Random Number Generation with a Simple Transistor Junction Noise Source

(Version 1 Revision 2.   22 June 2004)

Rationale

There is a perception amongst computer users that collecting true random data from an external source is difficult and/or expensive. As a result most people interested in, or with a need for this type of data make do with pseudo-random data, collect random data by sub optimal means, or rely on passwords and phrases that may not provide enough security against a determined attacker. Until about 1995 this difficulty with collecting random data was true, but now that a PC comes standard with a sound card, the PC includes the ability to easily sample external signals and this earlier position is now false. Through the sound card it is easy to collect true random data from an external source.

The intent here is to show that a source of true randomness i.e. a transistor junction noise is easy and inexpensive to construct and that data collection is equally simple through a PC sound-card.

This simple design takes into account (as much as is possible) the needs of those with limited electronic construction experience and those with few funds. It has to be acknowledged that every unnecessary component or requirement for a specific part reduces the number of people able to begin the project and also reduces the numbers who will successfully complete it.

This device is not what I would design for commercial use; these devices have to be fully self-contained, fully functional and idiot proof straight out of the box. This level of functionality means expense. The design presented here is for those with a basic understanding of their requirements and how to achieve them.

Revision 0 received unfounded criticism because of the unusual design, which was motivated by the need to extract maximum output from the minimum components. The normal design "rules of thumb" were bent with purpose and calculation to minimise component count and maximise reliability. I have expanded the design rationale to cover alternate designs and also the final design in more detail. This circuit has been independently constructed and verified to work with a wide variety of components, "as advertised". If you have the data CD you will see that the processed output from this device passes muster with all tests for randomness. You can test it yourself.

My tertiary qualifications and experience is in electronics. I am good at it. I know what I am writing about and I know how to design. Please read and consider before trashing. If you still don't like this design, put your design in the public domain so it can be evaluated.

eather@tpg.com.au

On The Cover

The picture is one of the author's white noise sources assembled to prove the concepts of this design and provide random data for testing. The white glow from underneath the unit is from two, high intensity white LED's shining on glossy photo paper. The excessive illumination serves no purpose other than to provide an indication that the device is on – and it looks really cool. The two LED's consume 5 times more power than the noise sources. A more rational arrangement is specified in this paper.

Abstract
The purpose of this project is the construction and description of a practical method of distilling large amounts of true randomness from transistor junction noise. Such a project needs to be both simple and inexpensive to be within the capabilities of anyone with modest electronic construction or financial abilities.

The project described meets those goals. It is similar (or with careful set-up, superior) in speed compared to commercial units that interface to serial ports but with only a few percent of the price tag. It contains no critical, special or difficult to find parts and it can be constructed in a poorly equipped workshop.

Anyone with electronic construction abilities can build this project themselves. Those with financial abilities will be able to find someone who can construct it for them. Finally a novice should be able to construct this project providing they take time and care plus have some access to a more skilled constructor – things can go wrong and a novice won't have the skills to trouble shoot.

This article lays out the project hardware.

You will have to provide your own software or use a combination of commercial and custom software. At the least you will need a sound recording program and a hashing function or method to isolate the bits in the captured data file. The software section gives some ideas on ways to proceed and some of the pitfalls.

To do: I would like to add nice software.

Intelligent comments, suggestions, and tested improvements are always welcomed. The objective is to increase knowledge and reasoned debate is always worthwhile. Comments made by people who haven't bothered to read the paper, attempted to understand it, or tested the circuits under consideration are just …

Index

Abstract 3

Hardware Construction Summary 5
- Schematic Overview
- Transistor Junction Noise Generators
- Power Supply
- Parts List

Construction 7

Testing 10
- Adding More Gain

Hardware Design Rational 12
- The Hardware Interface

Noise Generating Hardware 13

The Rejected Circuits 15
- The Basic Circuit
- Suggested Modification

The Final Design 19
- Design Procedure
- Temperature Effects

Adding A Regulated Power Supply 26

Software Summary - Traps and Pitfalls 27
- 8-bit Mode
- 16-Bit Mode
- One More Trap

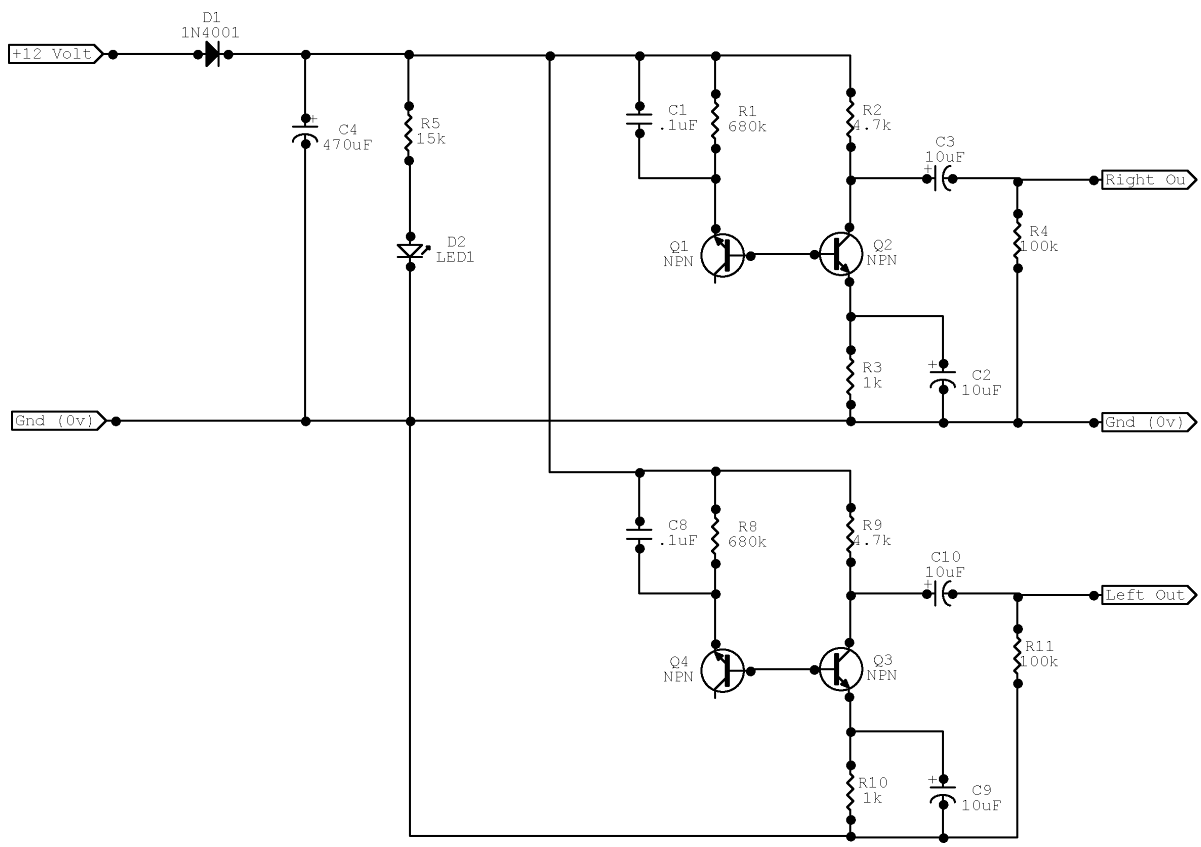
Appendix 29
- Typical Transistor Data Sheet
- Additional Data Sheets
- Component Identification

Hardware Construction Summary

Below is the schematic of a dual channel, transistor noise source suitable for interfacing with a PC's sound-card "line input" interface. With proper computer post-processing, this circuit produces high quality true random data. Intended applications include key generation, rapid refreshing of the random data pool used by programs such as PGP and dll's such as EBcrypt, and for use in computer simulations. For those so inspired the data generated is also suitable for constructing one-time-pads or Maurer type stream ciphers[1].

The use of the sound-card as the data collection device and the computer for post-processing makes this unit effective while still being simple and cost effective in the extreme.

Schematic Overview



Transistor Junction Noise Generators

This is not a complicated circuit. To the novice it may look complicated because it is the complete circuit but it logically breaks into just three sections and two of those are identical. Don't be put off. On the right hand side of the schematic are the two identical circuits: one above the other. Each of these is a noise generator. $R_1 - R_4$, $C_1 - C_3$ and $Q_1 - Q_2$ form the top

[1] Be sure to read Security Weaknesses in Maurer-Like Randomized Stream Ciphers by Ferguson, Schneier and Wagner

generator and $R_8 – R_{11}$, $C_8 – C_{10}$ and $Q_3 – Q_4$ form the bottom generator. The unusual connection of Q_1 (and Q_4) is quite correct. The "reverse voltage" applied to the emitter - base junction is the source of the transistor junction noise. The collector of Q_1 (and Q_4) remains unconnected.

Power Supply

As this project draws approximately three milliamps, battery power is entirely feasible. A battery pack that holds eight AA (LR6) type alkaline batteries will last approximately 300 hours, produce 30Gbyte of distilled, and FIPS 140 tested random bytes. A battery power supply also means there will be no problems with power supply ripple or mains hum. If you require or prefer mains power you can use either a regulated 12v plug pack or build the necessary voltage regulator as detailed latter. Back to the schematic…

The four components on the left of the schematic designated D_1, C_4, R_5 and D_2 constitute what could be called the power supply interface. Diode D_1 prevents any damage should the wrong polarity power supply be plugged in. Unlike science fiction, applying reverse polarity rarely makes electronics run backwards. Reverse polarity usually destroys them.

D_2, R_5 form a simple reminder light that the device is powered and is essential for battery power. If desired you can leave them out if you only ever intend mains power.

Parts List

None of these part's values are especially critical and if required the value next closest to the one specified can generally be substituted with no undesirable change in performance. For example, R_2 is specified as a 4.7k ohm ¼-watt resistor; any value from 3.9k ohm to 5.6k ohm will work as will any power rating from an eighth of a watt upwards. Resistors rated at ¼ watt are ideal; they have plenty of power handling capacity, enough size to handle during construction and are the most common and cheapest.

Transistors are listed in rough order of suitability but most NPN type of small signal transistors will work; see the appendix for a typical data sheet with the important specifications highlighted. Data sheets of the other recommended transistors are also included so you can check the pin connections for your transistors. European and American transistors tend to use different pin connections, so check the data sheets.

Some transistors have a letter suffix after their part number. This letter signifies a product with superior gain or tighter specifications, which is of some importance in this project. Thus, a BC549B is preferred over a BC549A, which is preferred over a BC549. A 2N2222 with no letter suffix will work if there is no alternative but is not otherwise a good choice in this design.

Components

C_1, C_8	0.1µf, 25 volt (minimum) ceramic or polyester type (µf =micro-farad or 10^{-6} farad). With a low gain transistor such as a 2N2222 this may need to be increased to 0.22µf.
C_2, C_9	10 µf, 25 volt (minimum) electrolytic or tantalum type
C_3, C_{10}	10 µf, 25 volt (minimum) electrolytic or tantalum type

C_4	10 - 470 μf, 25 volt (minimum) electrolytic. Note that on the schematic diagram this is labelled as a 470 μf capacitor but almost any value will do.
D_1	1N4001 or any other diode from the 1N400X range or any other diode with a maximum forward current of more than 500 milliamperes
D_2	any high efficiency LED – size and colour is unimportant
Q_1, Q_3	any small signal NPN type with emitter to base reverse breakdown voltage (V_{ebo}) specified as 6 volts or less – most small signal transistors fall into this category. (BC549, BC547, BC548, PN100, 2N3904, 2N2222A)
Q_2, Q_4	any small signal NPN type ideally with current gain (called h_{FE} or β) specified as over 100 and ideally less than 500 – again most small signal transistors fall into this category. (BC549, BC547, BC548, PN100, 2N3904, 2N2222A)
R_1, R_8	680k, ¼ watt
R_2, R_9	4.7k, ¼ watt
R_3, R_{10}	1k, ¼ watt
R_4, R_{11}	100k, ¼ watt
R_5	15k ¼ watt

Miscellaneous

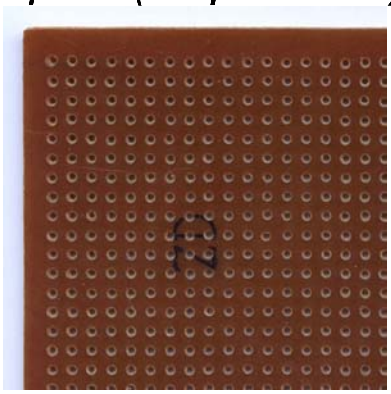
2.5 millimetre DC power chassis mounted socket
2.5 millimetre DC power line plug. You can use any plug and socket type as long as it is reliable, you have a matching pair and it is different from the 3.5 mm plugs and sockets used with the sound card.
3.5 millimetre chassis mounted mini phono stereo socket
"Vero Board" or strip board
Plastic case – You need to put it in some sort of case for any type of reliablity
Solder, drills, screwdriver, miniature needle nose pliers, miniature side cutters, tinned wire, hook-up wire, de-solder braid, soldering iron, wet sponge or paper towel, time, patience.

Construction

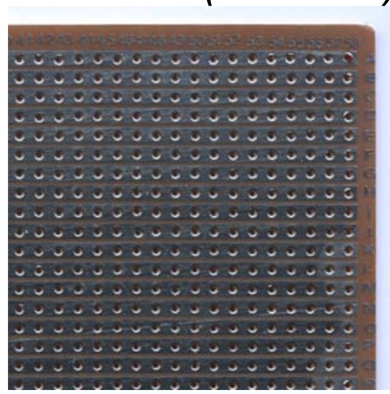
As already hinted in the parts list "Vero Board" or another type of "strip board" is the chosen construction method. This is because the result is durable, even if it lacks elegance, and mistakes are reversible. If you have construction experience you should use the construction method you are most comfortable with. A custom designed printed circuit board (PCB) would be best, but it is hard to imagine everyone who wanted a few random numbers investing the time and equipment required to produce a PCB. Strip board was chosen because a circuit that demands a PCB would be a barrier to many who want a random number source.

Below is a picture of a small section of strip board. Note that all the copper strips on the underside run left to right. The grid numbering on the bottom side is a sales feature and not of any help at all. Attempts to use a numbered grid simply result in premature insanity. The side without the coper strips is called the component side.

Top Side (component side)



Bottom Side (solder side)



Your layout will probably differ from mine and there is no need to slavishly copy. It is easier to try and line up all major components in one direction (either horizontally or vertically) and have the copper strips running perpendicular to the components. Use resistors and wire links on the component side of the board to complete the circuit. Remember that each electrical connection runs the full length of the strip unless you cut it. If you need to cut tracks on strip board you do it by hand twisting using a ¼" drill bit. Ensure the break is complete and clean by removing any loose bits of swarf. Allow more strip board than you need. Excess size can be tided up later but a too small board can't.

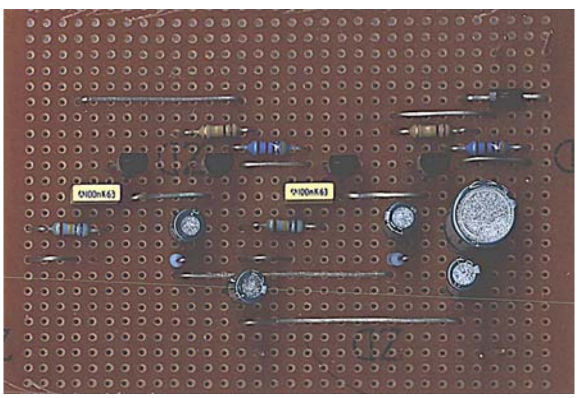


Figure 1 – A Completed Dual Channel Noise Source

(This board is rotated relative to the previous photographs – in this example the copper strips run top to bottom on the underside of the board.)

This is a completed dual noise source shown as an example layout using BC 549 transistors (minus the LED and associated resistor). In this layout and with these transistors the transistors' collectors are all closest to the right hand side of the photograph. This board is larger than necessary to allow two identical circuits to be laid out next to each other. This makes it easier to find layout mistakes, but note that the large electrolytic capacitor and diode on the right hand side. These two components are from the power supply and are not repeated like the other components.

With this layout there are also no cuts in the tracks. Incompletely cut tracks and stray copper whiskers can be a problem with strip board and inexperienced constructors.

Soldering is the most important skill. You have about ten seconds to complete each solder joint but with a little practice you will be doing this in two or three seconds. The need is that you are neither spooked into unnecessary rushing or that you dawdle too long. Counting the seconds as you solder may help. If you are taking near ten seconds each joint then you should practice on a scrap piece of board until you can reliably do it in less time. The reason for the fuss is that much too much heat can damage the semiconductors, electrolytic capacitors and the strip board.

Place the tip of the soldering iron at the junction of the strip and component. Wait one or two seconds and then apply the solder to the junction of the board and component preferably opposite the soldering iron. Use only a small amount of solder, about 4 mm of the most common solder size.

With a good solder joint the solder has "wet" both the component lead and the strip board. It will look a bit like a smooth sided "volcano" without a crater at the top. Assuming your board and components are clean, then if your solder joint looks like a "crater" then try more heat on the component lead or if it looks like an "onion" then the board needs a bit more heat. Craters and onions are signs of bad joints that need redoing.

If you don't have any experience in electronic assembly, ensure you can access someone with construction experience if needed. Learning everything while building and trouble shooting can be a frustrating task.

The circuit board should be installed into a case fitted with a correctly wired power socket. If the wrong polarity power is plugged into this socket the unit will not work, however the power diode D_1 prevents any damage. The case should also be fitted with a correctly wired stereo mini phono (3.5mm) socket. It is not important which noise source you send to the left or right channel but the ground or 0-volt connection must be correct. The picture below will help.

Knowing the connections for the phono plug will also help if you have the situation where only one speaker is producing sound. On the plug shown the metal sections from left to right are; the ground connection (0 volts), the right sound channel, and the tip is the left sound channel.

There are also two construction tips worth knowing about. If you do not have a good PCB vice to hold the board when you are soldering, a heavy, wide mouthed jar or bottle makes a much better work area than a bench. After a few components have been placed they will butt against the rim of the jar and stop the board sliding about. Also, never clean a soldering iron tip before you put it down. Put the iron down dirty and with excess solder on it. This protects the bit from oxidising: clean the tip just before making a solder joint. A dry, thick, paper towel is better for cleaning than a wet sponge. Lastly, don't flick excess solder off the end of your soldering iron; one mistake could cost you an eye, and you only ever get two of them. (OK it was four tips)

Testing

After construction, connect a battery pack. The LED should light immediately. If not, disconnect the power and check everything. Make sure you also check for accidental shorts on the strip board. If any component has smoked, burnt or is hot to touch, you should carefully examine for mistakes in the layout and component orientation. In particular check all the circuitry that is physically and electrically close to the suspect component; something made it get hot. The most likely culprit is an accidental short or incorrectly placed link or component. Smoked or burnt components should be replaced.

When you are satisfied, plug an inexpensive pair of **powered** computer speakers into the output socket. This unit will not produce any noise from un-powered speakers. Adjust the volume. You should hear a continuous hissing sound something like a strong wind, the ocean, or perhaps the sound of being inside an airliner. If there is no sound, unplug and check your work. If there is sound from only one speaker find which noise generator is responsible for that speaker and compare that circuit to the other nonworking circuit. Electronics is not magic. If it is not working, there is something wrong. Your task is to find it. This calls for logic, patience and a methodical approach. It is much more likely that something is wrong with your circuit layout than a faulty component. Check carefully before replacing components.

When all is well, it is time to plug the noise generator into the computer sound-card's "line in" socket. Start you favourite sound recording program and record a sound sample. For best results set to record at a sample rate of 44100 kHz. Adjust the record level so the sound is as loud as possible without clipping. A program with a real time "VU" meter is essential. If you don't have such a program I suggest Cool Edit 96. It is shareware / functionally-challenged-ware[2] but is still capable of everything needed for data collection and does not time-expire. If you make significant use of Cool Edit you might like to purchase one of the updated versions.

Adding More Gain

Sound cards vary considerably in terms of the gain they provide. If you cannot set your system to record at a reasonable level there are some "tweaks" you can try to improve the situation. With Cool Edit 96 "a reasonable level" might be described as –18db or better. Each 6db represents 1-bit. Thus -18db represents collecting only five data bits from a possible eight with an 8-bit sample. If you can't collect this many bits than it is worth trying to tweak your noise generator. If a tweak does not improve the result, remove it and try the next tweak. This circuit is designed to use a wide variety of components but if your sound card is picky tweaking may help. Do any alterations with the unit unplugged from the computer and un-powered.

You can also try increasing the value of the resistors R2 and R9 from the original 4.7k to 6.8k. This will give you a gain increase of 3db or ½ a bit, enough to make a noticeable improvement and perhaps enough for good results.

Replacing R_1 and R_8 with a 560k resistor may provide more noise. If you have low gain transistors, it might be possible to go down to a 470k value.

[2] Functionally-challenged-ware – the politically correct way to say "crippleware"

If it is feasible for you try increasing the voltage form 12 volts to 15 volts. This increases the noise signal generated by Q1and Q4.

Last, you might have a problem caused by an unusually high gain transistor biasing the circuit outside of its linear range. This will be quite rare so double check everything else first. In this case the resistors R_1 and R_8 should be replaced with 820k ohm resistors.

If these four tweaks fail to bring the desired results then try a different brand of sound-card. The gain provided can vary by as much as 20 db.

I have built about a dozen of these circuits using almost any components and values that came to hand. All of them worked without need for adjustment. Nothing is perfect but this circuit is tolerant of mismatched components and sloppy layout so you shouldn't have trouble.

That is it for the hardware. It is simple by design and necessity. Correct functioning is achieved by users being aware of the need to post-process the captured data signals. It is of course possible to design a USB interfaced source of true random numbers, with microprocessor control and a RAM based pool of entropy to provide a rapid response capability, but how many average people could build this?

The rest of the random number generation is in software by post-processing the sound samples collected on the host PC. Post-processing holds a number of traps for the unwary.

Hardware Design Rational

This unit costs about $30 and collects random data faster than some commercial systems costing several hundreds of dollars. It is an "odd" design and so it is not unreasonable to ask for the design rational. Judging from the questions so far, some also want to know, "why didn't you do it this way"?

The objective was to produce a true source of entropy that could be built by anyone with some electronic experience, was inexpensive, and required no special components. This is a restrictive design requirement.

The Hardware Interface

The most fundamental design question is how to present the data stream to the computer.

- A custom built PCI card is excluded on the basis of cost, the skill level required and the availability of parts

- External interfacing to a SCSI card is also excluded on the basis of cost, skill and availability

This leaves the built-in interfaces on the PC. USB, parallel port, serial port, keyboard, and sound-card (both the microphone and line inputs could be considered).

- USB would be possible, but requires a custom chip for the interface. These require a high level of construction skill and a printed circuit board (PCB). The "must have" requirement for a printed circuit board is unacceptable. If the PCB were optional that would be OK, but with a surface mounted quad flat pack IC with pins 0.05 inch apart it is not for the inexperienced.

- The parallel port is usable, but is easily damaged. To get good speed requires the use of all eight data input pins. This implies a large and relatively complex circuit. I think it would be too difficult, with too much risk of expensive damage.

- The serial (RS-232) port is much more robust. Collecting and converting data bits to RS-232 signals requires a project with a few integrated circuits. A PCB would not be a strict requirement but would greatly improve the successful completion rate. The serial port is somewhat slow even at 115k baud. Some commercial units use this interface but we could make considerable cost savings by simplifying the data collection and shifting data post-processing onto the host PC. The serial port would be a possibility, and if the data requirements are low, there is a very simple way to collect random data through this port.

- The keyboard is no good. It combines the worst of everything. It is fragile. It is very slow. It interferes with the normal use of the computer. No.

- The sound-card (or built in sound-chip) is the most interesting of possibilities. The inputs are surprisingly robust. The analogue filters that give this robustness cause some problems, but they are solvable. The simplest noise sources are analogue and the sound-card does the analogue to digital conversion for free. A sound-card has the potential to grab large amounts of data. Collecting single bits from the microphone input, (not from **a** microphone) will give five kilobytes of random data per second. From the two channel line inputs, in 16 bit mode, perhaps 180 kilobytes per second (try getting that by wiggling your mouse. It is very hard on the wrist!).

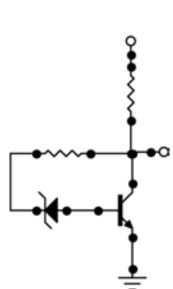
The interface of choice is the sound-card, with a simple analogue noise source. In this design option the sound-card does the analogue to digital conversion and host PC performs additional post processing. This is very inexpensive.

As a fall back position there is the serial port and two distinctly different generators, either a full serial interface circuit to collect up to 11 k Bytes per second (115k baud) or a simpler method of collecting 135 (perhaps more) bytes per minute[3].

[3] This would take the form of a properly conditioned random noise source fed to the serial port Clear to Send (CTS) and/or Data Set Ready (DSR) pins (pins 8 and 6 respectively on the DB 9 plug. Using the system timer interrupts, (18 times per second in Windows) call a small program that interrogates the status of the serial port. The status of the port provides 1's and 0's. A most useful feature is the virtually zero impact on system performance.

Noise Generating Hardware

Several circuits were considered for this project. The needs are, simplicity, repeatability and low cost. A sound card has a single channel “microphone input” and a dual channel “line input”. It is an easy decision to use the line inputs – two inputs allow doubling the data collection rate. The final circuit is unusual and is as close as possible to getting “a silk purse from a sow’s ear” as is possible – with just 2 transistors it achieves performance that would normally require 3 to 5 transistors or an op-amp integrated circuit. It is worth going over why other “prettier” circuits were rejected.



This is the most basic junction noise source of all. The zener diode used this way is inherently noisy. Unfortunately, semiconductor manufactures have worked hard to reduce this noise. For a noise source the result was pitifully quite.

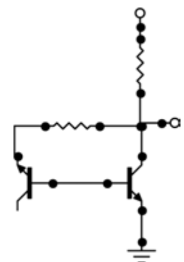


Figure 2

A transistor junction replaces the zener diode. This is much noisier but still lacks enough output to use the sound-card’s “line inputs” - essential for maximum speed. The design is also sensitive to board layout and will sometimes oscillate for no apparent reason. These two strikes are enough to disqualify the circuit.

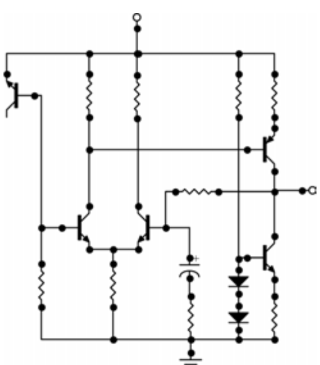


Figure 3

This circuit is pinched straight from an audio amplifier I designed for a magazine article (it’s also common to a lot of other amplifiers too – oops)

As expected the performance was brilliant, good output level, frequency response and gain adjustable too! But it is just too complex for the job required.

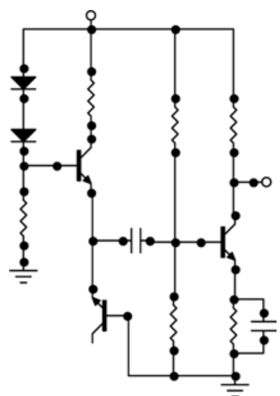


Figure 4

This was a step backwards from the previous design. It is still too complex for the task and was rejected before reaching a prototype stage.

Of passing interest – the configuration for the output transistor is the most common of all arrangements.

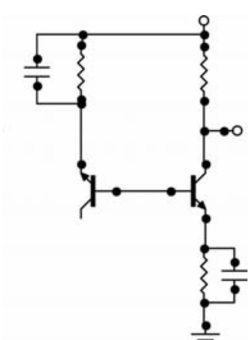


Figure 5

Re-examining circuit 2 and searching for more gain lead to this. DC bias of the noise transistor is low, but the bias resistor is bypassed by a capacitor meaning there is little resistance to the AC signal of the transistor junction noise so there is heaps of noise being feed into the second transistor (the output transistor). The small amount of DC feedback on the output transistor is also bypassed by a capacitor. Those 2 capacitors add up to an additional 30db of gain. Frequency response is surprisingly good. With a minimum of components, it is simple to build. Low amounts of feedback also means the circuit does not oscillate and is tolerant of different construction methods.

The Rejected Circuits

All these circuits have been assembled on the same prototype boards, tested with the same power supply, the same transistors and the same software settings as the final circuit i.e. the tests were as fair as possible.

The suggestions to the first post centred around figure 2 or variations of it. The basic circuit has several nice characteristics but has two down sides - it simply does not have enough gain and it has a tendency to oscillate. It has to be used with either the sound-card microphone input thus limiting data collection to a single channel or if used in the "line in" sockets then only a couple of bits are collectable.   Neither option is desirable, and there is the nasty instability problem waiting to raise its head.

The Basic Circuit – Figure 2

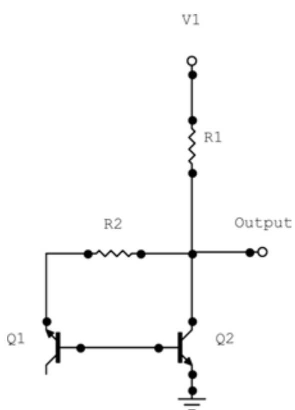


Figure 2

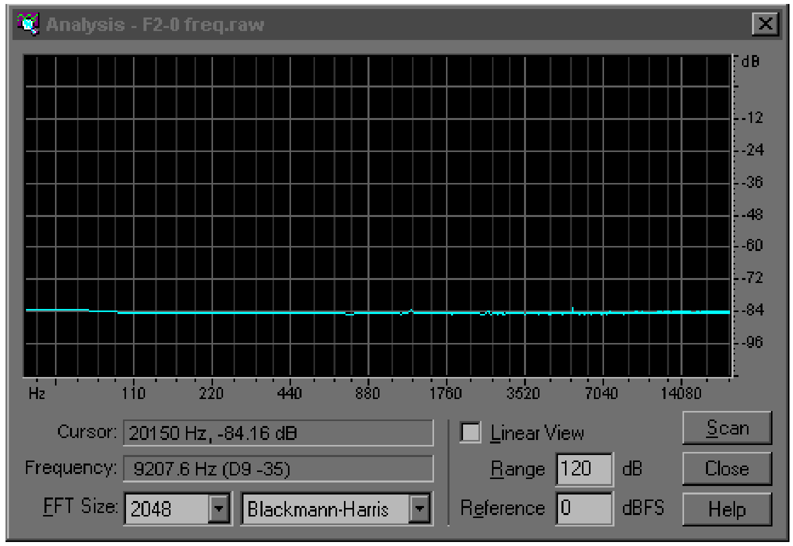
The most used commercial circuit for transistor junction noise generation is that shown to the left, figure 2. This circuit is commonly an audio noise source and is further amplified to produce a "line level[4]" output of white and pink noise. It is popular because it uses only four components. With a carefully designed PCB layout this circuit produces no problems. The design needed for this project must not require a PCB and must work with a wide range of transistors and layouts. This circuit does not offer that.

The final circuit is specified with R_1 as 4.7k, R_2 as 470k and V_{cc} as 12 volts. A sound sample recorded with Cool Edit produced the following data:

[4] "line level" currently has at least 3 different definitions – 100 mV into 10k for domestic equipment, 775mV into 600 ohms for professional audio equipment and 2 volts into 10k for CD players and "digital" equipment.

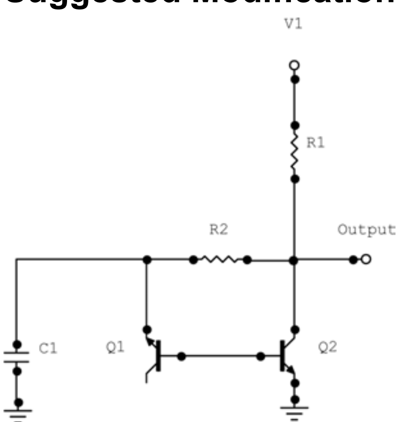
Min Sample Value:	-238
Max Sample Value:	175
Peak Amplitude:	-42.8 dB
Possibly Clipped:	0
DC Offset:	.002
Minimum RMS Power:	-72.17 dB
Maximum RMS Power:	-46.72 dB
Average RMS Power:	-56.65 dB
Total RMS Power:	-55.9 dB

Using RMS Window of .2 ms

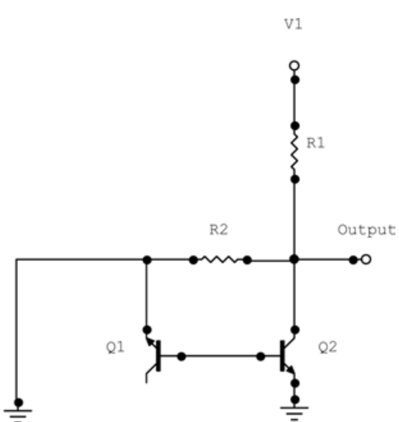


The output is linear but there is a problem with the lack of dynamic range. Note that the minimum sample is –238 and the maximum 175. This is very poor and represents a maximum sample of 8 bits of data from a possible 16. The amount entropy per sample will be way below 8-bits.

Suggested Modification



This was the most suggested and applauded modification. The difference of this circuit to the basic figure 2 circuit is the capacitor C_1 across the noise source. A simple DC/AC analysis shows the probable effects. For the DC analysis capacitors are considered as open circuits because at DC they have infinite impedance. In this case the DC circuit is equivalent to the basic figure 2 circuit so as expected it biases up nicely.

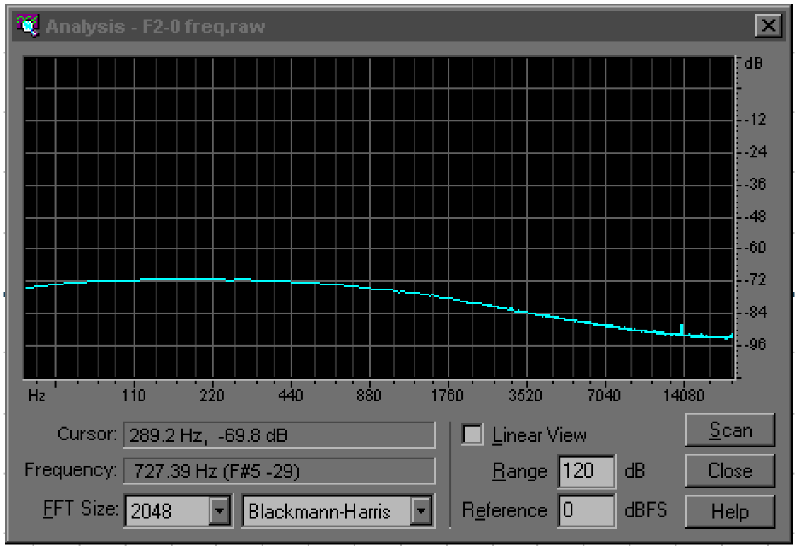


The AC analysis replaces capacitors with short circuits because the impedance of the capacitor reduces as frequency increases. The effect of this capacitor can be seen to short out the noise-generating transistor. At some point the reducing impedance of C_1 will load down the base input of Q_1. With C_1 set to 10µf the following results were produced.

(Inductors become short circuits for DC analysis and open circuits for the AC analysis – but there are no inductors in any of these circuits. It is mentioned only for your edification and to help show that I am not trying to pull the wool over your eyes)

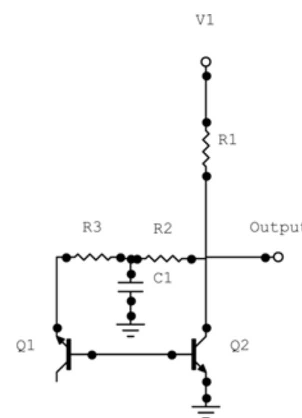
Min Sample Value:	-199
Max Sample Value:	228
Peak Amplitude:	-43.17 dB
Possibly Clipped:	0
DC Offset:	.001
Minimum RMS Power:	-78.38 dB
Maximum RMS Power:	-42.93 dB
Average RMS Power:	-56.08 dB
Total RMS Power:	-54.71 dB

Using RMS Window of .2 ms



As you can see the frequency response is not very linear and in particular it cuts out the higher frequencies. There is 12db of boost to the bass signal but linearity is one of the first requirements for this amplifier. A non-linear response from the amplifier reduces the amount of entropy in the output signal.

The circuit above could be improved by splitting R_2 in to 2 resistors each of half the value of the original R_2 and placing C_1 between the junction of the two new resistors and ground. This results in a flat frequency response with an improved gain of about 12db over the entire frequency range.



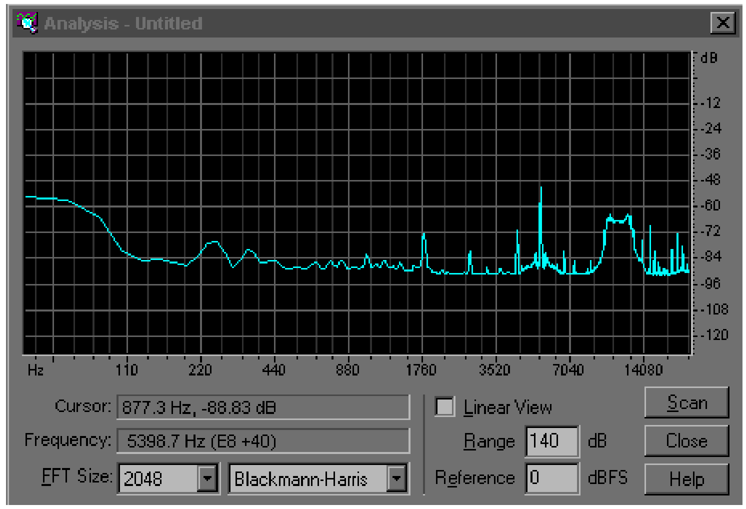
This is better, and is easily the best of this group – it provides a sample range of about 10 bits but this is still inferior to the results achieved by the final circuit. To make this circuit perform as well as the alternative circuit would require either another transistor stage for amplification or an integrated circuit and PCB. The extra parts cannot be justified.

This circuit (figure 2) and the derivations of it have been thoroughly investigated, they are simply not as good as the selected design for this application and there is potential for unexpected oscillation.

With the figure 2 circuit small alterations in layout can send it into oscillation. Without a custom designed PCB and/or restrictions on the transistor's specifications there is no guarantee that this circuit can be successfully completed. This ruins the circuit for very general construction needs. If you have some experience you may be able to make it work, or you might be lucky and just "jag" a working circuit, but not everyone will have such skill or luck. The problems relate to parasitic capacitance, inductance and cross talk and is compounded by the need to make the circuit layout and construction as general as possible.

Believe it or not, this frequency response is from the basic figure 2 circuit. It is the same circuit built with the same components on the same prototype board as figure 2. The only difference is the circuit **layout** was slightly changed. As a result it displays almost every unwanted characteristic possible – squawks, cheeps, burps and whistles.

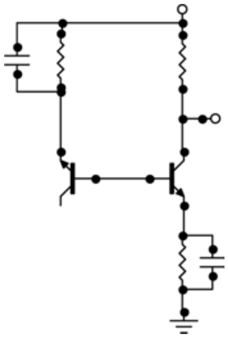
This data looks like an instantiations sample but is in fact an analysis of a 15 second sound sample!



The low frequency hump is not from mains hum – it is the circuit's own oscillations.

In the earlier graph of the suggested modified circuit there is a tiny spike at about 14 kilohertz, this shows how close even a "good" circuit is to instability.

The Final Design

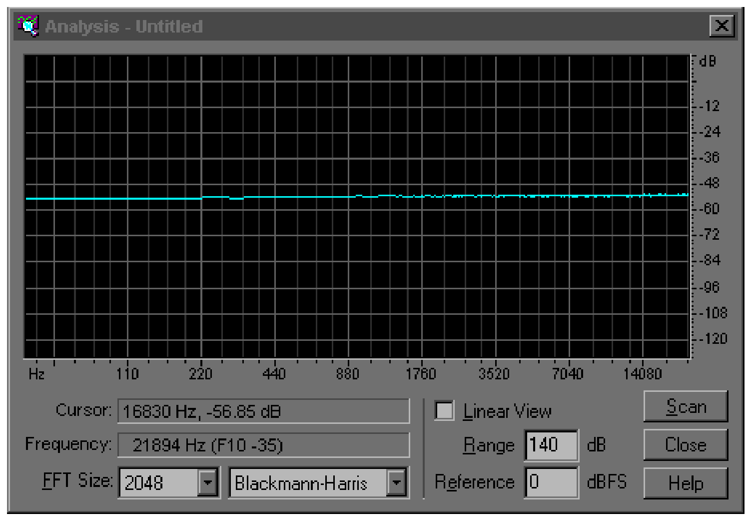


The final design works well with a wide range of transistors and other components, providing stable, high level outputs.

As you can see from the graph below, the frequency response is linear and the range of the sampled signals is much better (-6377 to 6722 compared to –238 to 175). This represents a possible 13 bits out of 16 compared to figure 2's result of a possible 8 out of 16 bits, or the second modification giving 10 out of 16 bits. This circuit is a major improvement.

Min Sample Value:	-6377
Max Sample Value:	6722
Peak Amplitude:	-13.76 dB
Possibly Clipped:	0
DC Offset:	.007
Minimum RMS Power:	-39.59 dB
Maximum RMS Power:	-17.6 dB
Average RMS Power:	-25.01 dB
Total RMS Power:	-24.32 dB

Using RMS Window of .2 ms



The reason for most of this improvement, is the capacitor at the top left of the circuit. If you do the quick AC analysis as was done to the previous circuits, the capacitor will be replaced with a short circuit. In this case this capacitor provides more AC bias current for the noise transistor, but blocks any runaway DC current flow. This "ugly" biasing circuit was selected for this application it works better and with fewer components than the other arrangements.

Final Design – Design Procedure

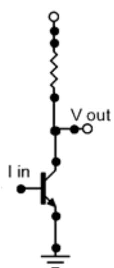
As already stated the design requirements were for a simple noise source that was cheap, easily constructed and required no special components. These are quite restricting to the design choices.

Battery Pack

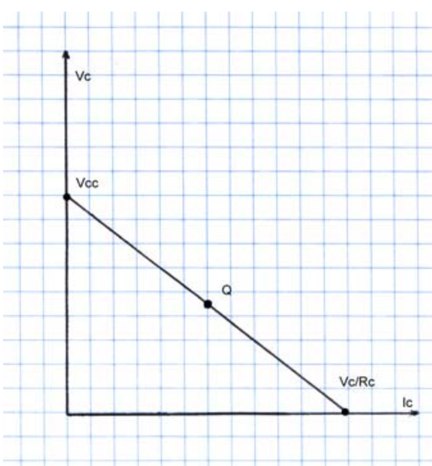
A battery pack was chosen because it is free of mains ripple and it will not pick up stray hum signals. For reasonable life current consumption must be kept down. If the total current is kept below 3-4 milliamps battery life for good quality AA (LR-6) alkaline batteries will be somewhere between 200 and 300 hours. I think this is quite acceptable.

Output Transistor Biasing

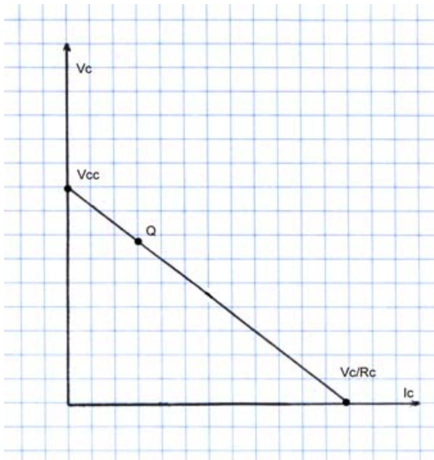
Load line graphing will help explain the major point of contention in this design.



Consider the circuit to the left. When the transistor is turned completely off the voltage at V_{out} equals the supply voltage V_{cc}. When the transistor is turned hard on V_{out} approximately equals 0 volts. The current that flows when the transistor is turned hard on is simply $Ic = \frac{Vcc}{Rc}$ (form Ohm's law). From these two points it is possible to draw a "transistor load line graph" which is shown below. Graphing makes the design process easier to understand and explain even though all the necessary calculations can be done without reference to any graphs.



This graph shows the load line of the simple transistor amplifier. The "Q" point (Q for quiescent) is **normally** selected so that without any signal input the circuit biases up to ½ of the supply voltage as shown here (in our case with a 12 volt V_{cc} the "Q" point would be expected to be 6 volts. This allows the amplifier to produce the widest range of output voltage swings without clipping. This "normal" selection is only a "rule of thumb" and not written in stone. In fact, the circuit in figure 2 (reproduced below) **cannot** operate at this "optimally" selected Q point because the noise source requires an additional 6 volts to operate.



This is the transistor load line graph for the circuit of figure 2. The Q point is approximately 9 volts rather than the "optimal" 6 volts suggested earlier.

What has happened? Rather than any strange break with the physical laws of the universe all that has happened is that the 6 volts required by the noise source has been accounted for and the remaining 6 volts has been divided in half by figure 2's biasing circuit. That is 3 volts plus the 6 volts for the noise source means the Q point is now at 9 volts.

The maximum possible output voltage swing is only half of the previous graph (+/- 3 volts rather than +/- 6 volts), however the circuit still works. That should be an interesting question, "why does this oddly biased transistor amplifier still work even though the maximum output voltage swing is halved compared to the "properly" based design?" Think for a minute before reading on.

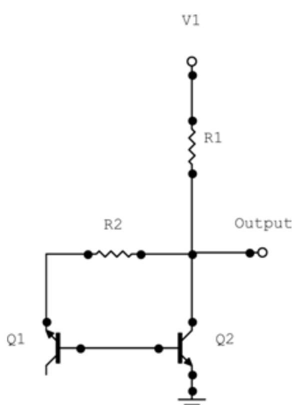
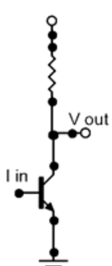


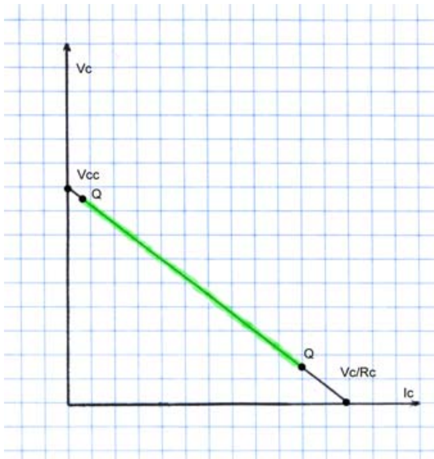
figure 2

The reason this circuit still works is because the amount of voltage swing required from this circuit is very low, about 100 millivolts or 0.1 of a volt and so the circuit works even though it is not "optimally" biased.



Consider again the simple circuit on the left and extend the previous line of thought further.

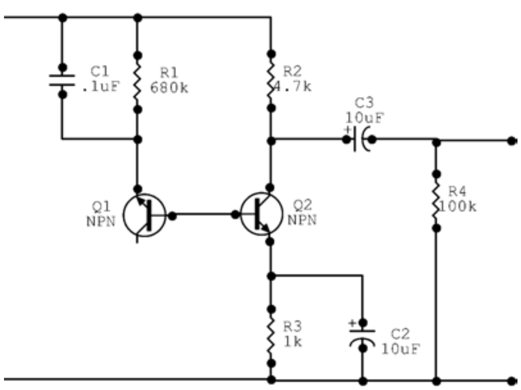
This circuit will work if the bias point (the "Q" point) is set anywhere from Vcc – 50 millivolts to 0 volts plus 50 millivolts and an additional two volts or so to keep the transistor operating in a linear mode.



This is the transistor load line showing the possible range of Q where this amplifier circuit will still work. The range is large because the total required output voltage swing is only about 100 millivolts.

This large range of workable quiescent currents provides extra latitude in the design stage. This latitude will be used to make a simple design tolerant to component changes.

As you would have seen in the parts list a wide range of transistors are suitable for use in this circuit. The design objective was that transistors with DC current gains (β) of 100 to 400 could be plugged in and used without design changes. This 100 to 400 range of DC current gains (β) represents the majority of small signal transistors.



The design process is started by a review of the conditions already imposed on the circuit.

There is a battery power supply of 12 volts but only 11.4 volts is available to the circuit; about 0.6 volt is lost because of the diode D_1.

Total current drain has to be less than 3 or 4 milliamps. The LED uses approximately 0.6 of a milliamp[5], so a maximum of 3.4 milliamps has to be shared by the 2 noise generators - 1.7 milliamps each.

Q_2 and R_3 need a minimum of 2 volts to work linearly so there is 9.4 volts to play with. The quiescent point is set to ½ this range i.e. the voltage drop across R_2 will be 4.7 volts. The point between Q_2's collector and R_2 (V_{out}) will then be 6.7 volts above the ground reference (0 volts).

Selecting R_2

The value for R_2 is selected somewhat empirically. The higher the value the more gain the Q_2 stage has, but too high and the sound card will load down the output. The line-input of the sound card is not supposed to have an impedance of less than 10k ohm, so a value for R_2 of 4.7k seems about right. This means a quiescent current through R_2 (I_{R2}) of approximately 1 milliamp, which is comfortably below the maximum allowed quiescent current for the battery pack.

[5] A standard LED would require between 5 and 20 milliamps for the same light output. This is too wasteful for a battery supply and is why a high efficiency LED is specified.

Selecting R_3

R_3 is also selected empirically. It provides negative feedback via the voltage on the emitter, making the design less sensitive to variations in transistors. Feedback also reduces the base current and hence reduces the collector current and overall output signal. Normal values for this resistor are 10% - 20% of the collector resistor. A value of 1k is selected but this may need to be changed if the following calculations show it is a problem.

Selecting R_1

R_1 is not selected empirically although the calculation has some imprecision involved. Estimates need to be made for the DC gain of the Q_2 transistor (β). In this case 200 is a fairly average value for the small signal transistors specified.

The value for the emitter-base break down voltage of Q_1 (V_{EBO}) needs estimating. The transistor data sheet, state this is a minimum of 6 volts. The data sheets don't specify an average or a maximum but a bit of experience will tell you that the minimum 6 volts is also very close to the average and maximum values. 6 volts is a good estimate. 0.6 volts is a reasonable approximation for the base to emitter voltage drop of Q_2 (V_{BEO}).

Lastly there is a need for an approximation of the effect of R_3 that was set as 1k ohm. This approximation is R_3 x β. In this case for R_3 the equivalent resistance is 200k ohms.

The purpose of R_1 is to provide enough base current to cause one milliamp to flow through R_2 and so bias the output correctly. One milliamp is divided by Q_2's β. Therefore, R_1 must allow 5 microamps of current to flow into the base of Q_2.

There is now all the necessary data to calculate the value R_1 by the simple application of ohms law.

$$R_1 + 200k\Omega = \frac{11.4 - 6 - 0.6}{5\mu A}$$

$$\therefore R_1 = 760k\Omega$$

Rechecking the Output With Real Values

This value of 760k for R_1 is a little unfortunate as it is almost smack dead in the middle of two possible resistor values 680k or 820k. Initially I selected 680k because the higher bias current causes more noise.

Recalculating for I_{R_2} gives 1.09 milliamps and V_{out} is 6.27 volts when Q_2's β equal 200. This is acceptably close to the ideal.

The circuit also has to be recalculated when Q_2's β equals 100 and 400.
With Q_2's β set at 100, I_{R_2} calculates to 0.71 milliamps and V_{out} is 8.08 volts.
With Q_2's β set at 400, I_{R_2} calculates to 1.78 milliamps and V_{out} is 3.04 volts.

In both cases the output is within the allowable range for Q. This proves the circuit correctly biases and works for transistors with β ranging for 100 to 400 as was stated in the specification.

In some situations it may be desirable to extend this range of workable β even if it means low gain transistors will not work so well. By changing R_1 to 820k ohms the maximum useable β is extended to approximately 475. This should cover any high gain small signal transistors you are likely to find.

Checking the Feedback Circuit

After posting this circuit I received a volume of critical e-mails from one person. This design section is the result of his unfounded complaints. The plus side is that you can follow the calculations to see that this circuit works and is not just a slight of hand magic trick

R_3 provides DC feedback for the biasing of Q_2. You can check its effect by replacing it with a short circuit and redoing the biasing calculations. With R_1 set to 680k the base current to Q_2 is

$Q_{2Ib} = \dfrac{11.4 - 6 - .6}{680k\Omega}$. This equals 7.06 microamps regardless of transistor β. With Q_2's β set to

325, I_{R_2} calculates to 2.29 milliamps.

This makes the Q point 0.62 volts – under these conditions the circuit simply won't work. A working β ranging from 100 to 300 does not cover the typical range of small signal transistors.

R_3 provides enough feedback to allow the circuit to work with β's up to 400 (or 475 if you use a 820k ohm resistor). This covers the typical range for the suggested transistors. The small amount of DC feedback extends the working range of β enough so the circuit works properly. This is the end of the argument about removing R_3.

C_2 provides a feedback bypass for the AC component of the output signal, without it the maximum AC gain of this stage would be 4.7 (13.4db); C_2 allows more db gain. Calculating from Shockley's formula for incremental resistance of a transistor junction shows a gain of 188 or 45 db. This use of a capacitor is very common in simple transistor amplifiers. The combination of R_3 and C_2 allow the use of a wider range of transistors without reducing the maximum AC gain.

C_1 bypasses R_1 and provides a low impedance AC signal to the base of Q_2 and allows enough noise signal to pass to eliminate the need for an additional amplifier stage. This one capacitor is responsible for almost 20 db of gain. C_1 is necessary and stays.

Frequency Response

There are three checks needed for frequency response of filters formed by the capacitors and associated resistors. In this case all of these checks are for high pass filters and it is necessary for each filter to have a cut off frequency below 20 hertz.

C_1

C_1 combines with R_3 x β to form a high pass filter that feeds into the base of Q_2.

$$f = \frac{1}{2\pi\beta RC_1}$$

C_1 0.1μF
$\beta = 100$
$R_3 \times \beta = 100k\Omega$
$f = 16$ hertz

C_2
C_2 combines with R_3 to form a high pass filter to provide an AC bypass to the emitter of Q_2. This effectively allows more AC gain of the signal to Q_2.

$$f = \frac{1}{2\pi RC_2}$$

$C_2 = 10\ \mu F$
$R_3 = 1k\Omega$
$f = 16$ hertz

C_3
C_3 combines with the parallel combination of R_2 and R_4 to set the low frequency output of the noise generator.

$$R_x = \frac{R_2 \times R_4}{R_2 + R_4}$$

$C_3 = 10\ \mu F$
$R_x = 4.5k\Omega$

$$f = \frac{1}{2\pi R_x C_3}$$

$f = 3.5$ hertz

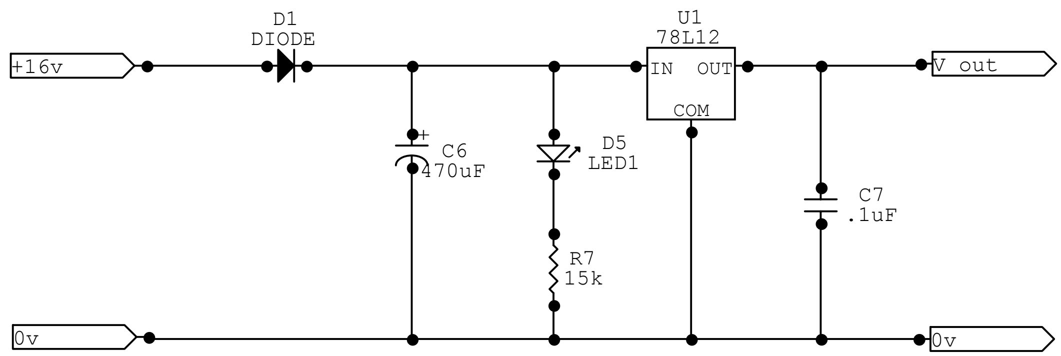
All of the filters formed by the capacitors have frequency characteristics that are less significant than the analogue filters provided on the sound card.

Temperature Effects

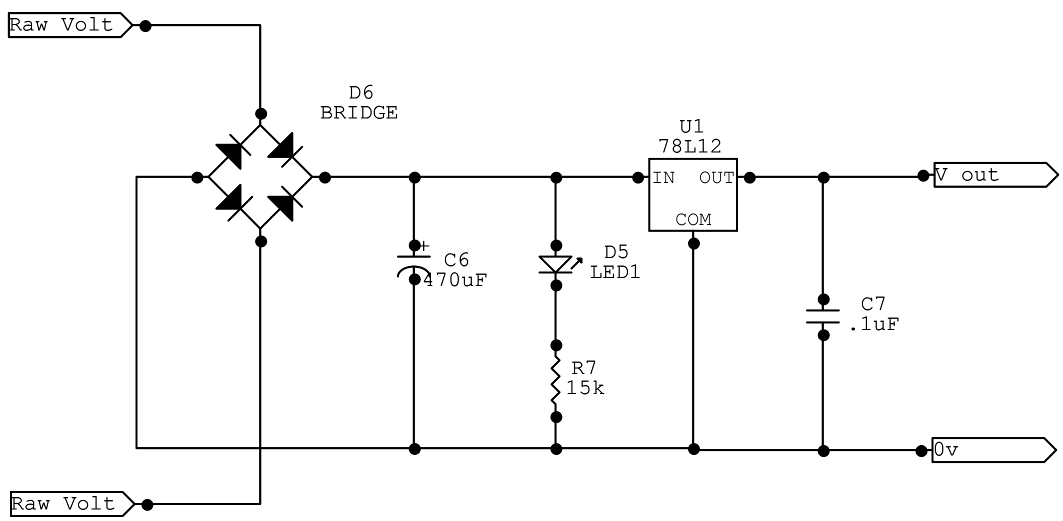
A plastic packaged small signal transistor has a thermal resistance -junction to ambient ($R_{\theta JA}$) typically of 200 degrees Celsius per watt. For Q_2 in the worst case situation the transistor conducts 1.78 milliamps while dropping 4.92 volts. This represents 8.76 milliwatts and a temperature rise of an insignificant 1.75 degrees. The temperature change in the normal ambient conditions is also insignificant, at perhaps 10 or a maximum of 20 degrees. Looking on the data sheets shows none of these variations are going to make a significant change to the operation of this circuit.

Adding A Regulated Power Supply

If you want to power your noise source from the mains then the simplest solution is to use a regulated plug pack with the original circuit. Regulated plug packs are expensive relative to this noise generator so you might want to use the cheaper un-regulated type. To do this you will need to build a small regulator circuit into the noise source. Either of the two circuits below will suffice. The circuits replace D_1, D_2, C_4 and R_5. The first circuit is for use with a 9 or 12 volt DC plug pack. This works because the indicated voltage is the output voltage at the rated current. The entire noise circuit draws very little current and the plug packs output voltage will be high enough for reliable operation. In these diagrams U1 is a 12 volt regulator and is marked as a 78L12. There are several different versions of the same chip; such as 78M12 and 7812 any of them will perform in this application. The 78L12 is just the physically smallest.



This second circuit will work with AC plug packs and DC plug packs regardless of polarity. This is the most trouble free circuit in use. Just grab any plug pack and plug it in. This is especially useful if you share plug packs between devices, as you don't have to worry about changing to "correct" polarity.



Software Summary - Traps and Pitfalls

The following statement is emphasised:

Transistor junction noise provides a source of true random data that can be sampled and stored on a computer. However, this sampled data does not contain 1 bit of entropy per bit of stored data. It is therefore necessary to post process the data samples to distil the existing entropy into a smaller number of high entropy bits.

The major reason this project is so cheap is the use of the sound-card for interfacing and data conversion, and doing post-processing on the host computer. This means you need some software. You can use any program that accesses the sound card to collect the data samples, but the data still needs processing to make it into acceptable random data with high entropy per bit. This post-processing is necessary because the noise output from the transistor junctions is normally distributed around a mean whereas true random data is uniformly distributed across the data samples' entire range. The analogue filters on the sound card and noise source also introduce auto correlations into the data stream. Unprocessed data samples are easy to detect from true randomness.

There are two ways to proceed: post-processing in 8-bit mode and post-processing in 16-bit mode.

8-bit Mode

Precision analogue to digital converters are extremely difficult and expensive to manufacture. The common faults with analogue to digital (A-D) converters include non-linearity, input offsets and missing codes. This is especially true with the A-D converters used to record sound for playback on a computer where there is the enormous pressure to keep costs down. Because the ear is such an imprecise listening device, little attempt is made to minimise these A-D faults. The A-D converter in a PC is not a precision, high performance device; don't fool yourself into thinking that it is.

The analogue interfacing circuitry is likewise designed primarily with cost in mind, giving rise to imperfections such as poor signal to noise ratio, poor dynamic range and cross talk.

Under these conditions it is simply not possible to sample 16-bits of data and use that as your random number source. It just won't work. However, if you change the sample mode to 8-bits the input offset and non-linearity problems are reduced by a factor of 2^8. The problem of missing codes disappears completely and you have voluntarily reduced the demands on the analogue interface to something it is likely to be able to provide[6]. This is a workable solution. Even so, you will still have to test the individual bits to find which ones are in the useable range.

[6] Consider the dynamic range of the sound card. If you short out one of the line inputs you can measure the noise floor of the A-D converter plus the analogue interface circuitry. A typical reading is –60db, which represents about 10-bits of dynamic range, a far cry from the 96db range needed for accurate 16-bit samples, but perfectly acceptable for 8-bit mode. Please note that this test can't be done with Cooledit 96. Under these conditions Cooledit 96 simply blanks the sound level meter, giving the impression of a precision piece of analogue equipment so fine that it can actually measure the noise of the spinning electrons.

I suggest at least a bit frequency test to quickly select the possibly usable bits, followed by a poker test to confirm the usable range of bits. This seems to be enough. Once these bits are reassembled into bytes, the data stream passes FIPS-140 and Die-Hard testing.

In the 8-bit mode simply using only those bits that pass the simple randomness tests rather than the entire byte reduces the problems of autocorrelations. Each unused bit reduces the effect of autocorrelation by at least 2^n.

It is a tedious approach but it provides a good quality data stream.

16-Bit Mode

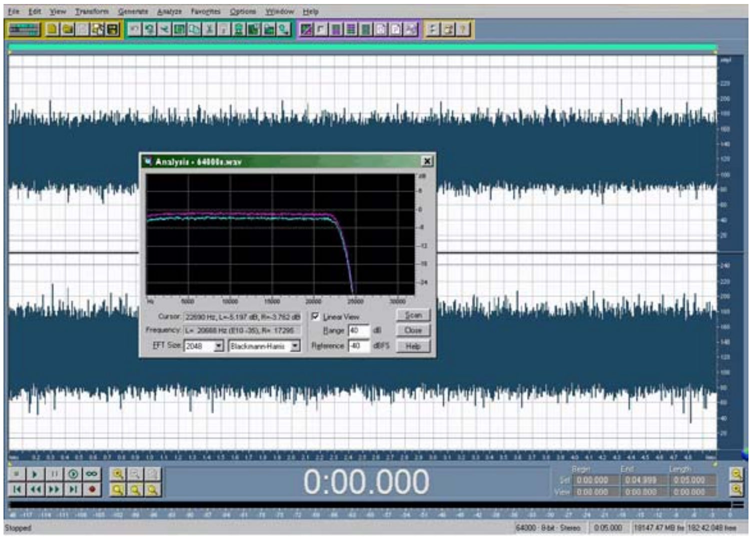
Another way of approaching the problem of the imperfect A-D converters is to run the data through a hashing program such as SHA-1, RIPEMD-160 or such. The hashing program does not have to be cryptographically secure; all the hashing needs to do is distil randomness from a large input of bits into a smaller collection of output bits. For this hashing MD-5 or MD-4 would also be suitable although they are no longer considered secure cryptographic hash functions. Hashing does appear to gracefully handle all the possible defects in the A-D conversion such as non-linearity, input offset, missing codes and autocorrelations providing the input has enough entropy.

For this reason it would also be desirable to know approximately how much entropy is in the collected samples before they are hashed. Another reason for an approximation of available entropy is that even with fast hardware, the data collection takes time, in some cases it would be desirable to ensure that collected data's entropy is spread over as many samples as possible.

The barest minimum entropy needed for hash functions are 2.66 bits per byte for MD4, MD5, RIPEMD, RIPEMD-128 and SHA-256. For RIPEMD-160 and SHA-1 this figure is 3.33 bits of entropy per byte. This is the barest minimum; more would be better –within reason.

One More Trap

There is one more trap not to fall into and that is over sampling. There are analogue filters on the inputs of a sound-card. All sound-cards have then and you can't and shouldn't bypass them. Unfortunately, one of their side effects is to limit the maximum frequencies that make it to the A-D converter for sampling. If you crank the collection rate above 44100kHz you don't get more entropy, you get the same amount only spread over more samples. Each sample has a high correlation with the adjacent samples.



Although this data has been sampled at 64000 samples per second, the analogue filters progressively attenuate the input frequencies above 20kHz. The extra samples contain no extra entropy than is available if the noise was sampled at 44100kHz.

Unless you spent $2000 or more on your sound card, the same comments apply to sampling in 24-bit mode.

There is one situation where over-sampling will bring benefits. If the input data samples have ample entropy for hashing, over-sampling is a pain free way of spreading that entropy across more samples.

Appendix

Typical Transistor Data Sheet

Note also the drawing of the transistor showing the orientation of the pins

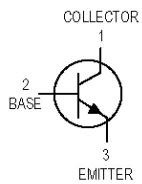
MOTOROLA
SEMICONDUCTOR TECHNICAL DATA

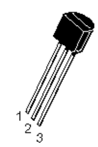
Order this document
by BC546/D

Amplifier Transistors

NPN Silicon

BC546, B
BC547, A, B, C
BC548, A, B, C





CASE 29–04, STYLE 17
TO–92 (TO–226AA)

MAXIMUM RATINGS

Rating	Symbol	BC 546	BC 547	BC 548	Unit
Collector–Emitter Voltage	V_{CEO}	65	45	30	Vdc
Collector–Base Voltage	V_{CBO}	80	50	30	Vdc
Emitter–Base Voltage	V_{EBO}	6.0			Vdc
Collector Current — Continuous	I_C	100			mAdc
Total Device Dissipation @ T_A = 25°C Derate above 25°C	P_D	625 5.0			mW mW/°C
Total Device Dissipation @ T_C = 25°C Derate above 25°C	P_D	1.5 12			Watt mW/°C
Operating and Storage Junction Temperature Range	T_J, T_{stg}	–55 to +150			°C

THERMAL CHARACTERISTICS

Characteristic	Symbol	Max	Unit
Thermal Resistance, Junction to Ambient	$R_{\theta JA}$	200	°C/W
Thermal Resistance, Junction to Case	$R_{\theta JC}$	83.3	°C/W

ELECTRICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
OFF CHARACTERISTICS						
Collector–Emitter Breakdown Voltage (I_C = 1.0 mA, I_B = 0)	BC546 BC547 BC548	$V_{(BR)CEO}$	65 45 30	— — —	— — —	V
Collector–Base Breakdown Voltage (I_C = 100 μAdc)	BC546 BC547 BC548	$V_{(BR)CBO}$	80 50 30	— — —	— — —	V
Emitter–Base Breakdown Voltage (I_E = 10 μA, I_C = 0)	BC546 BC547 BC548	$V_{(BR)EBO}$	6.0 6.0 6.0	— — —	— — —	V
Collector Cutoff Current (V_{CE} = 70 V, V_{BE} = 0) (V_{CE} = 50 V, V_{BE} = 0) (V_{CE} = 35 V, V_{BE} = 0) (V_{CE} = 30 V, T_A = 125°C)	 BC546 BC547 BC548 BC546/547/548	I_{CES}	 — — — —	 0.2 0.2 0.2 —	 15 15 15 4.0	 nA μA

REV 1

MOTOROLA

ELECTRICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted) (Continued)

Characteristic		Symbol	Min	Typ	Max	Unit
ON CHARACTERISTICS						
DC Current Gain		h_{FE}				—
(I_C = 10 µA, V_{CE} = 5.0 V)	BC547A/548A		—	90	—	
	BC546B/547B/548B		—	150	—	
	BC548C		—	270	—	
(I_C = 2.0 mA, V_{CE} = 5.0 V)	BC546		110	—	450	
	BC547		110	—	800	
	BC548		110	—	800	
	BC547A/548A		110	180	220	
	BC546B/547B/548B		200	290	450	
	BC547C/BC548C		420	520	800	
(I_C = 100 mA, V_{CE} = 5.0 V)	BC547A/548A		—	120	—	
	BC546B/547B/548B		—	180	—	
	BC548C		—	300	—	
Collector–Emitter Saturation Voltage		$V_{CE(sat)}$				V
(I_C = 10 mA, I_B = 0.5 mA)			—	0.09	0.25	
(I_C = 100 mA, I_B = 5.0 mA)			—	0.2	0.6	
(I_C = 10 mA, I_B = See Note 1)			—	0.3	0.6	
Base–Emitter Saturation Voltage (I_C = 10 mA, I_B = 0.5 mA)		$V_{BE(sat)}$	—	0.7	—	V
Base–Emitter On Voltage		$V_{BE(on)}$				V
(I_C = 2.0 mA, V_{CE} = 5.0 V)			0.55	—	0.7	
(I_C = 10 mA, V_{CE} = 5.0 V)			—	—	0.77	
SMALL–SIGNAL CHARACTERISTICS						
Current–Gain — Bandwidth Product		f_T				MHz
(I_C = 10 mA, V_{CE} = 5.0 V, f = 100 MHz)	BC546		150	300	—	
	BC547		150	300	—	
	BC548		150	300	—	
Output Capacitance (V_{CB} = 10 V, I_C = 0, f = 1.0 MHz)		C_{obo}	—	1.7	4.5	pF
Input Capacitance (V_{EB} = 0.5 V, I_C = 0, f = 1.0 MHz)		C_{ibo}	—	10	—	pF
Small–Signal Current Gain		h_{fe}				—
(I_C = 2.0 mA, V_{CE} = 5.0 V, f = 1.0 kHz)	BC546		125	—	500	
	BC547/548		125	—	900	
	BC547A/548A		125	220	260	
	BC546B/547B/548B		240	330	500	
	BC547C/548C		450	600	900	
Noise Figure		NF				dB
(I_C = 0.2 mA, V_{CE} = 5.0 V, R_S = 2 kΩ,	BC546		—	2.0	10	
f = 1.0 kHz, Δf = 200 Hz)	BC547		—	2.0	10	
	BC548		—	2.0	10	

Note 1: I_B is value for which I_C = 11 mA at V_{CE} = 1.0 V.

Additional Data Sheets

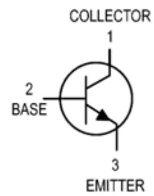
MOTOROLA
SEMICONDUCTOR TECHNICAL DATA

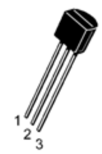
Order this document
by P2N2222A/D

Amplifier Transistors

NPN Silicon

P2N2222A





CASE 29–04, STYLE 17
TO–92 (TO–226AA)

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Collector–Emitter Voltage	V_{CEO}	40	Vdc
Collector–Base Voltage	V_{CBO}	75	Vdc
Emitter–Base Voltage	V_{EBO}	6.0	Vdc
Collector Current — Continuous	I_C	600	mAdc
Total Device Dissipation @ T_A = 25°C Derate above 25°C	P_D	625 5.0	mW mW/°C
Total Device Dissipation @ T_C = 25°C Derate above 25°C	P_D	1.5 12	Watts mW/°C
Operating and Storage Junction Temperature Range	T_J, T_{stg}	–55 to +150	°C

THERMAL CHARACTERISTICS

Characteristic	Symbol	Max	Unit
Thermal Resistance, Junction to Ambient	$R_{\theta JA}$	200	°C/W
Thermal Resistance, Junction to Case	$R_{\theta JC}$	83.3	°C/W

ELECTRICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted)

Characteristic	Symbol	Min	Max	Unit
OFF CHARACTERISTICS				
Collector–Emitter Breakdown Voltage (I_C = 10 mAdc, I_B = 0)	$V_{(BR)CEO}$	40	—	Vdc
Collector–Base Breakdown Voltage (I_C = 10 μAdc, I_E = 0)	$V_{(BR)CBO}$	75	—	Vdc
Emitter–Base Breakdown Voltage (I_E = 10 μAdc, I_C = 0)	$V_{(BR)EBO}$	6.0	—	Vdc
Collector Cutoff Current (V_{CE} = 60 Vdc, $V_{EB(off)}$ = 3.0 Vdc)	I_{CEX}	—	10	nAdc
Collector Cutoff Current (V_{CB} = 60 Vdc, I_E = 0) (V_{CB} = 60 Vdc, I_E = 0, T_A = 150°C)	I_{CBO}	 — —	 0.01 10	μAdc
Emitter Cutoff Current (V_{EB} = 3.0 Vdc, I_C = 0)	I_{EBO}	—	10	nAdc
Collector Cutoff Current (V_{CE} = 10 V)	I_{CEO}	—	10	nAdc
Base Cutoff Current (V_{CE} = 60 Vdc, $V_{EB(off)}$ = 3.0 Vdc)	I_{BEX}	—	20	nAdc

NPN switching transistor 2N3904

FEATURES

- Low current (max. 200 mA)
- Low voltage (max. 40 V).

APPLICATIONS

- High-speed switching.

DESCRIPTION

NPN switching transistor in a TO-92; SOT54 plastic package. PNP complement: 2N3906.

PINNING

PIN	DESCRIPTION
1	collector
2	base
3	emitter

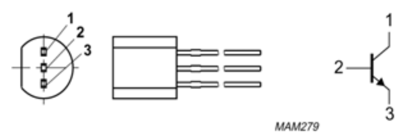


Fig.1 Simplified outline (TO-92; SOT54) and symbol.

LIMITING VALUES

In accordance with the Absolute Maximum Rating System (IEC 134).

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
V_{CBO}	collector-base voltage	open emitter	–	60	V
V_{CEO}	collector-emitter voltage	open base	–	40	V
V_{EBO}	emitter-base voltage	open collector	–	6	V
I_C	collector current (DC)		–	200	mA
I_{CM}	peak collector current		–	300	mA
I_{BM}	peak base current		–	100	mA
P_{tot}	total power dissipation	$T_{amb} \leq 25$ °C; note 1	–	500	mW
T_{stg}	storage temperature		–65	+150	°C
T_j	junction temperature		–	150	°C
T_{amb}	operating ambient temperature		–65	+150	°C

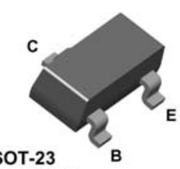
Note

1. Transistor mounted on an FR4 printed-circuit board.

FAIRCHILD
SEMICONDUCTOR ™

PN100 PN100A | MMBT100 MMBT100A

TO-92



SOT-23
Mark: NA / NA1

NPN General Purpose Amplifier

This device is designed for general purpose amplifier applications at collector currents to 300 mA. Sourced from Process 10.

Absolute Maximum Ratings* $T_A = 25°C$ unless otherwise noted

Symbol	Parameter	Value	Units
V_{CEO}	Collector-Emitter Voltage	45	V
V_{CBO}	Collector-Base Voltage	75	V
V_{EBO}	Emitter-Base Voltage	6.0	V
I_C	Collector Current - Continuous	500	mA
T_J, T_{stg}	Operating and Storage Junction Temperature Range	-55 to +150	°C

*These ratings are limiting values above which the serviceability of any semiconductor device may be impaired.

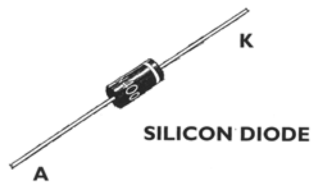
NOTES:
1) These ratings are based on a maximum junction temperature of 150 degrees C.
2) These are steady state limits. The factory should be consulted on applications involving pulsed or low duty cycle operations.

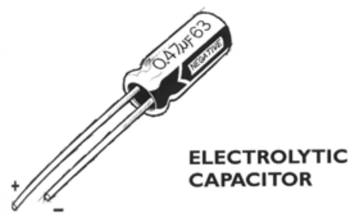
Thermal Characteristics $T_A = 25°C$ unless otherwise noted

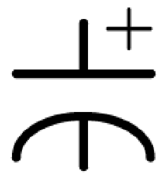
Symbol	Characteristic	Max		Units
		PN100 PN100A	*MMBT100 *MMBT100A	
P_D	Total Device Dissipation Derate above 25°C	625 5.0	350 2.8	mW mW/°C
$R_{\theta JC}$	Thermal Resistance, Junction to Case	83.3		°C/W
$R_{\theta JA}$	Thermal Resistance, Junction to Ambient	200	357	°C/W

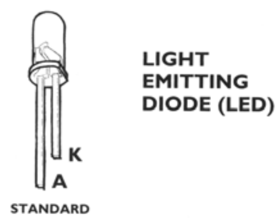
*Device mounted on FR-4 PCB 1.6" X 1.6" X 0.06."

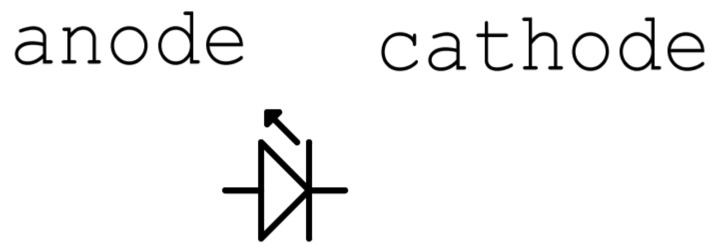
Component Identification











RESISTOR & CAPACITOR DATA

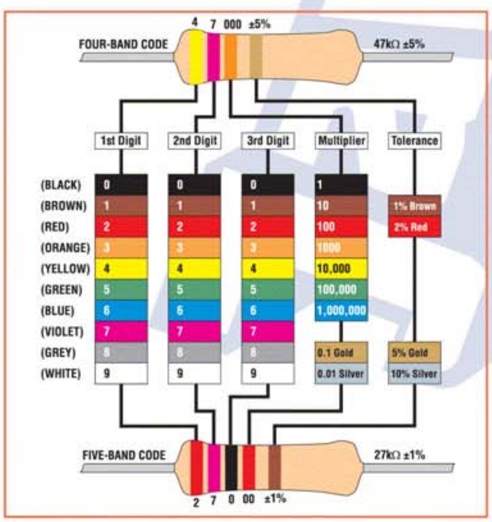
Resistors

Many resistors are so small that it would be difficult to print their value and % tolerance on their body in digits. To overcome this, a coding system based on bands of distinctive colours was developed to assist in identification. Learning this 'colour code' is not as necessary as it used to be (thanks to accurate, low cost digital multimeters!), but it's not hard to learn and it's quite useful knowledge anyway.

The first thing to know is that in each decade of resistance — i.e., from 10 - 100Ω, 100 - 1kΩ, 1k - 10kΩ, etc — there are only a finite number of different nominal values allowed. Most common resistors have values in the 'E12' series, which only has 12 allowed values per decade. Normalised these are 1.0, 1.2, 1.5, 1.8, 2.2, 2.7, 3.3, 3.9, 4.7, 5.6, 6.8 and 8.2. Multiples of these values are simply repeated in each decade — e.g., 10, 12, 15, 18 and so on. Note that the 'steps' between these values are always very close to 20%, because the E12 series dates from the days of resistors with ±10% tolerance.

To allow greater accuracy in circuit design, modern 1% tolerance resistors are made in a larger range of values: the 'E24' series, which has 12 *additional* allowed values per decade as shown in the table. As before, these nominal values are simply repeated in each decade. The table at right shows both the E12 and E24 allowed values for comparison.

The next thing to know is that there are **two** different resistor colour coding systems in use: one using a total of 4 colour bands, and the other 5. The 5-band system is generally used for 2% and closer tolerance resistors, even though the 4-band system is quite capable of handling any resistors with E12 or E24 values. Both systems use the same band colours to represent the various digits; the main difference is that 5-band resistors have an additional 'third band', which is almost always BLACK to represent a third digit of '0'. Here's how both systems work in practice:



4-band resistors will almost always have values in the E12 series, while 5-band resistors can have any value in the E24 series. This is worth remembering, because depending on the resistor's body colour, some of the band colours may not be easy to distinguish. Blue (6) and grey (8) sometimes look very similar, as do red (2), brown (1) and orange (3). So if you're in doubt, check the apparent coded value against the allowed E12 or E24 values to see if it's 'legal' — or check with a digital multimeter, just to make sure.

Preferred Resistor Values
(within each decade)

E12 Series	E24 Series
10	10
	11
12	12
	13
15	15
	16
18	18
	20
22	22
	24
27	27
	30
33	33
	36
39	39
	43
47	47
	51
56	56
	62
68	68
	75
82	82
	91

Capacitors

Virtually all of the capacitors stocked by Jaycar have their electrical values printed directly on their body, in digits and letters. However there's often still a coding system, which can make it a bit tricky to work out the capacitance, voltage rating, tolerance and so on until you know how it works. This is explained below.

Incidentally, so-called 'greencaps' (which can actually be brown, dark red or even blue!) are one type of metallised polyester film capacitor, like the 'MKT' type — which tends to be smaller, and in a more tightly controlled rectangular package. Similarly the 'monolithic' type is a type of multilayer ceramic capacitor, designed to combine high capacitance with very low self-inductance.

Plastic film, Ceramic & Monolithic Capacitors

Most of these types have their nominal value either printed directly on them or use the 'EIA' coding system, which is a bit like resistor colour coding, but in digits: the first two digits followed by a 'multiplier' showing the number of zeroes. With this code the value is generally given in picofarads (pF), which you'll need to divide by either one million or one thousand (respectively) if you want the value in microfarads (μF) or nanofarads (nF).

Hence a capacitor marked '104' has a value of 10 with 4 zeroes after it, or 100,000pF (which is the same as 100nF, or 0.1μF). Similarly '681' means 68 with a single zero, or 680pF, while '472' means 47 with two zeroes, or 4700pF (which is the same as 4.7nF).