THE WEST RAND AMATEUR RADIO CLUB

December 2008
Volume 9, Issue 6

ANODE

Inside this issue:

Special points of interest:

- Contact details on back page (updated October)

Editor's Comments

Volume 9 Issue 6
December 2008

{——}

Lightning does strike in the same place twice!
Both of my old "Venus" phones lost their ringer when lightning struck the TV tower on Thursday afternoon.

{——}

Amateur Radio in SA

Doing a search on "Amateur Radio" on Google, produced 21000 hits on only SA pages.

Not surprisingly the SARL was top of the list. Their site now has a news box that members can type in current news to be featured in the next Radio ZS.

The West Rand's club page appeared on page 2 most of the way down the page.

{——}

When was the club founded?

We are still trying to verify the establishment date for the club. If you can help please get in touch.

{——}

(continued on page 6)

A "Different" Direct Conversion Receiver

Poor pre-detection selectivity, second-order mixer intermodulation and high-gain amplification problems are all reduced by using differential circuitry for direct frequency conversion.

In single-ended transmission systems, the signal voltage is taken with respect to a reference (usually "ground") which is assumed common to all stages of the receiver. Unfortunately this assumption cannot be realized in practice. In the differential mode, by which I mean that connections from one stage to the next are made with two wires and the signal information is contained in their voltage difference, with proper device selection and circuit design, push-pull type circuits naturally evolve which reduce second order intermodulation products and untuned signal detection. Induced hum and power supply ripple can similarly be reduced by using the high common-mode rejection ratios available in modern opamps. The beneficial side-effects are at least these three

- A balanced receiving antenna such as a small directional loop can be directly interfaced to the receiver without the phase and gain inaccuracies attendant with baluns
- R.f. circuit layout problems are drastically reduced because in a differential mode we don't worry

(continued on page 2)

A "Different" Direct Conversion Receiver

(continued from page 1)

about maintaining a constant impedance to ground (there is no "ground")

- If we ever cared about transmitting with a balanced antenna we would need both + and drive signals -- these are automatically available with a differential layout.

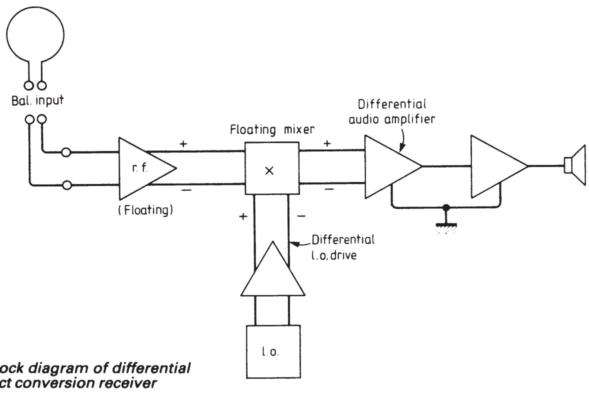


Fig. 1. Block diagram of differential direct conversion receiver

I have built the receiver whose block diagram appears in Fig. 1, and found that it does indeed achieve all of the advantages outlined above. I wouldn't waste your time describing something that doesn't work, so at this point you can either go ahead and duplicate my circuit, shown in Fig. 2, or read on and find out why it works.

I used a tuned-loop antenna to differentially drive the balanced inputs to IC1, an MC1590 untuned video amplifier with a voltage gain of about 10 and a 10 kOhm differential input impedance. A collection of small loop-antenna design equations is given in the appendix, which also explains why you should be using a loop antenna on the h.f. bands. The receiver's first stage performs two functions: it provides constant input and output impedances and acts as a buffer between antenna and mixer, and it provides about 20dB gain for the inefficient loop antenna. Due to atmospheric noise in the h.f. bands for which this receiver was designed, noise figure and antenna efficiency or G/T are unimportant factors. The important consideration is dynamic range, and for the direct conversion receiver, second-order intermodulation products. They must be low to reduce untuned signal detection. In the circuit, the MC1590 operates in a pushpull mode which suppresses this type of non-linearity by generating two equal and opposing non-linearities (when the device is driven non-linear) and cancels even order harmonics.

After the antenna is amplified, it is differentially converted to the audio frequency range by a doubly balanced mixer circuit incorporating a CA3049 | CA3086. In fact, this circuit could be considered "triply balanced" as there are no unbalanced ports at all (don't fall over on that one). The CA3049, to be described later, operates as a differential amplifier with its outputs switched in polarity at the local oscillator rate. The fact that gain is available in the upper switching transistors whose bases are at pins 1, 4, 7 and 10 and also the fact that the L. o. signal is a square wave ensures fast switching transitions from positive gain to exactly the same but negative gain. Lack of symmetry during the short switching transition time is known to cause intermodulation products. At h. f. relatively faster switching can be achieved with active devices using squared-up drive waveforms than can be achieved with diode ring-type balanced mixers. Up to 30 MHz, therefore, I think that the integrated bipolar transistor type of balanced mixer driven by a square wave is superior to other types of mixers.

It will, however, respond to signals at odd harmonics of the L.O. almost as well as it does to the fundamental; these can be easily filtered out as they are widely separated in frequency.

At points C and D of Fig. 2, then, there is a differential audio signal which is amplified differentially by IC2. These are the two op-amps connected in the classical differential instrumentation amplifier configuration with a differential gain of 40dB. This is where power line hum gets rejected and the amount of rejection depends on how closely you can match the gains to maximize common-mode

(Continued on page 3)

A "Different" Direct Conversion Receiver

(Continued from page 2)

rejection. You might want to make the gain of IC2(b) slightly variable (±10%) by using a 910 Ohm feedback resistor in series with a 200 Ohm potentiometer instead of the fixed 1 kOhm resistor shown. Residual power supply ripple from the 6.2-volt zener diode is in-phase at the inputs to IC2 but the desired signals are 180 degrees out of phase with each other at these points. With the circuit shown, differential signals get amplified by 40dB, whereas changes in the zener diode reference voltage come through unchanged. A 40dB improvement in signalto-ripple ratio over a single-ended design is therefore effected.

At the output of IC2 the signal has been amplified and filtered to a level where it is relatively impervious to the interference effects important to a direct-conversion receiver. From here on, we can use standard single-ended operational amplifier circuitry with one input, one output, and ground as a reference. I used a three-section Tchebychev 0.1 dB ripple lowpass active filter with cutoff at 500Hz for c.w. operation to drive the output stage. The lowpass filtering results in better c.w. copy than a high-Q, narrow-band-single, section active filter because of less ringing.

The local oscillator has to be stable, yet simple. In this circuit, oscillator transistor and integrated circuit provide an adequately stable differential local oscillator signal over about a 30% bandwidth centred at 3.6MHz. As the circuit shown is electrically tuneable, frequency stability depends on how stable a tuning voltage you can generate. After building and using the receiver, I came to the conclusion that unless some type of closed-loop frequency synthesis technique is used to stabilize a free-running wide-band v.c.o. mechanical tuning is preferable. I am, therefore, presently looking at electronically switched LC networks which may be the subject of a future article.

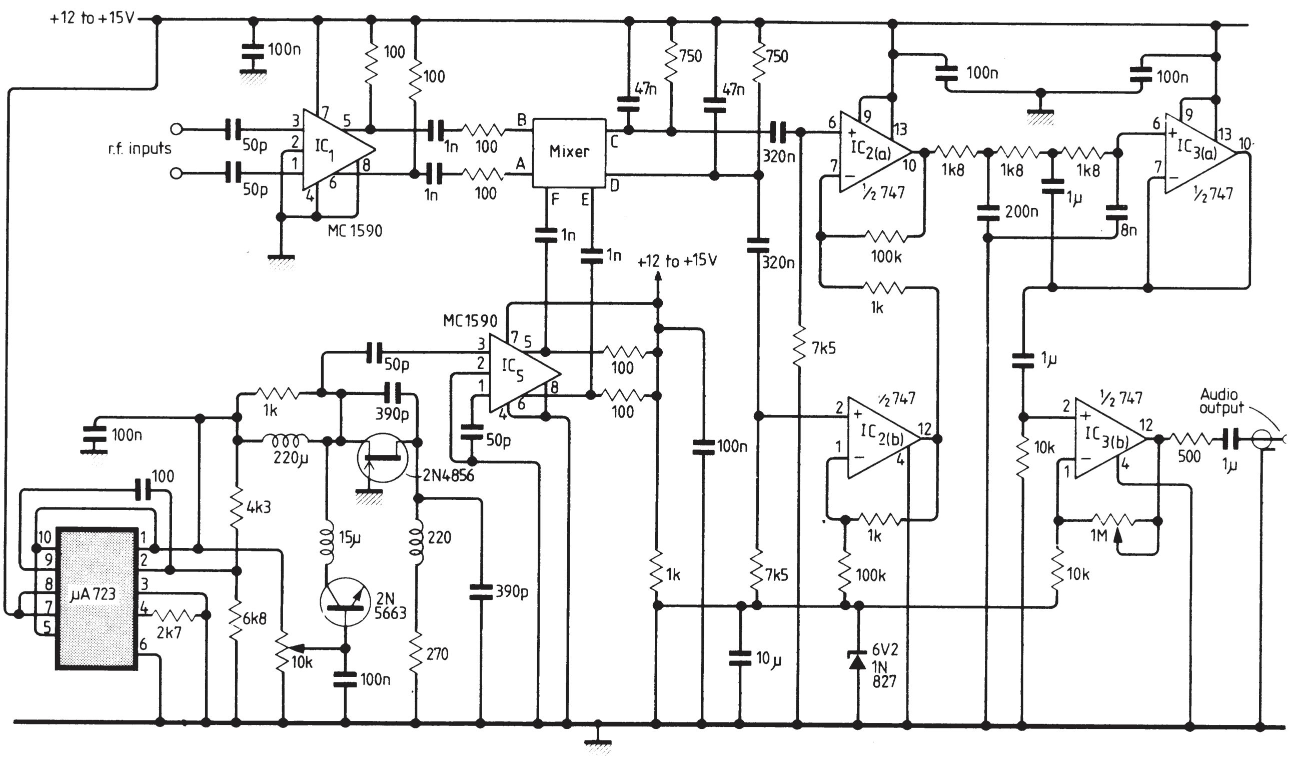


fig. 2 Differential direct conversion receiver shown uses only five IC's and three discrete transistors.

(continued on page 4)

A "Different" Direct Conversion Receiver

(Continued from page 3)

With the differential signals available, we can take a very different perspective on circuit layout. By maintaining relatively high common-mode impedances compared with differential-mode impedances, we can do without a ground plane.

Actually, the presence of a ground plane may negate some of the advantages achieved by the differential concept. If you intend to build a circuit of this type, feel perfectly confident that you can do it on perforated board with no decrease in the performance attributes outlined here. In fact. the further away you can get the balanced r.f. circuitry from a ground plane, the better off you will be.

Integrated circuits should still be bypassed at their supply voltage terminals. All differential signal leads should be tightly twisted pairs, as short as possible, and as far away from chassis or circuit ground as possible. This applies particularly to the following four transmission paths:

- connections from the balanced antenna outputs to pins 1 and 3 of IC1
- connections from pins 5 and 6 of IC1, to the A and B inputs of the mixer
- connections from points C and D of the mixer to pins 6 and 2 Of IC2
- L.O. connection from pins 5 and 6 Of IC5 to points E and F of the mixer.

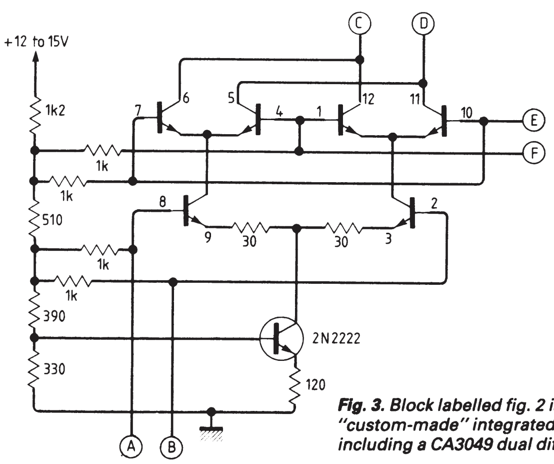
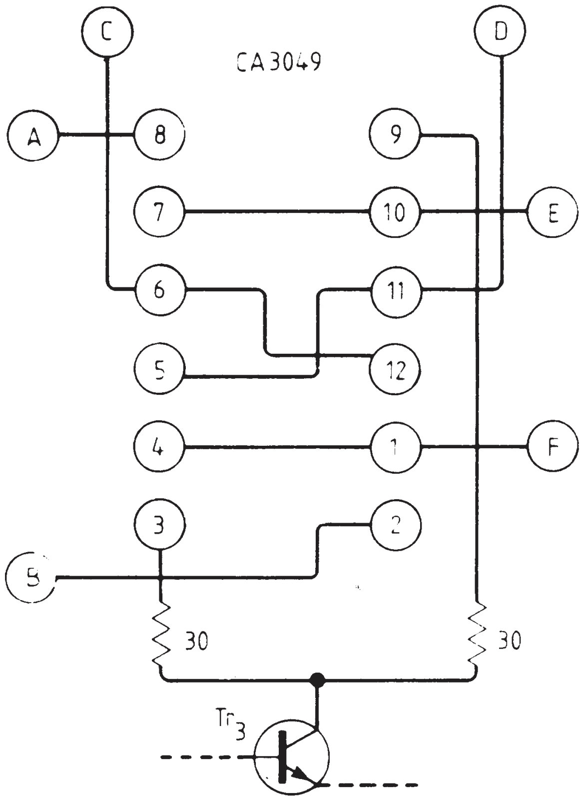


Fig. 3. Block labelled fig. 2 is really a "custom-made" integrated circuit including a CA3049 dual diff-pair.

The CA3049 should be laid out and wired as shown in Figs 3 & 4 which relate to the terminal connections of Fig. 2. This device is a high-frequency, layout sensitive component and we are using its high frequency capabilities to reduce intermodulation distortion, as described earlier. (The nomenclature of points A, B, C, etc, on the mixer of Fig. 2 has been



generated mainly to reduce the clutter which would appear on a detailed circuit schematic, but which is much more simple in an actual physical realization, as Fig. 4 shows.

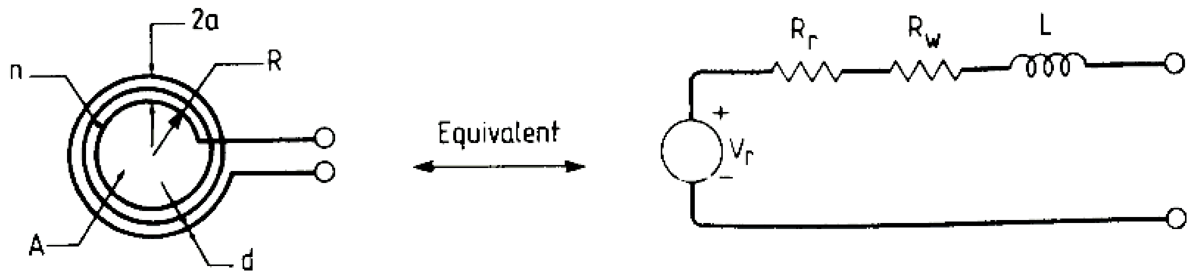
Appendix

Small loop antennas are useful for receiving systems in the h.f. band because their dipole-like directional characteristics are preserved

(Continued on page 5)

A "Different" Direct Conversion Receiver

(Continued from page 4)
even though their size is orders of magnitude smaller than an actual dipole at these frequencies. Their efficiency is extremely low, variables than can be measured or even defined; however, the equation does give an answer which probably is within an order of magnitude of the actual number.



but at h.f. where most of the receiver noise is due to random atmospheric disturbances, it doesn't matter if you have a lossy, noisy receiver front end. A small loop's main attribute is its ability to easily null out a coherent man made interfering signal by simple physical re-orientation of the loop. A 3.5MHz receiving loop antenna can be as small as two feet in diameter and have the same directional properties as a rotatable 80-metre dipole which would be about 130 feet long. Null depths of around 20dB can be achieved if phase and gain of the balanced antenna output are preserved. Naturally, you'll need a differential direct conversion receiver to do a good job at this.

For convenience in building small loop antennas, I've put together a number of design equations from various sources, shown on the right. Included is one relationship assembled by me which allows you to calculate the approximate r.f. voltage received from a transmitter as a function of the transmitter's range, power, frequency, and your receive loop antenna parameters. Of course the range depends on how the signal got to you, i.e. how many ionospheric bounces it took and how high the reflecting ionospheric layer was at the time.

Polarization of the received wave also plays a part, as does the angle of arrival from the sky with respect to the loop's orientation (elevation as opposed to azimuth). One can easily get carried away trying to account for more

The small loop antenna has an equivalent circuit which may be derived from its physical parameters.

Rr = 31,200 (nA / lambda ^ 2) ^ 2 Ohms

Rw = 2.61 * 10 ^ -7 SQR(f(2n R / d)) Ohms for copper

L = n ^ 2 u0 R(loge (8R / a) - 2) nanoHenries

Vr = (lambda / 4 pi D) SQR(8 Pt Rr) Volts

Definitions

R mean radius of loop
A area of loop, pi R ^ 2
d wire diameter
n number of turns in loop
a radius of bundle of wires in loop or d/2 if n= 1
Vr peak amplitude of received signal
Rr radiation resistance
Rw conductor loss resistance with skin effect
L inductance of loop (nanoHenries)
f frequency (Hertz)
X free space wavelength
g 12.56 nH/cm
Pt transmitted power (watts)
D range (same units as lambda)
u0 12.56 nanoHenries / cm

By Paul E. Gili, WA1WQH
From Wireless World September 1982

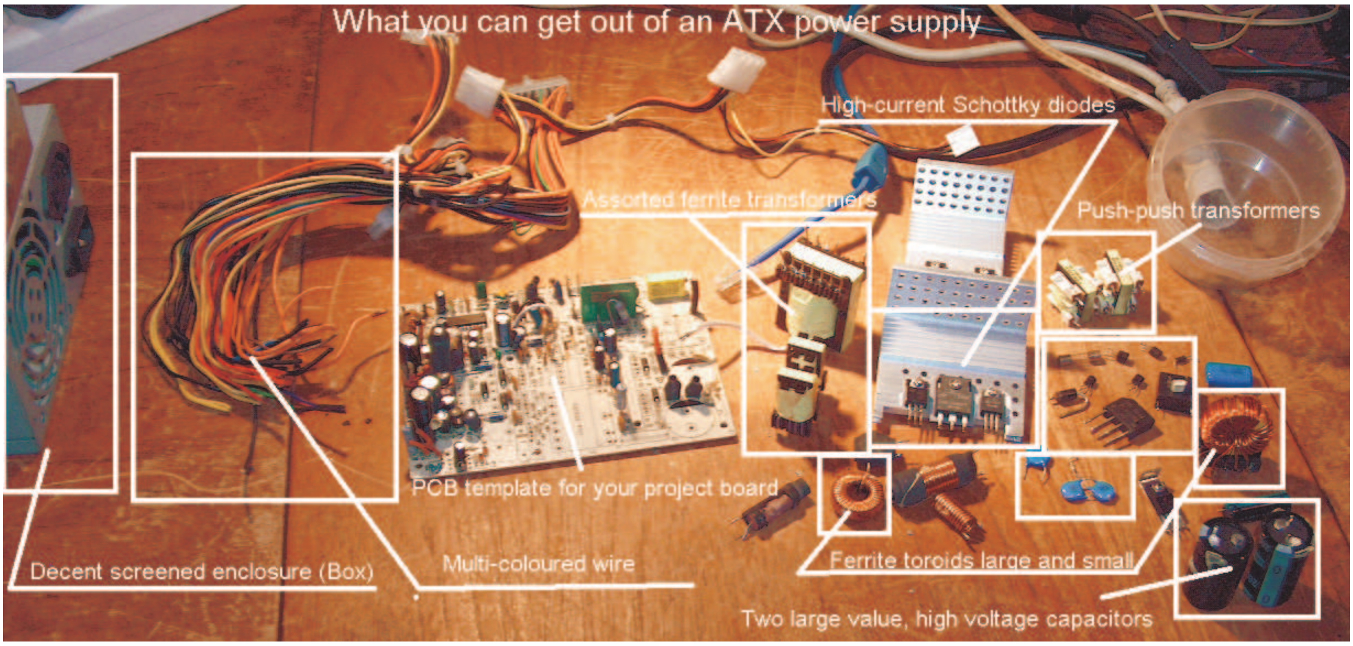
Editor's Comments

(Continued from page 1)

Ever heard of Macrovision?

It's the copy protection of the 80's. It is what stops you from 'backing up' or copying your old VHS tapes to DVD on your personal video recorder or PC.

{—}



What you can get out of an ATX power supply

With the lightning season upon us, the usual crop of blown ATX power supplies are starting to trickle in. These are the "modern" power supplies used in PC's that have no protection against mains borne surges.

As we have seen at the Ham-Comp sessions, you can use the parts for a multitude of projects. OM Dave, ZR6AOC demonstrated a direct conversion receiver built into the power supply case.

{—}

The HAM and programming

What does he use?
[With a huge variety of programming languages to choose from, what does the Amateur need to learn to program ?]

The average radio ham is not a programmer normally but that's not to say he cannot write programs for his computer(s). I have said on many of my training courses that "If you can write a shopping list, you can write a program". Formal training shouldn't stop you from writing good functional programs for your computer.

How To start writing programs

Set out what you want to do in any form that is comfortable to you. [I usually do this in WordPad but you can use Excel or any other medium you like] A simple list of requirements as headings will usually suffice. Then copy these further down the page and insert expansions on each point to explain to someone else what you want

(continued on page 7)

Editor's Comments

(Continued from page 6)
the program to do.

Having broken the stages down into simple ones, it becomes very easy to 'code' the various sections. You can also go back later and insert additional functions that you thought of during the process. You can also put in pictures at points in the text to show what sort of dialogue window you want the program to have. [You can capture using Alt+Print Screen and paste into WordPad/Excel etc.]

This definition of the program will serve you as your specification should you decide to hire/ persuade/coerce someone else to write the program!

Writing 'pseudo code'

A lot of programmers [not necessarily professional] will write the program first in what is called 'pseudo code'. This is not really a programming language but a way of thinking through the program code.

An example of pseudo code

NOTE I put all code and examples of code into Courier font (looks like a typewriter) to highlight it.

***

> Since a string that is provided to our engine will most likely (but not always) require searching and replacing, I'll wrap all the search-and-replace functionality into a single method called Expand(). The pseudo-code for Expand() looks like this:

```
Accept a string and an SQL statement
(so that the correct entity or attribute
data can be replaced)
Find first occurrence of a tag
While an occurrence is found
  Get the entire tag and its exact posi-
tion
  Get a replacement value for the tag **
  Strip out the tag and replace it with
the new data
  Find another tag occurrence in the
string
Loop
```

> Notice the asterisked line above, which gets a tag's replacement value. Let's pseudo-code its functionality as well:

```
Accept a string (the tag) and an SQL
statement (this was passed in to the
above Expand() method)
Determine if the string matches any of
our string constant tags
  If so, get the value and exit
Get the tag's data (which is a column
name)
Create a recordset using the SQL state-
ment
Get the data for the column name from
the recordset
Exit program
```

***

When you have this imitation program written, it will be a simple matter to translate the code to the programming language of your choice.

What low cost languages are available? [of particular relevance to the HAM]

Basically what can you get for free and are they of any real value. Well most free stuff nowadays means that there are some later 'strings' or baggage. The exception to this are the "Open Source" languages which are "free" as in air. GNU C/C++ is the main language for compiling the Linux operating system and it programs. It is totally free and downloaded from the Internet for free as well.

However there are a great number of training languages which generally means that they come to you free of charge, (well maybe the time on-line costs) but the cost of usage is zero. Unfortunately some of these "free" languages have 'small print' that says you cannot write any-

(continued on page 8)

Editor's Comments

(Continued from page 7)

thing useful for profit. So you may not sell anything written in these languages. Borland's Delphi is one such language. If you purchase the full package, you can make programs for profit.

Basic and its derivatives

Basic was originally designed as a training language. Nowadays versions usually are 'Visual' that is they are meant for a graphical user interface such as MS Windows or X. The language is supported in all versions of Windows(TM) and can access the operating system, its 'objects' and applications through the use of COM/DCOM. Versions 1,2 and 3 were 16 bit in executable (.EXE) and could even work in Windows 3.x. Version 4 is/was available as a 'training' version but was hamstrung by not using too many forms (windows) and code modules. Version 5 was available for 'free' as well but could only create ActiveX type controls for use on web pages.

Visual Basic (Last version 6 Service Pack 6) was the last in a line of Basics that go back to the days of DOS/CPM and before. 'VB.NET' versions 2003/4/5 are available as 'training' editions or 'Express' versions from M$ but these are NOT Basic languages any more. Whilst you are supposed to be able to do so much more with them, you will take a lot longer to learn and familiarise yourself with these .NET languages. .NET languages are .NOT a good starting point for writing simple programs. Intriguingly QBasic that came out with MSDOS 5.0, still functions under Windows XP. But with certain restrictions imposed by the operating system.

Visual Basic Scripting is now at version 5.6 and is very good at automating web pages and tasks in MS Windows. The language is supported in all versions of Windows(TM) and can access the operating system, its 'objects' and applications through the use of COM/DCOM.

Other Basics are available for 'free' and even compile to executables for Windows and Linux. GAMBAS is one such BASIC language environment for the Linux system. It produces very acceptable programs for the X Window system. Also useful are Liberty Basic and Quick Basic types. These are not necessarily to be written off as useless. They can produce quick and very user friendly programs which run either as text based interface or console mode or even produce graphic windows for a user interface.

Pascal and its derivatives

Pascal/Turbo Pascal was incredibly popular some years ago as it was strict in its methods of assigning variables and coding. It wouldn't let you make mistakes in your program and would not compile to object code if you did make a mistake. Nowadays the version is also a visual programming language known as Delphi or Kylix by Borland. Windows and Linux programs can be written with the same code base and can be used on either operating system. Unfortunately Borland has lost interest in this market and has sold Delphi to another company.

C/C++ and its derivatives

The C language was used to write the original Unix operating system and is still reckoned to be one of the most efficient programming languages bar an assembler (Machine code). Its support is wide ranging from web sites and tutorials to free downloads of source code and working programs.

C++ (c plus plus) has enjoyed great popularity with Windows and Unix programmers. It is highly structured and less tolerant of programmer's mistakes than C. In C you could bring down the entire system (usually DOS sometimes Windows) with a "BUG" in your program. In C++ its compiler will usually 'catch' the bug before you get to run the program. If you are working to build a small efficient program though, C

(continued on page 9)

Editor's Comments

(Continued from page 8)

still produces smaller and faster programs. The GNU C/C++ compiler is available for both Linux and Windows.

Assembler (Whoa! Which processor)

Well if you really want to program you should consider that 'real programmers' only write in machine code. (They also dream in hexadecimal.) Also 'wimp programmers' or 'hackers' would write in an assembler language for a specific processor. Intriguingly the one for the PC written by MS is still available for free download. Simply get hold of the Windows 98 DDK.

All of the above?

You can write in Basic and use an assembler to write code for use by the Basic program. This will speed up some operations. But it will make your program complex and tricky to keep updated. But if you are writing for yourself and maybe a few others, it can be easily managed.

You can also write programs for Windows in VB and call routines written in C/C++ that have been placed into a DLL (Dynamic Link Library). This can provide small sized programs that run rings round professionally written C++ applications. Little wonder a lot of starting out professionals have written programs or applications this way.

Personally I am very much in favour of BASIC, as it is easy to learn and use, as well as debug on a current or last years PC.

Which operating system should I write programs for?

What processor are you going to write programs for? Do you know where and how the program is going to be used? These and other questions should be answered carefully with the original layout of the program done above. You could write a simple program for use by HAM's on your cell phone or PDA.

CP/M / Z80 - (Was I serious?) Well yes I was. There are a lot of 'junked' CP/M machines out there that are still in working order. The Z80/8080 processors don't need noisy cooling fans or much in the way of RFI screening either. There are zillions of Z80 based modems as well that can do quite a bit in the way of processing. Not to mention most of the Nokia cell phones are based on the Z80.

Symbian (What's that?) Well its another operating system used in Cell phones. Take a Google at it.

Linux/Unix - There are now a lot of programs written by amateurs for use on the Linux operating system. Most are free to use and free to download.

MS Windows - Quite a number of programs have been written by amateurs for use under the MS Windows operating system. Most are free but not all as the amateurs in question don't want to be amateur for the rest of their life. Writing the program shouldn't take you long but the learning of the operating system will take some time.

DOS - (Really?) Well again I am. MS DOS and other Disk Operating Systems are now available for little or no cost either second hand or 'thrown out' along with the hardware. Ask any thirty-plus young HAM, you will probably find he has one in the garage that he used to use for Packet, RTTY or Morse. With the advent of "DosBox", the free utility that emulates a DOS environment, its possible to run programs that work under Linux or Windows XP/Vista.

John Brock
(Formerly ZS6BZF Nowadays 'PieRat')

The West Rand Amateur Radio Club
Established in 1948
KG33XU 26.14122 South - 27.91870 East

P.O. Box 562
Roodepoort
1725

Phone: **082 342 3280** (Chairman)
Email: **zs6wr.club@gmail.com**
Web page: www.jbcs.co.za/ham_radio

Bulletins (Sundays at …)
11h15 Start of call in of stations
11h30 Main bulletin start

Frequencies
439.000MHz 7.6MHz split
Input: 431.4MHz (West Rand Repeater)
145,625 MHz (West Rand Repeater)
10,135 MHz (HF Relay)

Radio Amateurs do it with more frequency!

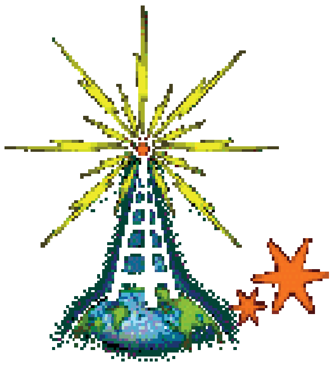
Chairman	Joop Hesp	ZS6C	082 342 3280	**zs6wr.club@gmail.com OR joophesp@telkomsa.net**
Vice Chairman	Geoff	ZS6GRL	082 546 5446	glevey@gmail.com
Secretary (Anode)	John Brock	'PieRat'	011 768 1626	brockjk@gmail.com
Treasurer	Craig Woods	ZS6CRW	082 700 0163	craig.woods@absamail.co.za
Member	Romeo Nardini	ZS6ARQ	082 552 4440	roshelec@global.co.za
Member (Repeater)	Phillip	ZS6PVT	083 267 3835	phillipvt@sse.co.za
Member (Technical)	Ron	ZR6RON	082 902 8343	ronnie@calidus.co.za
SARL Liaison	Willem	ZS6WWJ	082 890 6775	marie.w@absamail.co.za

West Rand members - we need your input!

To make this the best ham radio magazine in South Africa we need your input. Please submit articles, comments, suggestions etc.

Please send plain text with no formatting to the email address below.

In July 2003, we re-published an Anode Compendium on CD. It has the issues from July 2000 until June 2005. This included the new Adobe reader. It has been updated, check with the chairman for details.



We need your input! Email us articles, comments and suggestions please. zs6wr.club@gmail.com