Pic 'n' Mix Digital Injection System

Part One, by Peter Rhodes, BSc, G3XJP*

THIS CONSTRUCTION project brings together a number of themes which I have been kicking around for some time. But first, why "PIC 'N' MIX"?

TWO ESSENTIAL TERMS

PIC - A range of microcontrollers produced by Arizona Microchip Inc. In this application, the PIC 16C84.

DDS - Direct Digital Synthesis. The technique of digitally generating the output frequency directly (as opposed to typically mixing the output of a VFO with a crystal oscillator - or employing phase-locked loop techniques). In this application the Analog Devices AD9850 'complete DDS synthesiser' chip is used.

IN BRIEF

PIC 'N' MIX provides PIC-controlled direct generation of the required injection frequencies into the signal frequency mixer in your transceiver.

PIC 'N' MIX also in the sense that you can pick and choose which functional elements you build; and in the sense that there are, by design, a number of different mechanical configurations to best suit your circumstances.

You are also presented with the radical choice of using the software I have designed - or writing your own.

The PIC microcontroller (and about 400 hours of software development) provides control and operational flexibility, while the DDS chip is used to synthesise the RF output, giving stability and low phase noise.

CONVERGING THEMES

DISCOUNTING THE VALUE of your time, I would argue that for years it has been viable to build multi-band HF transceivers which outperform their commercial counterparts at any point on the Price v Performance graph - from the cheap and cheerful through to the truly exotic. Except, that is, for one critical element - the injection oscillator.

I have been building VFOs for years that for all practical purposes didn't drift. Almost all were based on the Vackar, running somewhere between 5-10MHz. Besides some time consuming temperature compensation, I never gave them a second thought.

But they need about eight crystals, a mixer and switched bandpass filters before they can feed both the signal frequency mixer - and a frequency counter which gives a natural display of exactly not quite the frequency you are on! It can all be made to work, but only at substantial cost in time, money and space. And the only incremental feature easily obtained is RIT.

Then, in February 1996, *Technical Topics* [1] reported the results of some phase noise measurements made by Colin Horrabin, G3SBI, and Jack Hardcastle, G3JIR, on a stable Vackar as 'rather disappointing'. This set me thinking.

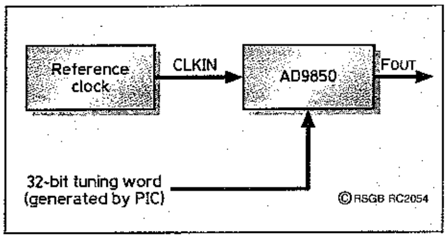

Fig 1: Basic DDS block diagram.

Most of us ignore oscillator phase noise because we can't measure it, myself included. Does it really matter in practice?

The *ARRL Handbook* has an excellent section on the subject [2] which concludes "...far-out phase noise can significantly reduce the dynamic range of a receiver. Far-out phase noise performance has effects just as critical as blocking dynamic range and two-tone dynamic range performance of receivers." Yes, but does it really matter in practice? I mean, am I truly going to fail to copy real signals on a significant number of occasions because of poor phase noise performance?

I determined to find out by adopting the simple expedient of fitting a change-over switch between my traditional VFO and a phase-quiet alternative of the same power output. Then, under a variety of practical conditions, could I tell the difference? The problem, of course, was to find this alternative without spending impracticable sums of money.

Technical Topics [3] came to the rescue again by first bringing to my notice the Analog Devices AD9850 DDS chip. A few minutes on the Internet produced the data sheet - and it all looked too good to be true. I set about designing some traditional TTL to control it, and actually got as far as building some of the boards before giving up. This is because, although I have no doubt it would have worked, 28 TTL chips to control one DDS chip - and provide a modest range of useful features - was ignoring any reasonable definition of the 'in practice' imperative.

It was obvious from the outset that some form of microcontroller would provide the solution to the control problem, and at the same time offer the ability to provide a range of operational features. What put me off for months was the costs of acquiring the development environment and the hardware to program the chip. A glance in the larger catalogues suggested little change from a £200 investment for PIC development - totally unacceptable.

The bottom line is this. Arizona Microchip provide on their website their complete development environment at no cost - as well as copious application material. And there are numerous circuits for PIC Programmers published on the Internet which you can build for less than £5. The project was born.

CONCLUSION

PHASE NOISE *does matter* in practice. On a substantial number of occasions it makes the difference between R2 and R5 signals on SSB.

For example, the home-brew net convenes daily around lunch time on 80m, just down from the SSTV calling frequency and just up from a prominent French coastal station. These are a convenient source of large adjacent channel signals.

If the band is flat and quiet, it makes no difference. If conditions are lively - using the DDS source - then I can often copy Ed, EI9GQ, at only just R5. Switch over to the VFO and the readability instantly degrades to near hopeless if - and only if - there is significant adjacent channel activity. The effect is insidious. It's not that Ed's signal goes down, it's that the base level of band background noise appears to go up. It doesn't, of course.

What is happening is that the noise sidebands on my VFO are mixing with adjacent signals to produce incremental noise in the passband. A very salutary experience, because this noise is totally indistinguishable from band noise and you could operate for years without realising what was happening.

It would seem that there is a basic conflict in VFO design. The traditional view is that you drive the oscillator gently to keep the heat (and therefore drift) down, and follow it with an appropriate buffer to get the power up to the required level. This approach also maximises

*44 Manor Park Avenue, Princes Risborough, Bucks HP27 9AS. E-mail pirhodes@aol.com

phase noise. Conversely, if you drive it hard, then it becomes increasingly difficult (in my experience, next to impossible) to maintain acceptable frequency stability.

With the DDS approach, phase noise and drift are intrinsically small. The topic is covered shortly.

PIC 'N' MIX SUMMARY

BEFORE COVERING the essential theory, these are the features on offer should you adopt my software.

GENERAL SUMMARY

- PIC 'N' MIX replaces the functions of the crystal oscillator bank, VFO, mixer, bandpass filters, power driver and frequency counter associated with a conventional HF transceiver, with significantly enhanced features and lower cost. Not merely a VFO!
- Alternatively, it acts as a programmable and/or tuneable signal source with output from audio to 40MHz in 10Hz steps.
- All functions are controlled by either a multi-function tuning knob - or by a simple telephone keypad with 65 discrete key combinations recognised by the software.
- A large 6-digit 7-segment display with auto-ranging gives a resolution of 10Hz.
- Two independent VFOs provide IRT, ITT and cross-band operation.
- A variety of tuning and scanning modes provides operational flexibility.
- Any desired frequency may be entered directly from the keypad.
- The switch-on frequency and 9 band initialisation frequencies are user programmable.
- As are 10 frequency memories.
- Any three IF offsets (USB, LSB and CW separately) in the HF range may be entered.
- USB/LSB/CW selection outputs - and band switching outputs to the host transceiver are provided as a hardware option.
- Front panel LEDs provide status information and double as a bar-graph to show tuning rate.
- Finally, there are a number of possible physical layouts providing flexible outboard or integrated configurations.

ADMINISTRATIVE FEATURES

- The frequency accuracy is determined by a reference oscillator in the VHF range. You may use any crystal in the range 100MHz-125MHz and program the actual frequency into the software yourself.
- Final calibration and any subsequent correction for crystal ageing are achieved using the tuning knob to drive a trimmer in software. A physical trimmer, which would inevitably introduce drift and phase noise, is neither required nor provided.
- IF offsets may be entered from the keypad and/or trimmed to zero beat with the host transceiver carrier crystals.
- As an injection oscillator, the output frequency is the selected IF frequency plus or minus the desired frequency. The choice of high-side or low-side injection may be made 'on the fly', with the sideband selection outputs to the host being switched to correspond.

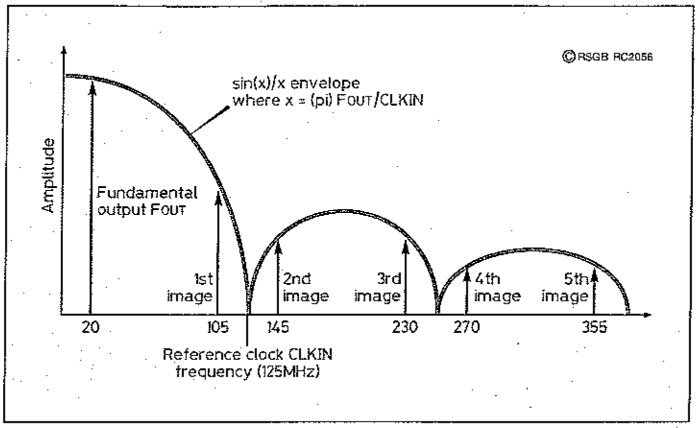

Fig 2: DDS output spectrum, showing the fundamental and image outputs.

OPERATIONAL FEATURES

- Intelligent tuning continuously monitors the speed and duration of tuning knob rotation to vary the tuning rate dynamically. Thus the longer and faster you turn the knob, the greater the tuning increments.
- A software flywheel engages automatically at high tuning speeds for rapid and/or large frequency excursions - and is disengaged by the slightest turn of the knob in the opposite direction.
- As opposed to traditional tuning where rotation of the knob alters frequency, a tuning rate option is provided whereby rotation of the knob alters the rate of frequency change - from zero to very fast. This is particularly useful for casually scanning around a band without having to continuously turn the knob.
- Guard channel operation provides normal tuning, but with a brief switch to another chosen spot frequency about every 20 seconds.
- Up to 10 memories may be programmed with frequency. As opposed to merely providing spot frequencies, they are also jumping off points for further tuning.
- Memory scanning mode cycles between the 10 memory frequencies at a speed determined by the tuning knob.
- Spot scanning switches between two chosen spot frequencies at a speed determined by the tuning knob.
- Range scanning tunes between two chosen limits with frequency increments determined by the tuning knob.

AD9850 DDS

THROUGHOUT THIS article, I have used the nomenclature used by Analog Devices in their data sheet [4] and only mentioned the features and configuration of the chip used in this project. There are others.

There is little you need to know about the internal workings of this device. The most significant consideration is that it contains the DAC - necessary to convert the digitally generated sine wave to analogue form - on the chip. This means you neither have to worry about specifying a suitable DAC nor interfacing it.

The basic block diagram is shown in **Fig 1**. There is a simple relationship between the output frequency F_{OUT}, the reference clock frequency CLKIN, and the 32-bit tuning word DPhase:- F_{OUT}= (DPhase x CLKIN)/2^{32}

Using a 125MHz clock, the highest frequency permitted, this gives us tuning increments of 0.0291Hz, orders of magnitude better than needed for this application. In practice this means that, using 10Hz tuning increments, an error of 0.0291Hz is significantly smaller than, for example, any drift on your carrier crystal.

Stability in a DDS system is the same (in parts per million) as that of the reference clock crystal oscillator. For example, if the 125MHz clock drifts by 10Hz, on 80m with 12.5MHz injection, you will drift by 1Hz. Phase noise on the DDS output is better than that of the reference clock - which contributes most of the system phase noise. The improvement is:

$$20 \log (CLKIN/F_{OUT}) \text{ dB}$$

Is it that simple? Unfortunately, not quite, for as well as generating the required frequency, aliased or image outputs are also present. This is inherent in any sampled signal and the output observes Nyquist's theorem. The aliased images are at multiples of the reference clock, CLKIN ± the output frequency F_{OUT}. Thus, with a clock frequency of 125MHz and the wanted output at 20MHz, the images will be at 105MHz (1st image), 145MHz (2nd image), 230MHz (3rd image), 270MHz (4th image), and so on.

Another consequence of Nyquist's theorem is that the maximum theoretical output frequency is half the reference clock frequency - but in practice, one third is usually taken as a rule-of-thumb limit - to provide a reasonable separation between the wanted signal and significant images.

The amplitude of the images follows a sine envelope, as shown in **Fig 2**. A low-pass filter is therefore inserted in the output to reduce the image outputs. On the highest bands, using a high IF, the Tx/Rx signal frequency tuned circuits offer further protection. Using the highest possible reference clock frequency obviously helps.

There are other discrete AM spurious outputs as a result of limitations in DAC technology. The significant ones are few in number and appear from the user's perspective to be at random frequencies. Analog Devices specify them as better than 50dB down, and the practical consequence of these is an occasional 'birdie'.

The remaining AM spurs form a continuous noise floor at about 70dB down, and these give rise to the greatest concern. A typical double balanced mixer will furnish about 40dB further suppression - so if the mixer is injected at +7dBm, weak birdies will be heard if the band noise is less than 2mV at the mixer RF port. On the LF bands with most receivers this will be academic but on, say, 10m a typical Rx will need to use an RF pre-amp with some 25dB net gain to both retain adequate sensitivity and to mask the noise floor. This topic will be much less of an issue when 12-bit DDS is available at affordable prices, but meanwhile this 10-bit DDS may not be suitable for all home-brew Rx topologies, particularly if you are reluctant to alter your gain distribution.

The final challenge with the AD9850 is its size, see **Fig 3**. Designed for surface mounting, it is truly microscopic. Much effort has gone into finding repeatable amateur methods of mounting it which do not compromise performance. Analog Devices recommend a 4-layer board with dedicated power and ground planes.

I tried it on double-sided board, both surface mounted and let into a slot so that it sat in the thickness of the PCB. I had no great problems hand-etching the boards - but found substantial difficulty in soldering the chip to the pads. The best I managed was with a medium-sized iron and a length of sharpened copper wire bound to the bit - and very fine solder. The propensity to bridge adjacent leads was enormous. Worst, it seemed impossible to maintain clean power and ground plane layouts - which ultimately prejudices the phase-noise performance.

After obtaining a batch of 50 unmarked devices in the same packaging at a rally and having destroyed many in the quest, I settled on a dead-bug approach with continuous power and ground planes - mounted as a sub-assembly on a DIL socket and with the input/output leads taken out to the DIL socket on fine wires.

This method is reproducible if you have average eyesight (or a good magnifier) and a short-term steady hand. The process is described in detail in Part Two of this article.

THE WORLD OF PICs

THE 16C84 IS ONE of a large and growing range of 8-bit microcontrollers. The devices vary according to speed, the amount of memory, built-in devices (including A-D converters) and other features. For the latest detail, consult the Arizona Microchip web site [5].

The 16C84 specifically is, in brief, an electrically reprogrammable device with 1k of program memory (ie room for 1024 instructions), 36 bytes of working data and 64 bytes of data EEPROM which survives power down, and 13 input/output pins.

Also, on the web site you will find the integrated development environment MPLAB, which was used exclusively in developing my software. It includes an editor, assembler and simulator. The latter is particularly useful, since you can progressively build and test code with your target chip simulated on the PC - no real hardware is needed. If you want to down-load MPLAB, watch your phone bill, because it is about 5MB when unzipped!

You can run elements of the software under DOS, but I used it exclusively under Windows® - at first under Windows 3.1 on a 386, and latterly under Windows 95 on a 486. Both were entirely satisfactory. C++ compilers are also available, but I haven't tried any of them, all my work being in assembler.

Of the various programmers available, I built TOPIC by David Tait [6] which runs out of the PC parallel port. You can also build ones for serial port operation and some even need no power supply, deriving their power from the port.

Having conducted the intellectual exercise of 'designing' some aspect of the software, the mechanics are easy enough. After typing in the code using the editor, you assemble it and then run it on the simulator - if necessary one instruction at a time - looking at intermediate and end results to see if it works. You can also check execution times. When you are happy, you download the software onto the PIC using the programmer (say, 10 seconds) and run your code in the real world. If you are careful, the PIC can be programmed *in situ* in the target environment, which speeds up the process enormously.

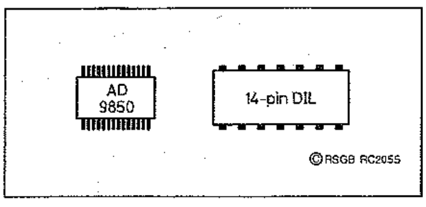

Fig 3: AD9850 28-lead Shrink Small Outline Package, drawn to size. The lead width and the gap between leads is about 1/3mm. A 14-pin DIL chip is shown for comparison.

The assembler language itself is easy to learn, with only 35 instructions. The art, it turns out, is usually not whether you can write something that works, but rather can you find an efficient enough way of doing it to squeeze it into the space without unduly compromising features, performance and ultimately maintainability? As Eric Morecambe once said "Composing good music is the same as composing bad music. It's just a matter of putting the notes in a different order." So it is with software!

So, if you have never written any software before and have a PC with at least temporary access to the Internet, you can have a go with no incremental cost. (Or you could buy a suitable second-hand PC for about £50 - and most Internet service providers offer a free trial period.)

Think of the range of applications - self-tuning ATUs, intelligent AGC generators, keyers and readers; in fact, any application involving control or logic is a potential candidate where one 18-pin DIL coupled with your intellect can replace acres of conventional hard-wired logic at trivial cost. Who says computers and amateur radio don't mix? In my view, these microcontrollers are going to dominate many aspects of home-brew construction before long.

THE INPUT/OUTPUT CHALLENGE

AS JUST MENTIONED, the 16C84 has 13 input/output (I/O) pins for controlling its environment. How many are actually needed? The following is the first-pass answer:

Inputs - total of 15, as follows:

PTT line monitoring 1
Keypad 4x3 12
Shaft encoder 2

Outputs - total of 74, as follows:

6 digits x 7 segments + decimal 48
Status LEDs 8
Band switch outputs 15
AD9850 control 3

Giving a grand total (apparently) of 89.

Clearly something has to give, and some supplemental hardware is needed. There is, however, one mitigating feature. The 13 I/O pins on the PIC can be used as either inputs or

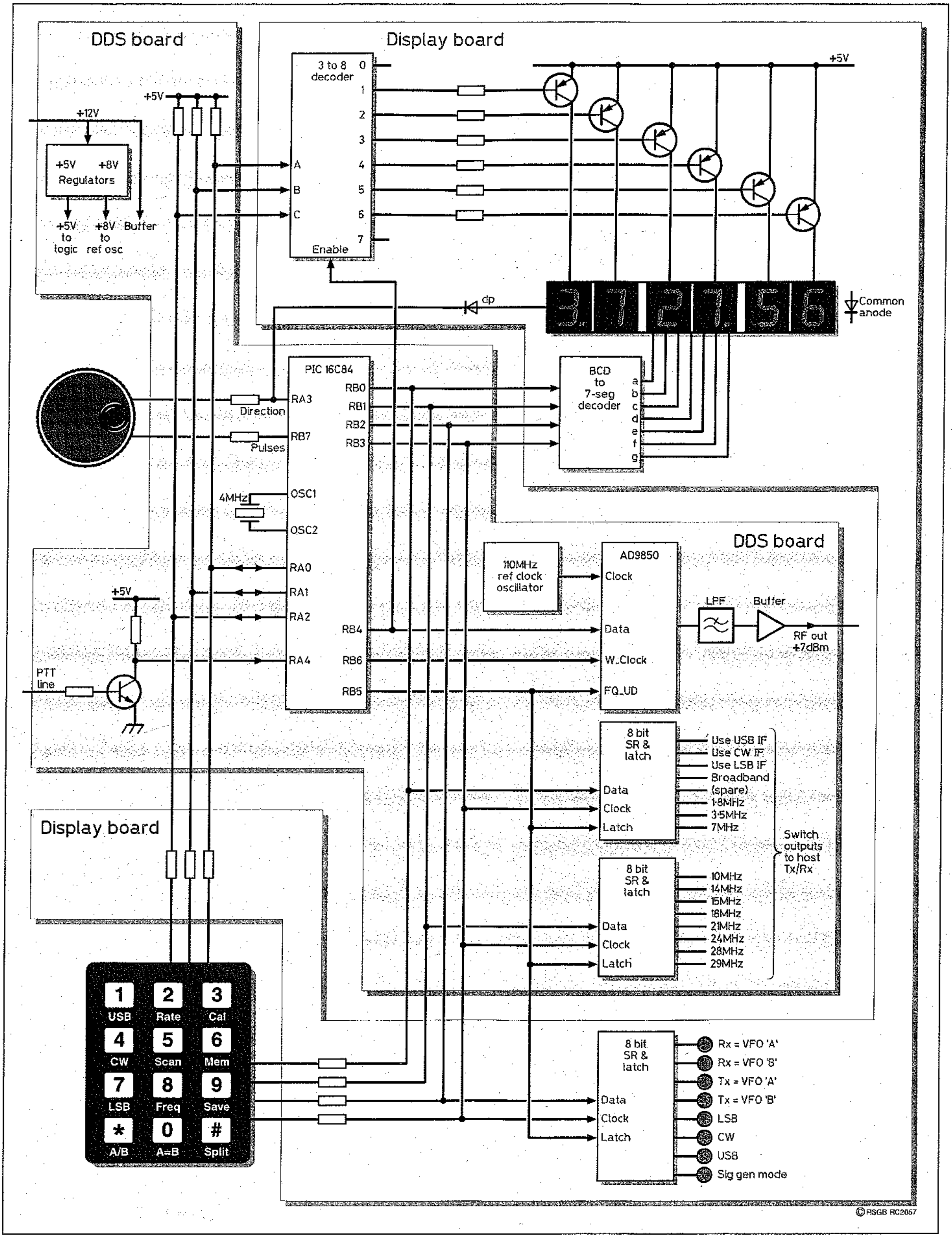

Fig 4: Pic 'N' Mix block diagram, illustrating PIC input/output allocations and physical partitioning. Besides power supply distribution and decoupling, all functional elements are shown.

outputs - and you can change them 'on the fly' in mid program, so with cunning they can be both!

Firstly, the 12 keypad switches aren't individually monitored. Each row is tested in turn, looking at each column in turn for key presses. This needs only seven I/O lines.

Next, rather than drive each display separately, each one is driven in turn - in rapid succession; ie they are multiplexed. Two low-cost decoder chips are added which gets the I/O count for the display segments down to seven. And of these, three outputs are in fact the same lines as used for the three inputs for the keypad columns; and the other four outputs are also multiplexed to drive the keypad rows.

Then three serial in, parallel out latches are added to handle status and band switching. These have three unique data lines, a common clock line (with all four again multiplexed with the display) - and a latch line shared with the AD9850. The final touch is to drive the decimal point output on the same line as the shaft encoder direction input.

If you have kept up with this, then you will agree that the total I/O count is now down to 13! **Fig 4** shows what it all looks like - and for good measure two lines are also shared with in situ programming. The only other viable approach would be a multi-PIC solution. It turns out to be marginally more expensive and significantly more intellectually demanding.

There remains one question. Can we achieve this multiplexing fast enough in the software so that the user sees 'instant' response and smooth 'continuous' operation? The answer, it transpires, is that it's not even difficult!

BUDGETS

Cost - If you were to buy all the electronic components from new, you should allow about £75.

Time - Construction time is obviously variable, but a good estimate would be one day each to make the PCBs and $1^1/_2$ days to assemble them. You will need about two hours to build the DDS sub-assembly. So this is not a weekend project, but it probably won't exceed two!

If you design your own software, times are impossible to estimate. But you can write some software to do one useful thing - say, generate a fixed DDS output frequency - very quickly. It's the integration of the whole which takes time.

Power - you need 12V DC at 400mA, smoothed but not necessarily regulated. From 10V-13V is acceptable.

REFERENCES

[1] *Technical Topics*, Pat Hawker, G3VA *RadCom*, Feb 1996

[2] *ARRL Handbook*, 1994, Chapter 10

[3] *Technical Topics*, Pat Hawker, G3VA *RadCom*, Sept 1996

[4] *CMOS,125MHz complete DDS Synthesiser*, Analog Devices, Rev 0.

[5] http://www.microchip.com

[6] http://www.man.ac.uk ♦

technical feedback

73kHz - THE FIRST YEAR

RADCOM, AUGUST 1997

THE MATHEMATICAL formula given in Reference 4 should have been:

$$L(\mu H) = \frac{N^2R^2}{460R + 1020X}$$

Roger Kendall, G0UPU

TECHNICAL TOPICS

RADCOM, SEPTEMBER 1998

THE CIRCUIT FOR 'Replacing The SL6270' won't work because the electrolytic capacitor on the gate of the FET is the wrong way round. I have checked the original *Electronics World* article, and it was wrong there too. The gate of the FET is driven *negative* by the diode, so the positive side of the capacitor must be grounded. The gain of the gain-controlled amplifier stage cannot decrease below 1, even when the FET is turned completely off, so an input of 4V would give an output of 4V and not 200mV. The 2N3819 FET has a very wide range of Gate-Source cut-off voltage. This parameter can vary from 0.8V to 8.0V, and it directly affects the level at which the AGC circuit establishes. I suggest the J309 or J310 FETs as an alternative, as they have a much smaller spread of characteristics.

D M Lauder, G0SNO

AN INTRODUCTION TO Q

RADCOM, SEPTEMBER 1998

SEVERAL READERS have pointed out that the relationship between decaying oscillations and Q factor that I introduced, is missing a factor of π. My apologies for this error. To estimate Q this way, count the number of cycles between amplitude 2.7 and amplitude 1.0, then multiply by π.

Peter Martinez, G3PLX

WHAT DO YOU KNOW?

RADCOM, NOVEMBER 1998

IN FIG 4, the right-most block of the receiver should have been labelled 'AF Amplifier'.

Steve White, G3ZVW

WHAT DO YOU KNOW?

RADCOM, JULY 1998

THE CORRECT answer to question 12 is 'b'. I wonder what suffix should have been used when halfway through the lock gates?

K Bartlett, G7LSR

[Assuming the lock opens into tidal waters, when the lower gates are open the vessel would be on tidal waters, irrespective of whether in the lock, part way through, or outside of it. This would put the station concerned 'At Sea', as BR68 puts it.

Under this condition, Maritime Mobile (/MM) would be called for. *-Ed*]

RADCOM NEWS

RADCOM, AUG & SEPT 1998

REFERRING TO the first European radio amateur to contact the USA, " . . .it was not 8AD but 8AB. Leon Deloy of Nice was the holder."

Allan Herridge, G3IDG

RX CALIBRATOR & TX MONITOR

RADCOM, JUNE 1998

THE CASE TYPE should be WB3, not WB2. Referring to Fig 7, the capacitor adjacent to the wire link, marked C11, should be C14; and the capacitor between R2 and IC2, marked C5, should be C4.

The rightmost IC is IC6, and the connection to the wiper of S1b connects to the wire link below R11.

J S Linfoot, G0CPP

2M & 70CM SIGNAL SOURCE

RADCOM, AUGUST 1998

IN FIG 1, the ground connection of IC1 should be to pin 7. Pin 1 is NC. The PCB layout is correct.

Godfrey Manning, G4GLM

TECHNICAL TOPICS

RADCOM, AUGUST 1998

IN THE CAPTION accompanying Fig 8, Rf1 and Rf2 should be 5kΩ (pre-set to 2k), and Rb1 and Rb2 should be 1MΩ.

D M Lauder, G0SNO

EUROTEK

RADCOM, NOVEMBER 1998

A READER has called my attention to a couple of errors.

Firstly, the MOSFET in the 50MHz amplifier should be IRF610 (not MRF610).

Secondly, the author was Klaaas Spaargaren, PA0KSB (not PA0KLS).

Erwin David, G4LQI