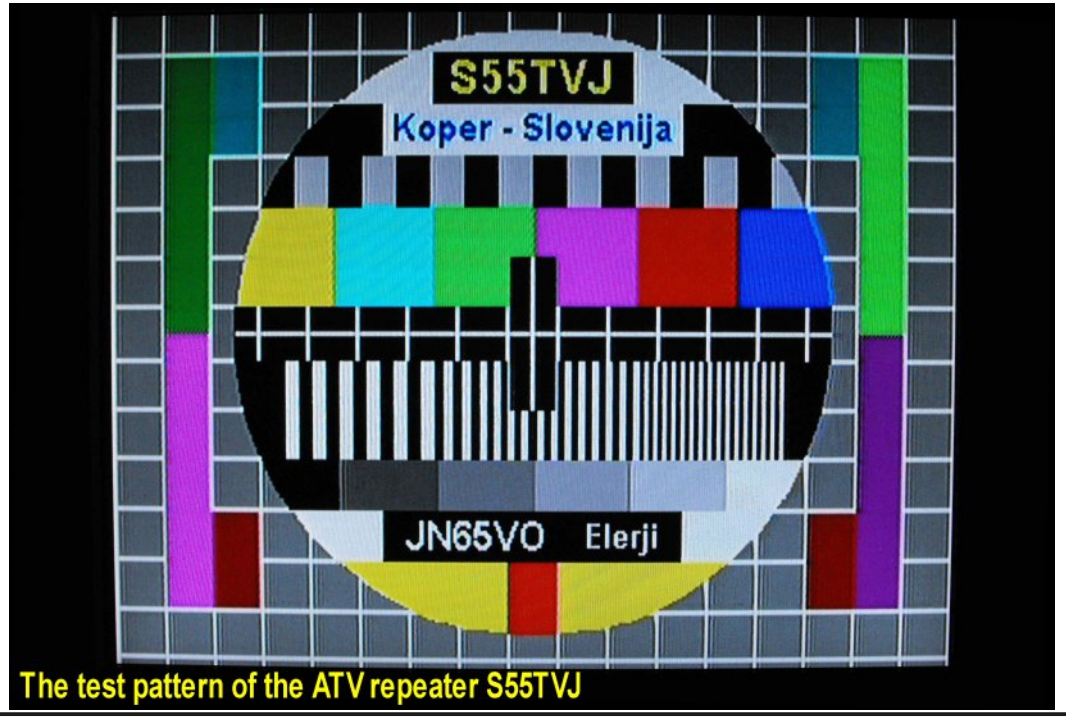
For transmission, we used a local oscillator (10GHz) controlled by a PLL, with a low phase noise, a mixer, two stages of amplification in class A home-made, and finally, a filter calibrated to 10.400MHz.

Rudi bought the BR-100 two years ago, and currently it is out of production.

The BR-100, can be set in reception and in transmission with a range of bandwidth from 1 to 8MHz, then, if it becomes necessary to change the bandwidth, because of the European legislation then this will not be too much of a problem.


Mauro IV3WSJ with the
ATV repeater S55TVJ
12 5 2012


Rudi S58RU
Mauro IV3WSJ
Elizej S57AZW
Preparation for mounting of the antennas ATV


S55TVJ
Koper - Slovenija
JN65VO Elerji
The test pattern of the ATV repeater S55TVJ


S55TVJ JN65VO Slovenija
8 11 2012
Mauro IV3WSJ, camera maintenance

Simple ATV Callsign & Bitmap Graphic Generator

Mike G7GTN

Project description

I wanted the simplest possible video caption generator that could display text but also simple bitmap graphics. This design is the result of this requirement.

Circuit Diagram

The circuit shown in figure 1 is used with an Ardunio UNO which is clocked at 16MHz and handles the video generation within software.

The only required components are two resistors. The push button layout is shown in figure 2

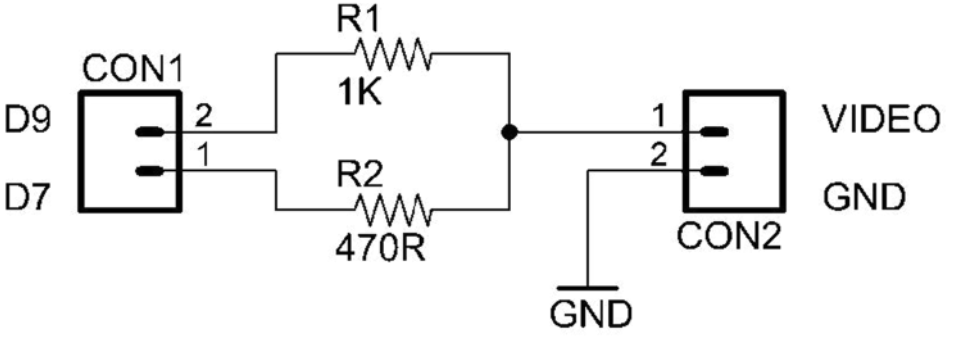


FIGURE 1 - VIDEO OUTPUT CIRCUIT

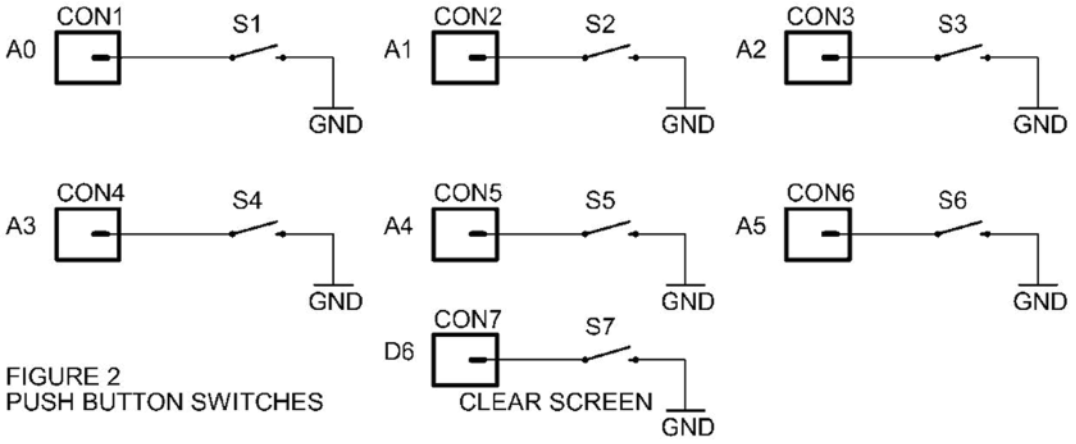


FIGURE 2
PUSH BUTTON SWITCHES

A0-A5, Are Ardunio Analog I/O Pin References. D6 is a Digital I/O Pin Reference

Construction

I used a small prototype pad style board to construct the two resistor output circuit, but these could easily be wired across the video output socket. The same method was used to mount the small tactile switches I wished to use for the front panel. The case was recycled and hence the fetching pink colour. I suspect a quick respray will be required to make the project look more suitable.

Download Firmware

The download for this project CAPGEN.ZIP also has the required Ardunio TVOUT Library. You need to correctly install this in to your additional libraries folder to compile and make use of the supplied code. Of course you will also need to change the text for your own required captions or messages. To check that you have the layout correct for all captions you can uncomment the #define TESTLAYOUT on line 30 to quickly go through each caption with a ten second delay between each. When you are happy with the results comment this line back out and re-upload the code. Now you select your captions via the push buttons.

Bitmap Graphics

Very simple bitmap graphics are also quite possible; these are all 1 bit BMP files and converted to array data to be loaded in as a complete image. You do very much have to bear in mind the very low starting resolution of the actual display. The trade-off will be in getting a reasonable quality for a recognisable image along with keeping the size of the font large enough to easily read for your actual captions text.

Operation

As configured we have six buttons to select either a different caption or compiled bitmap graphic, with theseventh button acting as a clear caption or graphic button. The code is simple enough for you to configure for your own specific preferred operating requirements.

Conclusion

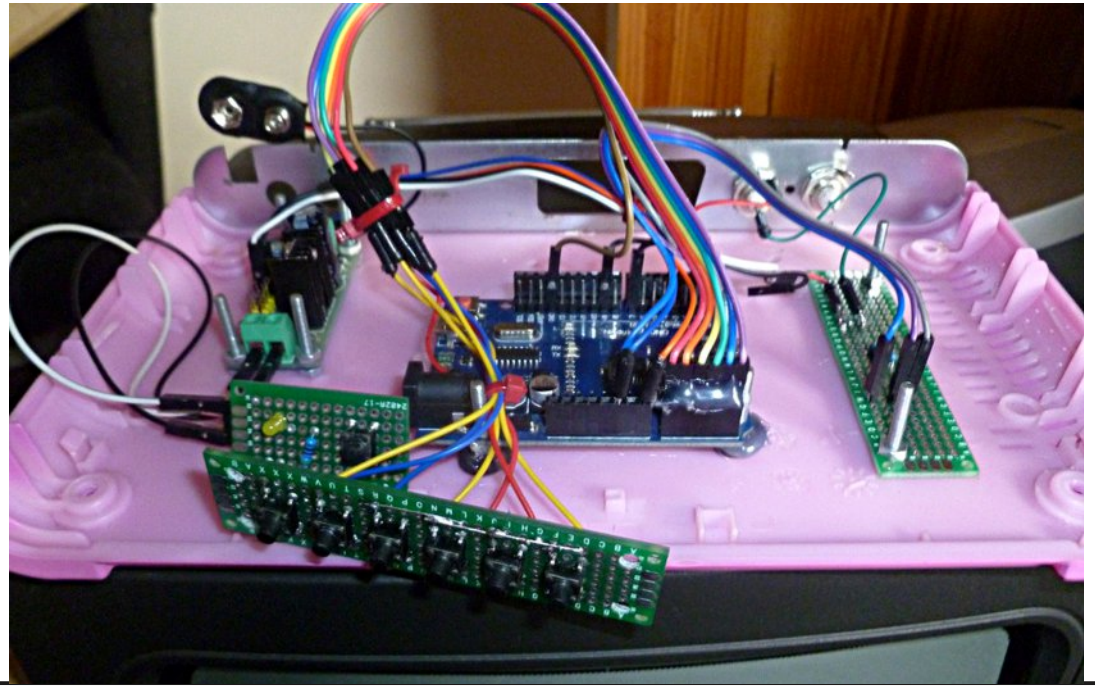
In essence the final video output from this circuit is somewhat like the video output from the ZX-81 microcomputer. A low pixel resolution using no expensive parts, but still suitable for a simple fixed caption or graphical bitmap. You should be able to easily build this project for a total of around £12 and a couple of hours of your free constructional time.

References

The firmware and required library files are available from the usual *CQ-DATV download page*

The TVOUT library general functional details
http://code.google.com/p/arduino-tvout/

Prototype in the process of being boxed (Case respray certainly required...)



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 too 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.

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 - *http://cq-datv.mobi*

Author Guidelines

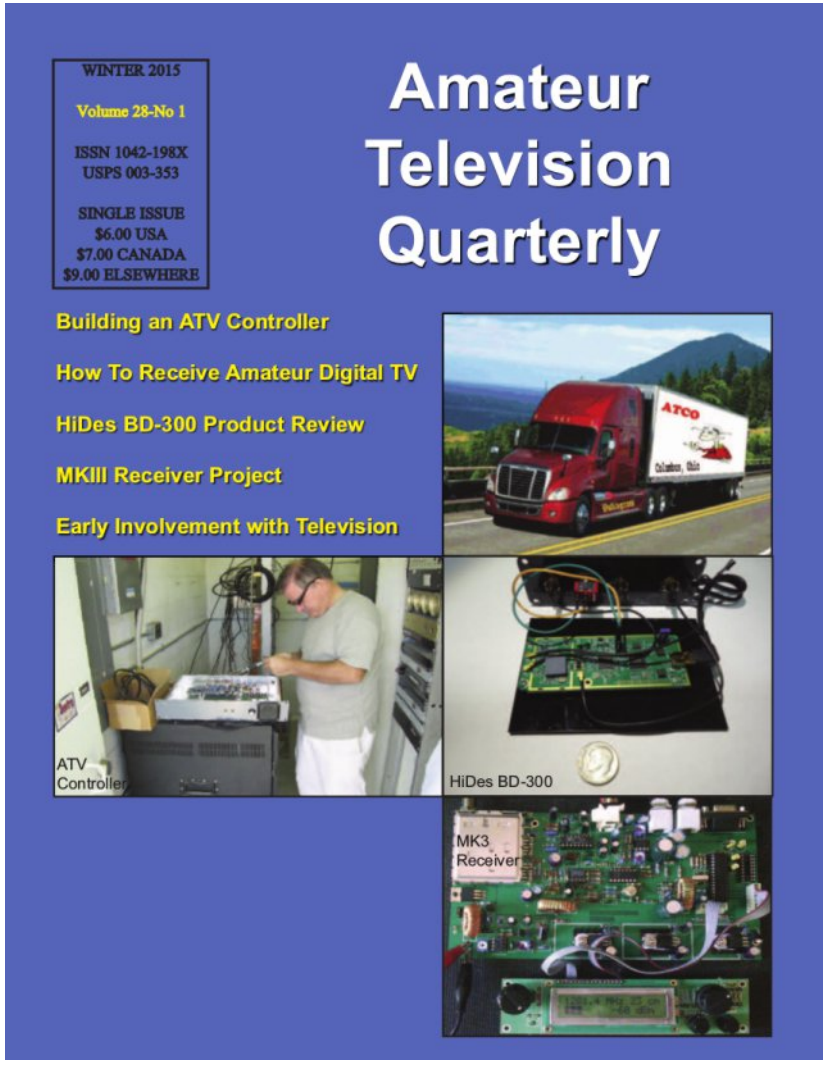
CQ-DATV welcomes contributions from our readers. It does not necessarily have to be on ATV, as long as it is of interest to our readers.

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.



ATV Quarterly - Don't miss another issue! Subscribe Today

USA $22.00 year, Canada/Mexico $25.00 year DX $32.00 year (US $) Cyber: $15/yr. Visa, M/C, AMEX, PayPal via Internet: www.atvquarterly.com
Cheques or Money Orders to P.O.Box 1594 Crestline CA 92325
Published by ATV Quarterly tel (909) 338-6887 email: wa6svt@atvquarterly.com

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.