An advanced digital Spectrum Analyser

Several problems exist with the current simple analyser. Firstly, in order to obtain a sharp display, the sweep rate must be low (10-25 sweeps per second). This is primarily a limitation of the low spec 5MHz bandwidth oscilloscope in use. But low sweep rates would also be needed if lower bandwidth filters were used in the analyser. The low sweep rate causes the display to flicker unattractively.

Secondly, the display is less bright on tall peaks because the oscilloscope beam spends the same time covering greater distance than on smaller peaks or the baseline. This makes the display difficult to read unless ambient lighting levels are somewhat dim. A related issue is the difficulty of photographing analyser output for the purposes of web documentation.

Both these problems can be solved by digitising the analyser and building a dedicated display system instead of using an oscilloscope as the display. Including the above problems solved, this scheme would also have several other advantages:

- Variable sweep rate without flicker
- Bright and uniform display
- Multiple traces stored in memory
- Dual channel capability
- Download to computer via parallel port (good for web page snaps)
- Frequency and amplitude readout
- The oscilloscope is freed up for its normal usage
- It's a cool project

What, no microprocessor?

At this point most people start to consider a microprocessor as the ideal way to implement all these functions. Such it may be, for a commercial instrument where profits are going to be maximised by minimising parts count. A micro may also provide numerous nice custom features and a convenient user interface, with on-screen measurements etc.

Yet I am often (and probably unreasonably) biased against using microprocessors. There are many reasons for this. I am attracted by a purely logic-based design. No host computer is required for programming the micro, no host software, no downloads, no programming, no cost, no hassle. There is a certain associated minimalism to the all-logic approach despite the distinct non-minimalism of the high circuit complexity compared to the micro route. Or perhaps it's just that I work with a PC all day long in my day job, and don't have any wish to go near them in my hobby.

Whatever the reason, I have more fun with logic so humour me.

The new display

More truthfully an old display: the 4.5-inch (diagonal) television tube from an old bedside TV/Radio/Alarm clock. This unit has an interesting history. It was purchased from a car boot sale a year or two ago for £5. The label said the TV sound, radio and clock worked Ok, but the picture needed attention. The stall owner said the picture was just a bright horizontal line in the middle of the screen. I guessed that was most likely a problem with the vertical deflection coil driver and would be fun to fix even if I didn't particularly want a TV/Radio/Alarm Clock…

Sure enough on inspection with the oscilloscope at home, one set of coils showed a large waveform and the other almost nothing at all. I followed the wires back to the PCB and the PCB traces back to the driving transistor and it was clear the amplitude of the signal on its collector was a lot smaller than the base signal.

Next began a game of trial and error with various different transistor replacements from the junk box. The first one tried caused the picture to expand to about 1/3 the screen height, a nice confirmation of the diagnosis and solution. The next transistor tried filled the whole screen perfectly. But after 20 seconds a small puff of smoke and the display returned to a thin horizontal line. Ok, so driving those coils must be a difficult job and need a transistor capable of higher power dissipation. So I tried a BD139 and was rewarded with a permanent full screen.

Interestingly even the BD139 gets uncomfortably hot to touch. The original transistor (I forget the type) must have been slightly under-specified for the job required of it, causing reduced lifetime. Bad engineering.

At any rate this TV tube will make an ideal display for a spectrum analyser, compact and bright. Driver electronics for this TV tube will be built using the components salvaged from the original. The screen picture above shows BBC television introducing Marco Fu in the World Snooker Championships 2003 at the Crucible theatre in Sheffield, UK (not easy to photograph).

Digital spectrum analyser scan

The UK TV standard is 625 lines per frame, with 50 frames per second. In actual fact only half the lines are displayed in each frame, odd numbered lines one frame, even numbered the next, then odd etc. This technique is called interlacing and the idea was to generate a low flicker display with 50 frames a second but consuming half the UHF bandwidth that if every frame contained the full 625 lines.

For my purposes it's ignorable, from now on I'll consider the display to consist of 312 lines per frame and 50 frames per second.

Calculation shows that each line of the display has a duration of 64uS. A certain proportion of this time is used for retracing the electron beam back to the left side of the tube. The edges of the display aren't useable due to pin-cushion effects and other non-linearity (these effects aren't noticeable on an ordinary TV picture). Experience shows that 48uS for of the horizontal scan time is a good useable display.

Similarly in the vertical direction 256 of the 312 lines used, in conjunction with 48uS horizontal lines, will produce a nice rectangular display area. For this project each line will be divided into 512 pixels. The display area will therefore have a resolution of 512 x 256 pixels, nice easy to work with dimensions.

One analyser sweep will consist of amplitude measurements at 512 discrete frequencies. These values will be stored in memory and read out during the display active period. The VCO will be driven by an 9-bit digital to analogue converter. The digital input will be from a 9-bit counter which will be incremented once per horizontal TV display line. An analogue to digital converter will take its input from the logarithmic amp output of the spectrum analyser. This will be written to the appropriate memory location on the next counter increment 64uS later.

Half the frequency sweep will occur per frame (256 lines) therefore the sweep rate will be 25 per second. Arrangements will be made for slower sweep rates by making one frequency measurement per 2 lines, 4 lines etc. resulting in ½ and ¼ sweep rates. More later on what will be done in the 56 non-displayed lines.

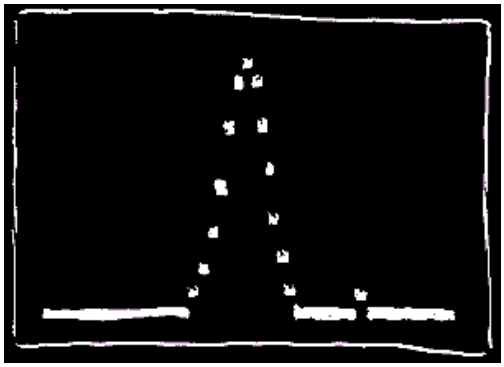
Display cosmetics: border and grid

The screen will use white lines on the black background. The display rectangle will be framed by a solid white line – this will be accomplished by detecting pixel rows –1 and 256, and columns –1 and 512 using comparator

logic. Similarly a grid will be created by colouring every 32nd row and column grey, thereby dividing the display into an 8 x 8 grid; this will be switchable from the front panel.

I may add some logic to displace the display very gradually and imperceptibly to prevent screen burn.

Generating the display from memory

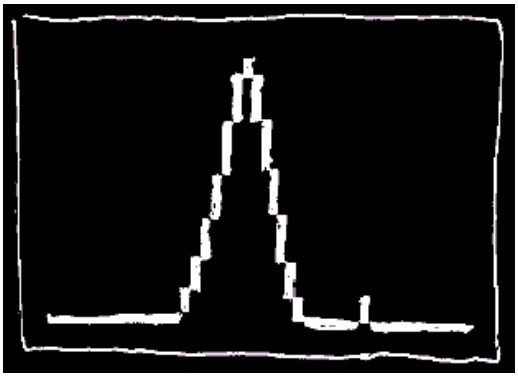


Each frequency sweep will result in 512 bytes of memory used which must be read out to the screen to generate the display. To do this every column pixel will correspond to one memory address. Each pixel is displayed on the TV for 93.75nS. To determine whether or not a display pixel should be coloured, the read byte (dB measurement value) will be compared with the row pixel count, and the pixel lit only if it matches. Therefore the whole frequency sweep results are read out every display line (512 values in 48uS), but only pixels where the value matches the vertical counter are displayed.

Unfortunately this scheme creates a problem as illustrated. If implemented as above the result is rather like a join-the-dots picture in a child's drawing book where the picture outline only appears when the dots are joined by lines.

A method for interpolation between adjacent frequency steps

The join-the-dots problem can be solved by modifying the comparator function described above. Where I just drew a dot where the vertical row count matched the dB measurement. Now I will use two values, the current dB value and that of the previous frequency step. All pixels where the vertical row count is between the current and previous dB values will be lit.



The implementation of this idea is not difficult. Data read out of memory will enter a 2-stage "pipeline" consisting of 8-bit flip-flops. The 2 stages outputs are the current and previous dB data values. Two comparators compare the output of each stage with the row count producing <, > and = signals. A few logic gates can then process these outputs to generate the required light-the-pixel signal.

The fact that display of the pixel is delayed by one "pixel clock" isn't an issue and merely shifts the display rectangle imperceptibly to the right. Pixel 0 has no previous value to be compared against so should probably not be displayed, this column can become the border (column 0 rather than –1 as stated previously).

This method does not produce true straight line joining of the dots, in which every line would not be vertical but would include a 1 pixel horizontal component. Half the vertical line should occur on the left column pixel, half on the right. However in practice one 512th of a 4.5-inch TV screen is going to make this slight inaccuracy completely invisible to the beholder.

Dual channel display

It seems to me useful to be able to simultaneously display two separate channels on the analyser screen. This could come in handy when building many circuits, e.g. with an RF amplifier one might want to see both the dB gain and whether the amplifier created any nasty spurious responses – one channel as input and one on the output would show this nicely.

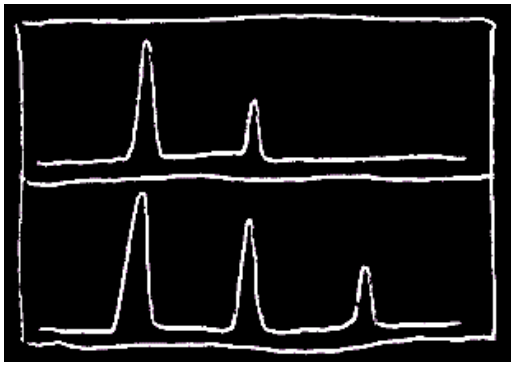
Duplicate input circuits are required, along with some means of electronically switching the rest of the analyser between the two channels, this switch will need a high degree of isolation between channels as well as high bandwidth and low distortion of signals.

Following that it is simple to arrange for frequency sweeps to alternate between each input channel, writing into successive 512-byte blocks of memory. Both channels will share the same frequency sweep range. The sweep rate is thereby halved from 25 sweeps per second to 12.5.

There are two ways the dual channels could be displayed on the screen. My plan is to implement both, switchable from the front panel.

First, the two displays can just be superimposed on the same display. The comparator and interpolation circuits described above are duplicated for the second channel, and the results OR-ed together. This presents some slight complication in so far as each column pixel must read a byte from both the 512-byte blocks of sweep memory. In practice since each pixel lasts 93.75nS and SRAM memory chips with 15nS access times are readily available, there will be plenty of time to read both bytes.

Although the television tube is black and white only, it is possible to "colour" the two channels differently for easy identification using grey levels, e.g. one channel can be white and one grey.



The alternative method to display both channels is to divide the screen rectangle into two separate regions along the 128th pixel row. This halves the vertical height of each sweep display. The border comparator can easily be extended to include the 128th row to nicely delineate each region. Some additional logic will be required to accommodate the half height display and further complicate the logical monstrosity of the comparator section.

Stored frequency sweeps

Two channels take 1Kbyte of memory, a very small amount when SRAM chips of 128K capacity are commonplace. The smallest SRAM chip available today is 2K, the next size up is 8K. Rather than only use the first 1K of the memory chip, it can be used to provide a storage facility.

To accommodate this each of the two channels will be selectable to one of the 16 available blocks of an 8K memory chip. Selection from the front panel will be via an up/down switch which will increment or decrement the block selection. The block selection will for each channel will be displayed on a single 7-segment display as blocks 0-9 and A-F. Each channel will be switched into either a live "Record" mode or a "Playback" mode in which previous data is read from the memory.

This system provides interesting extra features, interesting traces can be read stored during a circuit development process for comparison with results when the circuit is modified slightly.

Computer download

An asynchronous computer interface can be introduced very easily by outputting one parallel byte dB measurement on each of the 256 displayed rows per frame. The 56 non-displayed rows per frame are used as placeholders for the host computer to know the start of each frame. 4 frames are required to output the complete 1K of the dual channel display. For simplicity the complete set of 16 stored sweeps won't be output, only the currently selected pair. The resulting data transfer rate of 12.5Kbytes per second is easily within the capabilities of a PC parallel port. A simple software program will scan the parallel port and be able to detect the start of each frame and read the 1024 dB-value bytes. The program can produce a graphic output to reflect the screen display. Labelling facilities can be included as well as the ability to save the images as .gif files etc.: useful for including spectrum analyser outputs on project web pages avoiding the need to photograph the actual display.

Frequency readout

It is planned to add a frequency counter display to the analyser. The user will specify the position of the reading within the sweep range by setting a front panel potentiometer which is connected to a 9-bit analogue-to-digital converter. This 9-bit digital value will be compared to the column pixel count using yet another

comparator, so that a vertical line can be drawn on the display to indicate the position of the frequency reading. In this way the user can accurately measure the frequency of the left or right edges of the sweep range, as well as any of the 512 points in between.

The frequency readout will be a 6-digit 7-segment LED display. The count period will be 1mS giving the counter a resolution of 1KHz. This count will be performed by setting the 9-bit frequency measure setting to the VCO control input during the 56 non-displayed lines of each frame.

The 1mS gate time will be generated from the display timebase which will be a divided down quartz crystal oscillator. The frequency counter is simply a set of 6 binary coded decimal (BCD) counters driving 7-segment display decoders.

Amplitude readout

Similarly an amplitude measurement will be performed and displayed on a 3-digit 7-segment display with a resolution of 0.1dB. The vertical position of the measurement will be set using a front panel potentiometer via an 8-bit analogue to digital converter. The channel being measured will be selectable (only one channel measureable at a time – duplicating the amplitude measurements for two simultaneous channels is where I draw the unnecessary complication line).

It is interesting to consider two possibilities now – either the vertical position of the measurement can be determined as above by the front panel potentiometer, or it could be locked to the amplitude measurement of the currently selected frequency measurement position. Both methods will be supported, selected by a front panel switch.

A comparator in the infamous comparator logic section described above can cause a horizontal line to be drawn across the screen at the position of the amplitude measurement. In this way a “cross hairs” is drawn on the screen with the intersection showing the position of the accurate frequency and amplitude measurements displayed. The position of the cross hairs is determined by the two front panel potentiometers. In the case of measurement of the value at the frequency setting, only the “horizontal” (frequency) setting has any effect since the vertical setting is determined by the amplitude of that frequency step.

Actual measurement and display of the dB value is a little more complex than the frequency counter. It takes the form of an analogue to digital converter, but cannot be implemented as a single A-D chip since a 3-digit BCD output is required to control the 7-segment display drivers. Again it will take place during the 56 non-displayed rows.

My current plan for performing the BCD analogue to digital conversion is to use a standard analogue to digital converter with at least 10 bits of resolution (for 0.1dB displayed resolution). The input of this analogue to digital converter will be the analyser logarithmic amplifier output, after suitable scaling for accurate dB measurement.

Conversion of the binary output of the ADC to BCD will be via two sets of counters. The first will be a 3-digit BCD counter, which drives directly the 7-segment decoders. The other will be a 10-bit binary counter. Both are clocked simultaneously from zero, until the binary counter output matches the ADC output (use another comparator!).

Accessories

No self respecting spectrum analyser should be without the following accessories, which will be built into the analyser:

- Tracking Generator
- 10MHz and 1MHz marker generators
- Simple signal generator for testing