PC Based Ni-Cad Analyser

By
Ken Hewitt

Introduction

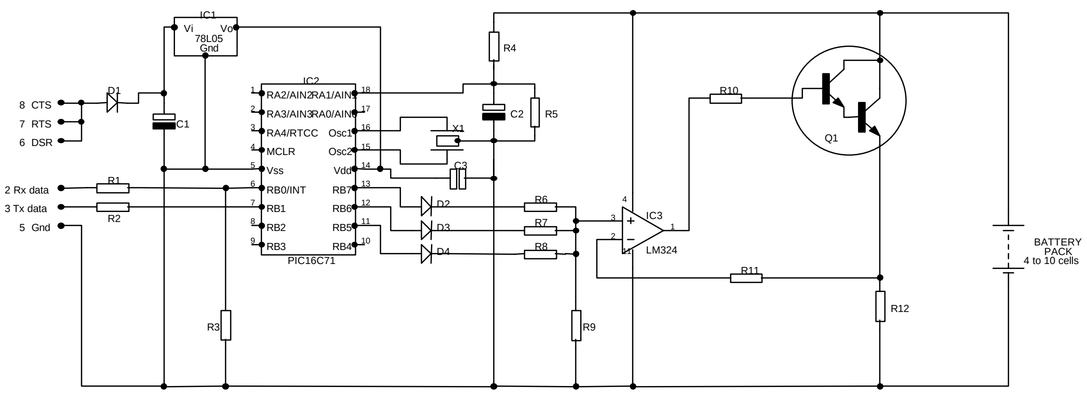
This projected started as a result of an item I saw in an Internet newsgroup, someone asked about a circuit for a simple Ni-cad discharger, one of the people who replied described a unit he had made to work with a PC, this unit was very basic and a bit difficult to set up, but it got me thinking. I looked at designing a similar unit to plug into the printer port of a PC but found that most PC's send data to the printer port during the boot up process and this would mean you would have to keep disconnecting the unit to stop it getting odd commands during boot up.

I decided that it would be better to use the serial port and because of this I could also get data back from the unit (battery voltage), this would make a much more versatile unit. The unit as described here can discharge a battery pack of 4 to 10 cells at 3 different discharge rates.

The PC program that is supplied with this unit is a bit basic as I am not a PC programmer, but someone may like to write a better (win95) driver program for it, if so I will put it on my web site (http://www.welwyn.demon.co.uk) for all to download. The program allows you to select the number of cells in the pack 4 to 10 and which of the 3 discharge currents you wish to use for the test. The test is then started by pressing the CR key, while the test is running it shows the current battery voltage and the measured pack capacity and also the time the test has been running. It also draws a graph of battery voltage against time.

Technical Bit

The discharger is based around a PIC16C71 which reads the commands from the PC serial port and also sends back the current battery pack voltage. It controls the discharge current by setting the value of the constant current load based around IC3 and Q1.



The microcomputer used in the mixer is a PIC16C71, which has a RISC like CPU, and supports 33 instructions. The chip contains everything that is required to form a fully working micro-computer, it has 13 input or output pins, 4 of which can be used for voltage measurement, 1k program memory and 35 bytes of RAM. This may not sound like much but because of the RISC type architecture the resulting code can be very compact. It also has a wide range of power supply limits, 2.5 volts to 6.25 volts at less than 2mA, making it ideal for use in model avionics systems.

Commands

The unit accepts 4 basic commands and when ever it receives a command it will respond with the current measured battery pack voltage. The 4 commands are each one byte long and must be sent without any CR or LF added. The commands are as follows:

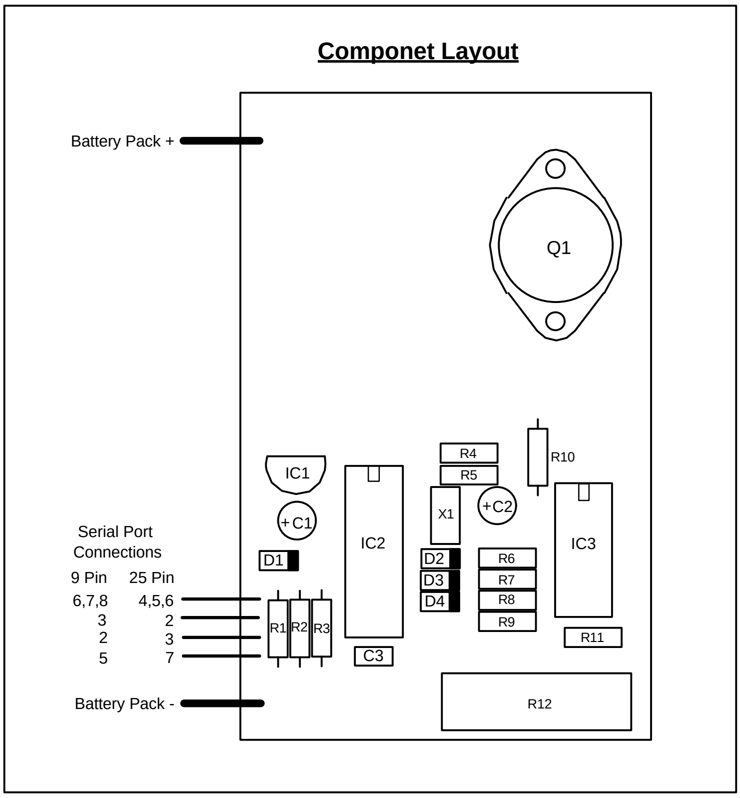
Command meaning	Command
Stop discharge	P
Discharge at low rate	Q
Discharge at medium rate	R
Discharge at High rate	T

The response back from the unit is a single byte (8 bits) which has to be multiplied by 0.060596955 volts to get the current battery pack voltage.

Assembly

There are a few components which must be fitted the correct way around they are as follows: IC1, IC2, IC3, C1, C2, D1-4 and Q1.

The main component IC2 used in this design is a CMOS device and can be damaged by static electricity. When handling this item it is advisable to take some basic precautions, do not wear clothing which builds up a static charge, or handle the item until needed and before you touch it, try to touch a water pipe which should earth any static charge you have built up. DO NOT connect yourself directly to the mains earth.



One of the main things to make sure of in the construction is that the transistor Q1 is mounted on an adequate heatsink, the one in the parts list is the minimum you should use as if you are discharging a 10 cell pack at 1.5 amps then this device is having to dissipate 16.5 watts. The case and heatsink listed in the parts list is suitable for discharge rates of up to 0.5 amps any rates over this then Q1 should be mounted on a larger heatsink. Also if the case and heatsink listed in the parts list are used then be careful as the case is connected to the battery positive terminal not the negative so you could short out the battery by letting the case touch battery negative.

The unit should be house in a case to prevent anything shorting out the underside of the PCB.

If you wish you can build the unit to have any 3 discharge rates you wish just select the value for resistors R6, R7 and R8 from the table and change the discharge current value in the nicad.ini file (see section Setting up nicad.ini file).

Discharge Current	Value	Maplins	Farnell
1.4 amps	33k	M33K	543-743
1 amp	56k	M56K	543-809
700mA	82k	M82K	543-846
500mA	120k	M120K	543-883
400mA	150k	M150K	543-901
200mA	330k	M330K	543-986
85mA	820k	M820K	544-085

Testing

R6 = Low
R7 = Med
R8 = High

There is little to be checked with this design, but one thing that must be checked is that the PIC is getting the correct supply voltage as this is also used as the reference voltage for the analogue to digital conversion. Measure the voltage across pins 5 and 14 of IC 2 it should be 5v $\pm$ 0.05v. If this voltage is wrong then it may be that the PC's serial port is unable to supply the correct voltage, if so the unit can be powered from a 9v battery, just connect it in place of the supply from the serial port feeding diode D1.

There are two error messages you may see when you run the program and press the key to start a discharge test, the first one is 'Interface not found' this could be a couple of things, you are trying to use the wrong Comm port, see the section on setting the ini file or it may be that the supply voltage is too low see previous paragraph.

The other error message is 'Battery not connected' this means the unit can not see any voltage across the battery input connector check that the battery is connected correctly.

Setting up Nicad.ini file

The PC program that is supplied with this unit can be configured by changing settings in the file nicad.ini, this file should be edited with a text editor such as DOS edit or Windows Notepad. The items that can be changed follow a line in the file which starts with REM:. The first two items only have a single entry which can be changed these are :-

Comm port number, this should be COM1, COM2, COM3 or COM4

Then a value which is the volts per cell that the unit will discharge the battery pack to this can be any value you like but should really be in the range 0.9 to 1.1.

The last three sections of the file allow you to alter the information for the 3 discharge rates, these 3 section have 2 items which can be changed the first is the text which will be displayed on the screen when you select a discharge rate. The second item is the discharge current. Once the unit is working you can measure the current for each of the discharge values and input these values into the config.ini file for a more accurate measurement of pack capacity.

Parts List

		Maplin's	Farnell
	Resistors R1 - R11 1/4 watt 1%		
R1	10K	M10K	543-627
R2	470R	M470R	543-305
R3	470K	M470K	544-024
R4	8K2	M8K2	543-603
R5	3K9	M3K9	543-524
R6	330k or See Table	M330K	543-986
R7	120k or See Table	M120K	543-883
R8	33k or See Table	M33K	543-743
R9	10K	M10K	543-627
R10	220R	M220R	543-226
R11	1K	M1K	543-380
R12	0.68R 3watt	W0.68	342-830
C1	10uF 16v	VH06G	920-393
C2	10uF 16v	VH06G	920-393
C3	470nF	RA52G	146-234
X1	4MHz Resonator 3 leg	....	295-346
D1,2,3,4	IN914	QL71N	368-076
Case		YN50E	....
Heat sink		FE38R	………

Q1	MJ3001	QH58N	426-430
IC1	LM78L05	QL26D	412-430
IC3	LM324N	UF26D	400-208
IC2	PIC16C71-04/P inc PC disk	Available from author	
PCB		Available from author	