

April 2009

Volume 9, Issue 10

# ANODE

## Inside this issue:

Editor's Comments	1
Amateur Radio and PICs.	1
Simple PIC Programmers.	7
Ham Radio and the PIC.	8

## Editor's Comments

**April 2009**  
**Volume 9, Issue 10**

This month as promised I have written an article about PICs. These low-cost devices have replaced microprocessors in a large number of applications over the last ten years.

### Happy Easter!

There will be a boot sale on the 2nd of May this year. Starting at

12:00. There will also be an auction later in the afternoon to sell used amateur radio equipment from a deceased estate. A catalogue will be available with reserve prices. If you would like to sell some amateur radio equipment at this auction, please contact the secretary of the West Rand Amateur Radio club. [zs6wr.club@gmail.com]

The next meeting is on the 13th of April. It will feature a presentation

*(continued on page 2)*

## Amateur Radio and PICs

### PIC Stuff (Links)

[see further down for the real stuff]

### Microchip Web Site

<http://www.microchip.com/>

### Open Source PIC stuff

<http://opensourcepic.org/>

### gputils

<http://gputils.sourceforge.net/>

### gpsim

<http://www.dattalo.com/gnupic/gpsim.html>

### gpsimWin32

<http://gpsim.sourceforge.net/gpsimWin32/gpsimWin32.html>

### PIC Microcontroller Programmers

<http://www.piclist.com/techref/>

[microchip/devprogs.htm](http://microchip/devprogs.htm)

### GNU PIC

<http://www.gnupic.org/>

### PicForth

<http://www.rfc1149.net/development/picforth>

### SDCC - Small Device C Compiler

<http://sdcc.sourceforge.net/>

### Microchip Open Source Forum

<http://www.microchip.com/forums/tt.aspx?forumid=182>

### Quozl's Open Source - Electronics

<http://quozl.linux.org.au/?category=Electronics>

### Intro to Programming PIC Microcontrollers in Linux

<http://www.micahcarrick.com/04-19->

*(continued on page 3)*

### Special points of interest:

- Contact details on back page (corrected & updated)
- Ham-Comp Latest on web site.

# Editor's Comments

(continued from page 1)  
from Jawug.

## Electronics Info Online

<http://www.electronicsonline.com/CircuitBook/CompleteList.html>

## The U.S. electrical grid: How big of a cyber target is it?

<http://blogs.zdnet.com/BTL/?p=16001&tag=nl.e589>

## Costing Africa's new cable ties

<http://news.bbc.co.uk/1/hi/world/africa/7987812.stm>

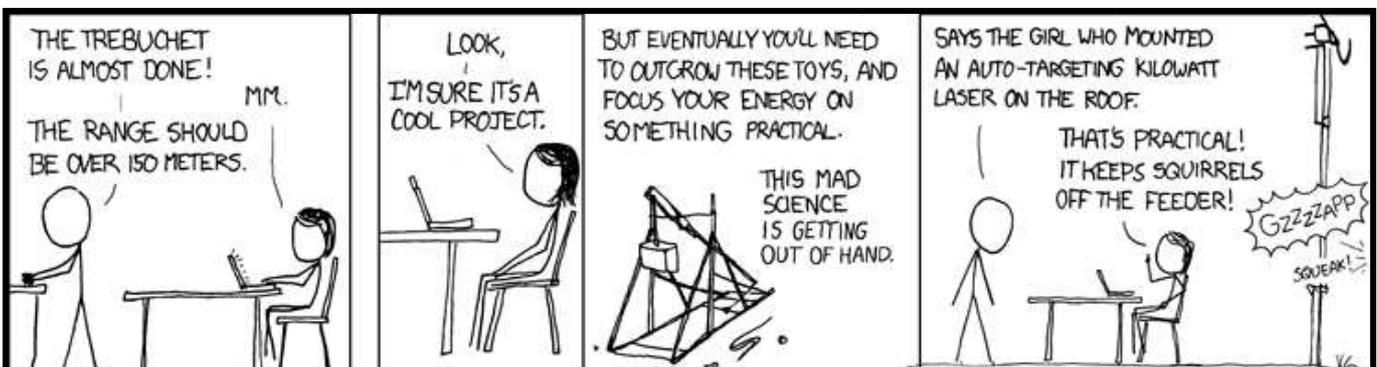
## Spock gives fans Star Trek treat

<http://news.bbc.co.uk/2/hi/entertainment/7989146.stm>

Oh, and I wasn't kidding about this by the way. —>

### Warning!

Do NOT break your or neighbour's energy saver light bulb. There is around 5 mg of mercury in it. If you do break it, call in a HazMat clean-up crew immediately. Do NOT vacuum it up as this will spread the mercury into the atmosphere around you. See:-  
<http://www.dailymail.co.uk/sciencetech/article-506347/An-energy-saving-bulb-gone--evacuate-room-now.html>



From:- <http://xkcd.com/382/>

# Amateur Radio and PICs

(Continued from page 1)

2005/pic-programming-linux.html

## Programming Languages

[http://www.opencircuits.com/Programming\\_Languages](http://www.opencircuits.com/Programming_Languages)

### Notes:

One of the most "off putting" hazards of using the latest technology is the dropping of functions from the latest hardware. The most annoying to a great many tech users is the loss of the RS232 serial communications adapter. Virtually all the new PC's do not have even one serial port! If your PIC programmer is the cheap type as recommended by a lot of web sites, it will not work with one of the new PC's. To make matters worse, the software having been written last century, probably won't work with a USB serial port adapter. The programmer also may sometimes work with the adapter and sometimes not due to the lack of 15 Volts on the RS232 pins. The PCI RS232 cards might work with the software but the programming might not program the PIC reliably. Again the RS232 port will not have 15 Volts on the pins. The original RS232 spec says +/- 15 Volts.

- I recommend the use of separate power supply with the programmer. A small "battery eliminator" is quite adequate to supply the miniscule current and 14 Volts to program the PIC.
- I also recommend using a 80386/80486 Linux PC as your dedicated PIC compiler/assembler/programmer PC. You will then be able to use all of the GNU (Open Source) tools available as well as the serial / parallel programming hardware. The Linux box can also be networked (Ethernet or WiFi) to your Windows box to operate it with out a screen or keyboard...
- Alternatively decide on the motherboard in your new PC and ask the assembler to use a motherboard with a single serial port on the board such as the Intel "Dragontail". You might have to make a flat ribbon cable for it as the suppliers don't believe in ordering the cable from Intel. This port is Com1 and will allow the use of the old (last century) software.
- If you are really gung ho about writing programs in assembler for your PIC, then the only assembler for you is the MPLAB assembler. It is "free" and downloadable from Microchip. BUT if you want to write your programs in another high level language, you will have to obtain the GNU assembler and language from the Internet. The language of your choice will compile to an intermediate file of assembler and then use the GNU AS assembler. For example most serious users will use C or C++ to write their code which is very close to assembler and is also used in Linux to compile the programs to executables. It is possible to write your program in BASIC and compile it using a GNU compiler to the intermediate assembler file. You would then run the assembler to produce the hex code file for download to the programmer. Only if you are really a masochist consider writing your program in ADA or ALGOL and then assembling it to the PIC. But if you are interested in FORTH, you can visit PicForth. (Link above).
- If you want to stay in your programming "comfort zone" then you should download SDCC from the link above. As the **gptools** also runs on Windows it complements this very well.
- Examples! There are loads and loads of examples. Some are free some are not. Also there are some amateurs that will sell you the source code for their work. And there are some only too happy to have you 'correct' or modify their code. Especially if you share the code with them and others. For reliable code

(continued on page 4)

# Amateur Radio and PICs

(Continued from page 3)

that usually has been checked, try the university web sites. These generally supply example code snippets and libraries that are checked by building working models.

For a general overview of the GNU PIC Microcontroller scene, check out:-

<http://www.piclist.com/techref/microchip/devprogs.htm>

## What is SDCC? [extracted from web site]

SDCC is a retargettable, optimizing ANSI - C compiler that targets the Intel 8051, Maxim 80DS390, Zilog Z80 and the Motorola 68HC08 based MCUs. Work is in progress on supporting the Microchip PIC16 and PIC18 series. SDCC is Free Open Source Software, distributed under GNU General Public License (GPL).

Some of the features include:

- SDCC ASXXXX and ASLINK, a retargettable assembler and linker, is Free Open Source Software, distributed under GNU General Public License (GPL).
- extensive MCU specific language extensions, allowing effective use of the underlying hardware.
- a host of standard optimizations such as global sub expression elimination, loop optimizations (loop invariant, strength reduction of induction variables and loop reversing ), constant folding and propagation, copy propagation, dead code elimination and jump tables for 'switch' statements.
- MCU specific optimizations, including a global register allocator.
- adaptable MCU specific backend that should be well suited for other 8 bit MCUs
- independent rule based peep hole optimizer.
- a full range of data types: char (8 bits, 1 byte), short (16 bits, 2 bytes), int (16 bits, 2

bytes), long (32 bit, 4 bytes) and float (4 byte IEEE).

- the ability to add inline assembler code anywhere in a function.
- the ability to report on the complexity of a function to help decide what should be rewritten in assembler.
- a good selection of automated regression tests.

SDCC also comes with the source level debugger SDCDB, using the current version of Daniel's s51 simulator.

SDCC was written by Sandeep Dutta and released under a GPL license. Since its initial release there have been numerous bug fixes and improvements. As of December 1999, the code was moved to SourceForge where all the "users turned developers" can access the same source tree. SDCC is constantly being updated with all the users' and developers' input.

AVR and gbz80 targets are no longer maintained. [That's the AVR PIC and the Gameboy Z80 - ED]

{---

## A PIC C Code Wizard - By Carlos Buelna

<http://www.codeproject.com/KB/applications/picpc-wiz.aspx>

## <http://www.eclipse.org/cdt/>

Eclipse C/C++ Development Tooling - CDT

## Quincy 2005 A Simple IDE for C/C++

<http://www.codecutter.net/tools/quincy/>

## BloodshedSoftware

<http://www.bloodshed.net/devcpp.html>

## Writing your program

Whilst your first "Hello World!" program can

(Continued on page 5)

# Amateur Radio and PICs

*(Continued from page 4)*

be written in Notepad or vi, you will very soon need a helpful programming environment. This is really a personal preference thing, so I am only going to point you in the direction I went. You will probably find one that suits you after you have written a few projects.

Perhaps the only IDE (Integrated Development Environment) you should consider on an amateur radio budget, is the Eclipse IDE. The link above is for the C/C++ environment. You can check out the original at:-  
<http://www.eclipse.org/>

[Or if you just want a simple programming environment, try Quincy or Dev C++.]

"Eclipse is an open source community, whose projects are focused on building an open development platform comprised of extensible frameworks, tools and runtimes for building, deploying and managing software across the lifecycle. The Eclipse Foundation is a not-for-profit, member supported corporation that hosts the Eclipse projects and helps cultivate both an open source community and an ecosystem of complementary products and services.

The Eclipse Project was originally created by IBM in November 2001 and supported by a consortium of software vendors. The Eclipse Foundation was created in January 2004 as an independent not-for-profit corporation to act as the steward of the Eclipse community. The independent not-for-profit corporation was created to allow a vendor neutral and open, transparent community to be established around Eclipse. Today, the Eclipse community consists of individuals and organizations from a cross section of the software industry."

## Doing it manually

You can write your program in a notepad or joe (linux), save it as a file. Then compile it, using a compiler such as gcc. Even debug it in gdb (the

GNU debugger). Correct it in the editor and re-compile the program. Check it again in the debugger and fix the errors again (these are called "issues" in Moft speak). This is called "iteration" and at three o'clock in the morning can lead to madness or a cola dependency.

Most IDE's will compile by spawning a compiler, taking the error output and inputting it, so that the IDE takes you to the first line of errors in your code. Rather like being led by the hand to the spot where you typed the misspelled word...

## Simulators - like flying a plane!

So you wrote your program and compiled it without the compiler whinging. Now you have a hex file that you need to program but you haven't even bought the PIC. What do you do while the shops are shut? Well you can simulate the program's operation on the PIC by using another program that simulates the PIC on your PC.

Download the **gpsim** for Windows or Linux and install it on your PC. Insert the hex program and watch the pins of the PIC go up and down... Or you can add the LCD display simulator and watch that print hieroglyphics, because you forgot the order of the pins and ports...

When you have fixed that and seen your program working for the first time on the simulator, you can go out and buy the PIC. Program it in the sure knowledge that it will work, if you have wired up the prototype correctly...

*More Links...*

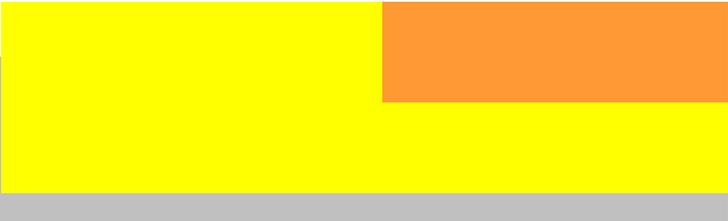
## USB PIC Programmer

<http://sourceforge.net/projects/usbpicprog/>

## David Tait's PIC Archive

<http://www.nomad.ee/PIC/>

*(continued on page 6)*



# Amateur Radio and PICs

(continued from page 5)

## DL4YHF's Software Archive:

### WinPic - A PIC Programmer for Windows

<http://www.qsl.net/dl4yhf/winpicpr.html>

{---

## Hardware

<http://pikdev.free.fr/hardware.html>

## PIC Zone files - downloads

<http://www.ianstedman.co.uk/Downloads/downloads.html>

## So now you want to carve it in stone!

Well not really. You want to program the chip, the PIC that you chose for the function. No its not going to be permanent. Its an EEPROM device that means its an "Electrically Erasable" device. So it can be programmed again and again. It does have a finite life, even so it can be programmed a few hundred times.

Erasing (blanking) the chip's program is simple and is usually done just prior to programming.

A multitude of programmers and associated software are available. The **JDM** and **Tait** programmers probably being the most simple and cheap to construct. Remember my note about the separate power supply.

USB programmers are now becoming more common as the PC's drop the serial and parallel ports from the hardware supplied.

## Examples, we want examples!

Of course you do. So do a Google search on "PIC program examples" and stand back...

Be prepared to do a lot of reading and thinking. There is a lot out there. I generally save the complete web pages as files on my hard

disk in a folder called 'html-files'. These files can be viewed off-line and indexed and searched using the keywords in them.

## More Help with your addiction? Try a support group...

"hdanak -- 01-09-2008, 01:24 PM

you may find these useful:

<http://nixbit.com/cat/text-editingprocessing/ides/pikdev/>

<http://www.micahcarrick.com/04-19-2005/pic-programming-linux.html>

for the first link, I didn't link to the actual project page since this page contains links to other pic IDEs.

if you didn't already know, you could use eclipse with mcl18 fairly easily (i think there is a guide on the wpilib website)

also look at some of the adambots stuff ( i think mentioned above).

good luck. "

## Spin offs

When I started programming microprocessors in the 1980's, there was no help or support from the suppliers/manufacturers or agents. Nowadays its plentiful in the extreme. The application is still the end product of your design.

PICs can be used for just about anything. From flashing L.E.D.s to Martian Rover controllers. I was astounded to see a comparison of circuits between a 555 timer flashing an L.E.D. and a PIC. The PIC is simpler!

[JB 2009-04-12]



## Ham Radio and the PIC

[Updated 8/Apr/99]

The PIC is a great processor for ham radio applications. A really novel application is Mark Sullivan's PIC-based beacon - the PIC itself is used as the transmitter. The (unfortunately no longer available) Perth Radio Experimenters Group prefer the PIC for their projects (a PIC-based frequency counter and 50 MHz synthesizer).

Jeff Otterson's 16C84 based talking repeater controller. PicCon - a hidden radio transmitter controller by Byron Garbrant. GOBZF kits (morse tutor, keyers and beacon controller). Embedded Research claim to have the World's smallest iambic keyer (based on an 8-pin PIC). The PIC is popular with packet radio enthusiasts as controllers for radio modems (TNCs). The Bavarian Packet Radio Group page (or mirror) describes a 9600 bps FSK modem incorporating a 16C84. GMSK Data Products use PICs in their high-speed TNCs. The G-QRP club have featured several PIC-based projects in their magazine SPRAT. In the Winter 1996 edition they claimed a UK first by publishing a 16C74 controlled DDS design by Mick Hodges (Mick was the author of several of the PIC projects and apart from the DDS controller he designed a frequency counter and iambic keyer; Mick now supplies the source code).

The April 1997 edition of the RSGB journal RadCom has an article by Andy Talbot and Lee Wiltshire describing a 16C71 based serial port ADC for a PC - connect a radio and try some DSP! (Andy and Lee have generously supplied full details including several DSP-oriented programs for the PC complete with source code).

The January 1998 issue of QST described Bob Anding's PIC-based morse IDer (the source code is available). Amateur TV (ATV) enthusiasts (and others) might be

interested in Alain Fort's single PIC video generator. The British Amateur Television Club have featured a few PIC projects in their CQ-TV magazine. Some really polished projects for ATV are described by S51KQ. Make sure you have a look at Charlos Potma's page as he has a couple of interesting things based on the 16F84 (50 MHz micro wattmeter and a frequency display) and promises more.

If you are struggling with learning morse code check out the Morse tutor from Brian Jones. Even better, Lawrence Foltzer's article in Ham Radio Online explains how a PIC can read morse for you. K1EL sells iambic keyer chips (programmed 12C5XXs) and provides info on how to "roll your own". TAPR have some projects based on the PIC (currently you can hear RealAudio presentations by TAPR members).

Jacques, VE2EMM describes a "Fox" controller and doppler unit. New Jersey QRP club host the "Ham-PIC" page.

If you want the links, go to the following page.

From:-

<http://www.piclist.com/techref/microchip/piclinks.htm>

**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](mailto:zs6wr.club@gmail.com)****Web page: [www.jbcs.co.za/ham\\_radio](http://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!**

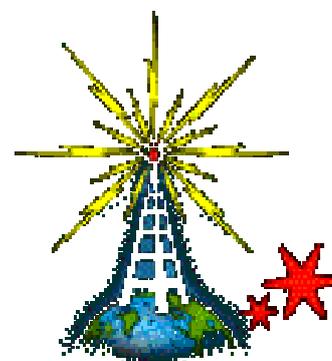
Chairman	Joop Hesp	ZS6C	082 342 3280	<a href="mailto:zs6wr.club@gmail.com">zs6wr.club@gmail.com</a> <b>OR</b> <a href="mailto:joophesp@telkomsa.net">joophesp@telkomsa.net</a>
Vice Chairman	Geoff	ZS6GRL	082 546 5546	<a href="mailto:glevey@gmail.com">glevey@gmail.com</a>
Secretary (Anode)	John Brock	'PieRat'	011 768 1626	<a href="mailto:brockjk@gmail.com">brockjk@gmail.com</a>
Treasurer	Craig Woods	ZS6CRW	082 700 0163	<a href="mailto:craig.woods@absamail.co.za">craig.woods@absamail.co.za</a>
Member	Romeo Nardini	ZS6ARQ	082 552 4440	<a href="mailto:roshelec@global.co.za">roshelec@global.co.za</a>
Member (Repeater)	Phillip	ZS6PVT	083 267 3835	<a href="mailto:phillipvt@sse.co.za">phillipvt@sse.co.za</a>
Member (Technical)	Ron	ZR6RON	082 902 8343	<a href="mailto:ronnie@calidus.co.za">ronnie@calidus.co.za</a>
SARL Liaison	Willem	ZS6WWJ	082 890 6775	<a href="mailto:marie.w@absamail.co.za">marie.w@absamail.co.za</a>

**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](mailto:zs6wr.club@gmail.com)