USER’S MANUAL
VERSION 1.0
23-JUNE, 2018

for

ROBOEAR-1 CIRCUIT BOARD

Lucid Technologies
http://www.lucidtechnologies.info/
Email: info@lucidtechnologies.info

CONTENTS

1.0 Introduction

The ROBOEAR is designed to detect the direction to nearby sound sources and transfer that information to another system. It uses a three microphone array to capture the difference in time that sound arrives at each microphone. It can also function as a sound acquisition front-end for computers more powerful than the PIC microcontroller on the ROBOEAR.

1.1 General Specifications

1. Pin-selectable interface modes
   a. RS232 asynchronous serial interface
      i. Pin-selectable baud rate: 9600,19200,38400,76800
      ii. Serial data format: 8 bit, No Parity, 1 Stop Bit
   b. SPI synchronous serial interface
      i. Pin-selectable SPI modes: 0,1,2,3
2. Based on PIC16F1847 enhanced mid-range CPU
   a. Operating Voltage: 5 VDC
   b. No external oscillator or crystal required
3. Three microphone array
   a. Electret microphones
   b. Arranged in an equilateral triangle, 10 cm on a side
4. Power input (J1)
   a. Coaxial power plug, 5.5mm O.D., 2.1mm I.D., center positive
   b. External 9 volts DC at >= 100 ma

2.0 Circuit Description

This section describes the circuitry on the ROBOEAR-1 circuit board.

2.1 Power Supply

Schematic sheet 1 (Appendix J) shows the power supply circuitry. The ROBOEAR circuit board is set up for an external 9 VDC power supply connected via the 5.5 mm coaxial connector, J1. All ROBOEAR circuitry runs on +5V, which is provided by U3, the 78M05, which regulates the 9 VDC input down to 5 VDC. The TLE2426 rail-splitter provides a 2.5 VDC *virtual ground* for the single supply op-amps used in the analog circuitry.

2.2 Microcontroller and Serial Communications

The PIC16F1847 microcontroller, or PIC for short, is designated as U1 on sheet 3 of the schematics. The PIC has 14 Kbytes of flash program memory, 1 Kbyte of data memory (RAM), 256 bytes of EEPROM memory, an internal clock oscillator, a universal asynchronous receiver transmitter (UART), a synchronous serial port (SPI), and a 10-bit analog-to-digital converter. Lucid Technologies firmware is initially programmed in the PIC's flash memory. The PIC's internal oscillator is programmed to operate at 16 MHZ.

The RS-232 serial port connector (J2) is described in detail in Appendix E. U40 is a MAX232, 5 volt powered RS-232 interface with two drivers and two receivers. One receiver/driver pair handles RS-232 data to/from the ROBOEAR. The other receiver/driver pair handles the control lines RTS and CTS. Typically, the ROBOEAR receives RTS and sends it back to the host as CTS.

See Appendix F for how to set up the control lines.

2.3 Microphone inputs, Op-amps and Virtual Ground

The ROBOEAR has three analog channels. Each channel has a microphone input and two op-amps packaged in a single MCP602. Components in each channel are labeled 1X, 2X and 3X respectively. The analog circuitry is shown on sheet 2 of the schematics.

The ROBOEAR is designed to work with most inexpensive electret microphones that have an output impedance of approximately 2200 ohms. The microphone is AC coupled to the first op-amp which is configured for a positive gain of 100. The output of the first op-amp is AC coupled to the second op-amp which is configured for a negative gain of 10 and a lowpass roll-off of 4800 Hz. With no sound input (in quiet) the output of the second op-amp will be at the *virtual ground* value of 2.5 volts.

2.4 Analog to Digital Converters

The final component in each channel is the MCP3001 analog-to-digital converter (ADC). The MCP3001 is a successive approximation 10-bit ADC with onboard sample and hold circuitry. Communication with the device is done using a simple serial interface compatible with the SPI protocol. The device is capable of sample rates up to 200 ksps at a clock rate of 2.8 MHZ at 5 volts.

Although the PIC has an internal multi-input ADC it is not suitable for this application. The PIC has one ADC with an analog multiplexor on the input; thus it does sequential conversions. Because cross-correlation depends on knowing the value of multiple inputs at the same points in time, simultaneous sampling is required. This is accomplished by connecting the clock and chip-select inputs of all three MCP3001's in parallel. The serial data outputs go to three inputs on the same PIC input port. The serial output from the ADCs is stored by the PIC in packed format and unpacked into individual channels after all the samples have been acquired. Because the MCP3001 is a serial successive approximation converter the conversion can be halted before reaching 10 bits. Because the ROBOEAR has 8-bit RAM it routinely uses 8-bit conversions; 10-bit conversions are available but must be analyzed externally.

3.0 Theory of Operation

In psychoacoustics the ability to determine the direction and distance of a sound source is called localization. There are two classical methods used to describe how humans determine the direction to a sound source, phase difference predominates at low frequencies and amplitude difference at high frequencies, at middle frequencies (around 1500 Hz) neither works particularly well. Differences in amplitude are more effective at higher frequencies because there is more shadowing effect from the head producing a more pronounced amplitude difference between the ears. Differences in phase are more effective at low frequencies because sound wavelengths are long compared to the size of the head and diffraction allows the sound to bend around the far side of the head.

3.1 Phase Differences

In signal processing a common way to look at phase differences of noisy signals is to use cross correlation. Cross-correlation is the comparison of two different time series to detect if there is a correlation between their peaks and valleys. Simply put, one signal is time shifted with respect to the other and the correlation is measured at each time shift. The metric used to measure how well they correlate, or line up, with each other is the cross-correlation function shown here.

$$cross(L) = \sum_{n=0}^{N} x(n) * y(n+L)$$

For each time shift, called a lag (L), the values of the first signal (x) are multiplied by the time shifted values of the second signal (y) and the N product terms are added. The lag step-size is determined by the sample rate; for example, if the signals are sampled at 20 kHz, lag values would change in increments of 50 microseconds. Consider Figure 3.1.

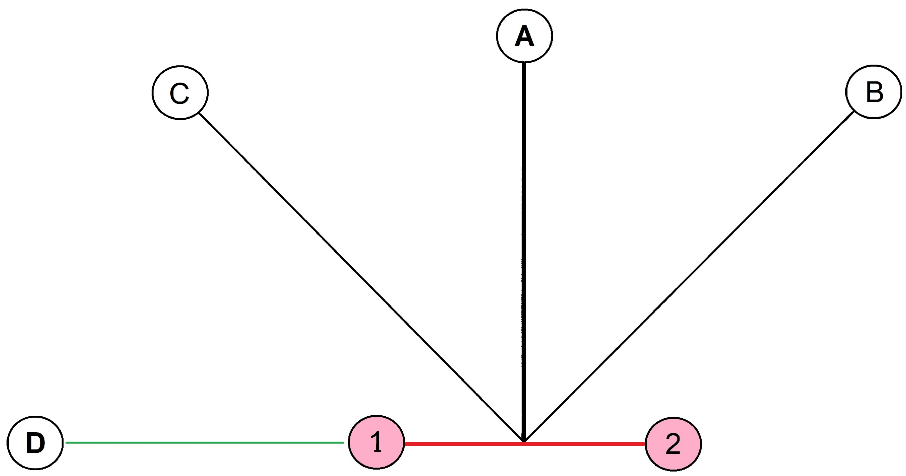


Figure 3.1

The baseline between two microphones is shown in red. Sound from A will arrive at mic1 and mic2 at the same time so the max cross-correlation value will be at zero lag. Sound from B will arrive at mic1 last; in this case the sampled values from mic1 would have to be shifted backward in time (-time) to line up with the sampled values from mic2 so the maximum cross-correlation value would be at a negative lag. Conversely for C, the values from mic1 would have to be shifted forward in time (+ time) to line up with the signal from mic2 so the maximum cross-correlation value would be at a positive lag. Any source to the left of A will have its maximum cross-correlation value at a positive lag. Because D is on the extension of the baseline (green), this produces the greatest time difference, the maximum lag, between the two microphones. If the distance between the microphones is 10 cm and the speed of sound is 34,320 cm/sec then it will take 291 usec for the sound to travel from mic1 to mic2. If the signals are sampled at 20 kHz the maximum cross-correlation would probably occur at a lag of +6 because (291 usec / 50 usec) = 5.8.

However there is a problem with the simple diagram in Figure 3.1. What if A was on the other side of the baseline? Could we tell this from the from sampled values at mic1 and mic2? The answer is no, we could not tell which side of the baseline the sound came from, the solution is ambiguous. There is always a second potential solution as shown in Figure 3.2. This is why the ROBOEAR has three microphones and three baselines which yield three ambiguous solutions from which we are able to produce one unique solution. The time difference between sounds arriving at mic2 and mic1 is the same for sources on different sides of the baseline if they are at equal angles from the baseline, as shown by angles A and B in Figure 3.2. But how do we determine angle A?

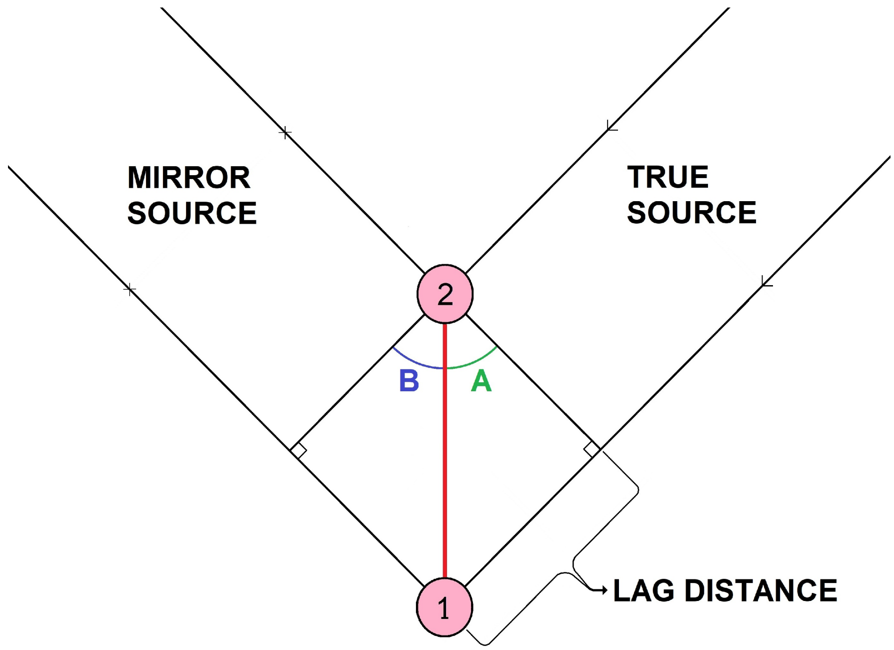


Figure 3.2

$$\sin(A) = \frac{(Opposite)}{(Hypoenuse)} = \frac{Lag}{Baseline}$$

Where the Lag distance equals the lag time (in multiples of the sample period) times the speed of sound. The angle A is the arcsine of this ratio.

For example, if the maximum cross-correlation occurs at L=4 the Lag distance is 4*(50 usec)*(34,320 cm/sec) = 6.86 cm. If we assume the same 10 cm baseline then the angle A is the arcsine of 6.86/10 or 43.3 degrees. This illustrates how we can use phase, or time, difference to determine the direction to a sound source.

$$A = \sin^{-1}\left(\frac{Lag}{Baseline}\right) = \sin^{-1}\left(\frac{6.86}{10}\right) = 43.3^0$$

3.2 Amplitude Differences

As mentioned earlier, differences in amplitude are more effective in determining direction at higher frequencies because there is more shadowing effect from the head and the amplitude difference between the ears is more pronounced. The ROBOEAR does not have a head and it has three ears (microphones) instead of two. So the way the ROBOEAR uses amplitude is to consider the amplitude of the samples from each microphone as a vector from the origin to the microphone. Figure 3.3 shows the geometry.

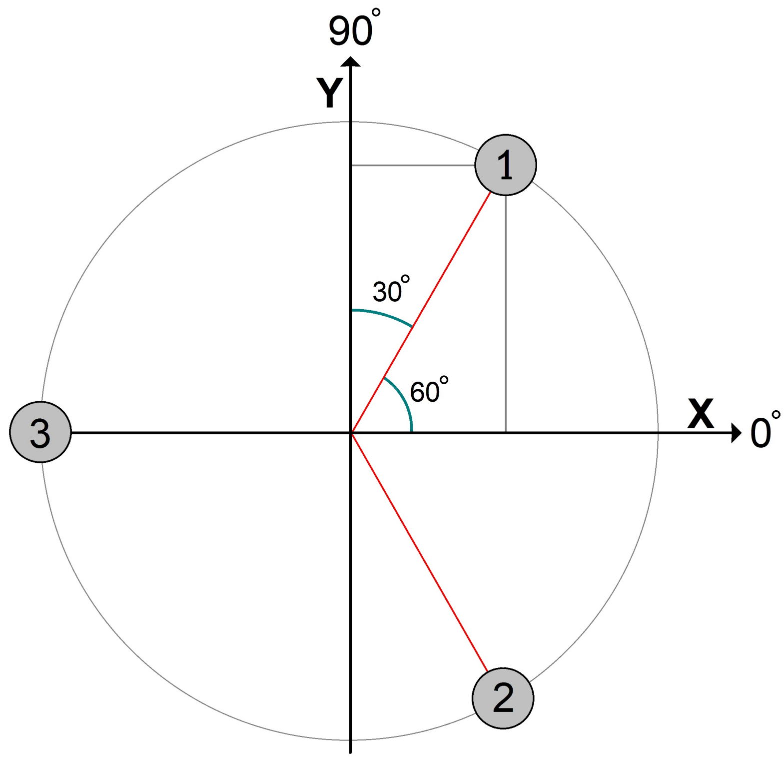


Figure 3.3

The amplitude of the samples from each microphone is the maximum value minus the minimum value in a 127 sample set. Let:
h1 = the amplitude of the samples from mic1,
R1x = the X component of the vector from the origin in the direction of mic1,
R1y = the Y component of the vector from the origin in the direction of mic1,
and similar definitions for mic2 and mic3. Then:

R1x = cos(60°)*h1 = 0.5*h1	R1y = sin(60°)*h1 = 0.866*h1
R2x = cos(-60°)*h2 = 0.5*h2	R2y = sin(-60°)*h1 = -0.866*h2
R3x = -h3	R3y = 0

Then the vector sum (Rs) = Rsx + Rsy = (R1x + R2x R3x) + (R1y + R2y) and the angle to the sound source is the arctan of Rsy/Rsx. Obviously these calculations will only work if the samples have a reasonable amplitude and there is some difference between channels, otherwise the results will be noisy. Consider the following example:

h1 = 150	h2 = 120	h3 = 90	
R1x = 75	R2x = 60	R3x = -90	Rsx = 45
R1y = 130	R2y = -104	R3y = 0	Rsy = 26

arctan(Rsy/Rsy) = arctan(26/45) = arctan(0.577) = 30°

3.3 Practical Problems

The theory is all very nice but there are a lot of practical problems with trying to replicate the human ability to localize sound. Several problems affect both phase and amplitude processing. Of course noise is always a problem in signal processing, and noise can come from many sources. There is noise from sources you don't want to localize. There is noise from vibrations, such as robot motion, that affects all three microphones simultaneously. There are echos from nearby walls, interior corners, and ceilings; echos can be thought of as acoustic multipath. The ROBOEAR microphones are laid out in a horizontal plane and the theory assumes a sound source in the same plane. If the sound comes from more than 45° above or below the microphones, accuracy drops off rapidly.

For phase difference using cross-correlation on an 8-bit microcontroller there are a whole host of problems. The most obvious is memory limitations. I originally thought I could work around that by using a zero-crossing-detector (a 1-bit ADC) and doing all the multiplies needed to compute cross-correlation functions as exclusive-ORs. Theoretically it should work - I have the simulations to prove it - but analog noise, differences in op-amp offset voltages and comparator response time necessitated such a high sampling rate that I was still memory limited. The ROBOEAR processes 8-bit ADC data sampled at 20 kHz. The packed raw data uses 508 bytes of RAM. The 127 unpacked data bytes from each channel are stored in a 128 byte buffer. That uses 892 bytes of the PIC16F1847's 1024 bytes of RAM leaving only 132 bytes for variables and 24-bit math registers.

For amplitude difference processing the major problem is that there usually isn't much amplitude difference between the channels. Sound pressure drops off with the square of distance. This means the sound source has to be close to the ROBOEAR microphones in order for there to be an appreciable difference in sound pressure over the few centimeters between microphones.

3.4 Software Implementation

The ROBOEAR requires the amplitude of each channel to be above a minimum value to ensure there is sufficient signal-to-noise-ratio (SNR). The default minimum is 31 out of the maximum 8-bit amplitude of 255. When the host tells the ROBOEAR to acquire and process data the process works like this:

1. Acquire packed raw data
2. Unpack the data into the 3 channel buffers
3. Compute the average and amplitude for each buffer
4. If any channel's amplitude is < the minimum goto 1
5. Compute the amplitude bearing tangent
   a. Look up the amplitude bearing angle
6. Convert the channel buffers to signed binary values
7. Compute the cross-correlation function for a range of Lag values for each baseline
   a. Find the maximum cross-correlation value and matching Lag value for each baseline
   b. Look up the bearing angle for the three Lag values
8. Send the data to the host

The processing power required for all this is a severe challenge for an 8-bit microcontroller. For example, calculating the cross-correlation values involves three baselines, thirteen lags per baseline (-6 to +6) and 115 points per lag (the fully overlapped portion of each channel's 127 data points). That means it takes 4485 eight-bit multiplications and 24-bit additions just to do the cross-correlation. In spite of the processing demand, if the amplitudes are okay in step 4, the ROBOEAR can repeat the process at least 4 times a second.

Cross-correlation works best with long time series, this produces very high and narrow peaks when the time series line up. One of the consequences of the limited memory size is the relatively short (127 samples) time series from the three channels produce broad peaks in the cross-correlation functions. Figure 3.4 is a real example from the ROBOEAR.

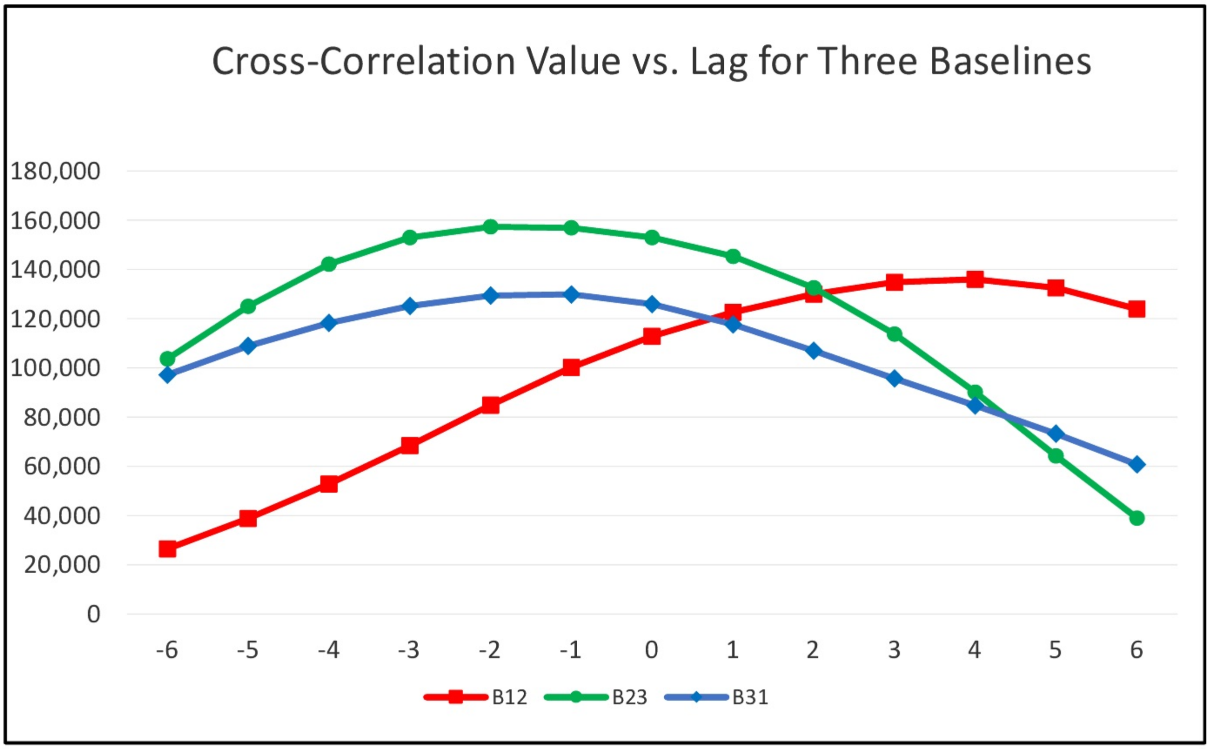


Figure 3.4

Because the peaks shown in Figure 3.4 are so flat the position of the maximum can easily shift ±1 lag from one data set to the next, solely due to noise. This limits the accuracy of the computed bearings from each baseline and means that even though I could compute precise bearing angles they would not be very accurate. Therefore I decided to have the ROBOEAR report bearings as sectors. Using sectors also simplifies the process of reducing the cross-correlation ambiguities to a single solution. In fact, the only thing we need to know is if the maximum cross-correlation occurred at a positive lag, zero, or a negative lag. Figure 3.5 and Table 3.5 show how this works. For example, consider a source at 90°. The maximum cross-correlation for any source between 0° and 180° should be found at a positive lag for baseline 1-2 (color coded red). For baseline 2-3 (green) the max cross-correlation should be at a negative lag; the same is true for baseline 3-1 (blue). Thus if the order of the sign of the lags for the three baselines is (+, -, -) then the source is in the sector centered at 90°. But what happens if max cross-correlation for one of the baselines has a lag of zero? A lag of zero means the source is on a line perpendicular to the baseline. For the example above, if the lag for baseline 3-1 was zero then the order would be (+,-,0) and the solution would be the sector centered at 120°.

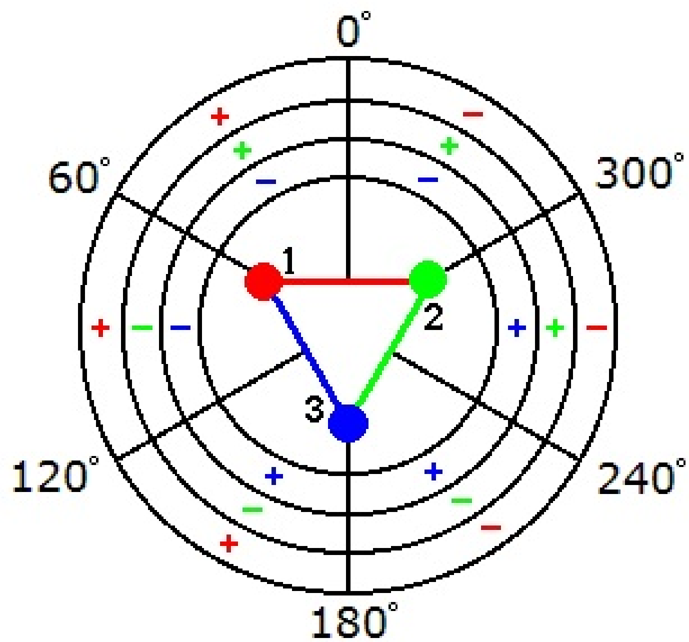


Figure 3.5

Sector Center	Baseline		
	1-2	2-3	3-1
0°	0	+	-
30°	+	+	-
60°	+	0	-
90°	+	-	-
120°	+	-	0
150°	+	-	+
180°	0	-	+
210°	-	-	+
240°	-	0	+
270°	-	+	+
300°	-	+	0
330°	-	+	-

Table 3.5

There are twelve 30° sectors shown in Table 3.5, each one is identified by a unique sequence. However there are more than 12 possible sequences - suppose the lag sequence was (+,+,+)? The source would need to be on the left side of all three baselines, an impossible situation. This can happen in the presence of noise and echos. When a unique solution is impossible the bearing angle is returned as 0xFF in BAM or its equivalent decimal value of 358.

The performance of the ROBOEAR in a real world situation depends on so many external factors it is impossible to predict. I ran a series of indoor tests using a radio at a distance of 1.5 meters as the sound source. Because of the way the amplitude bearing is calculated it will always yield a solution. My results showed a mean error of 28° with a standard deviation of 114°. The cross-correlation bearing gave valid results about 90% of the time, with the valid results having a mean error of 2.6° and a standard deviation of 26°. To me this implies averaging the last few valid cross-correlation bearings is probably the best way to use the ROBOEAR.

4.0 Software and Operating Modes

The ROBOEAR communicates with the host via SPI or UART (RS-232). The method of communication is determined by jumper J6. If J6 is open communication is via the UART, if it is jumpered it is via SPI.

See Appendices E and F for details on UART communications. See Appendix G for details on SPI communications.

4.1 Assembler Source Code

The assembly language source code for the standard ROBOEAR is included with the kit. The source code is well commented and highly modular. If you know PIC assembly language it should be easy to understand. If you want to learn more about PIC programming and the free Microchip Assembler (MPASM) consult some of the excellent resources on the Microchip web site (see Appendix H for the URL).

4.2 ROBOEAR Commands

The ROBOEAR boots up in "human" mode and displays the menu below. In human mode, when a command is entered the ROBOEAR echos the command. If the command is not valid then a NAK is returned and the menu is sent again. If the command is valid, an ACK is returned, the command is executed and the menu is sent again. If the ROBOEAR is connected to a controlling host computer its program should wait until it sees a "?" then send a "C" to the ROBOEAR to start "computer" mode. In computer mode commands are not echoed and the menu is not sent between commands.

```
  ROBOEAR, firmware 1.10
[E] Acquire one set of 8-bit data
[T] Acquire one set of 10-bit data
 [R] Send raw data in CSV format
[P] Acquire and process one set of 8-bit data
 [S] Send processing statistics
 [B] Send bearing angles
[M] Set minimum channel range
[H] Use Hex values for raw data
[D] Use Decimal values for raw data
[A] Use BAM format for bearings
[G] Use decimal degrees format for bearings
[C] Computer control mode, ESCape to exit
 ?
```

The commands are all single ascii characters. The channels are designated CH1, CH2, CH3. The following conventions apply to the command definitions:

# = ascii numeric character	( ) = single ascii byte
[/] = one of two possible ascii bytes	{ROBOEAR action}

The E, T and R commands are useful in computer mode when the host computer will be processing the raw audio data from the three channels. E and T will acquire a data set, that is 127 samples at 20 kHz from all three channels. R will send the unprocessed raw data to the host in the format specified by the H or D command.

Acquire one set of 8-bit data.

Host → ROBOEAR	[e/E]
ROBOEAR → Host	(ACK), set BUSY-RB3
	{acquire data}
ROBOEAR → Host	{clear BUSY-RB3}

Acquire one set of 10-bit data.

Host → ROBOEAR	[t/T]
ROBOEAR → Host	(ACK), set BUSY-RB3
	{acquire data}
ROBOEAR → Host	clear BUSY-RB3

Send the raw data from the preceding E or T command to the host.

Host → ROBOEAR	[r/R]
ROBOEAR → Host	(ACK), set BUSY-RB3 or (NAK)
	{send raw data to host as 127 lines where each line has three values: CH1,CH2,CH3}
ROBOEAR → Host	clear BUSY-RB3

The P, S and B commands are useful in human and computer mode. The P command acquires data until all the channels meet the minimum amplitude set by the M command. Then it processes the data and calculates bearings by the amplitude and cross-correlation methods. The S command reports some of the processing statistics and the B command reports the bearing angles in the format specified by the A or G command.

Acquire and process one set of 8-bit data.

Host → ROBOEAR	[p/P]
ROBOEAR → Host	(ACK), set BUSY-RB3
	{acquire data and process data}
ROBOEAR → Host	clear BUSY-RB3

Report statistics on the processed data in decimal format. Must follow P command.

Host → ROBOEAR	[s/S]
ROBOEAR → Host	(ACK), set BUSY-RB3 or (NAK)
	{send statistics as one line in the following format: Minimum dynamic range,CH1 range,CH2 range,CH3 range, CH1 average,CH2 average,CH3 average}
ROBOEAR → Host	clear BUSY-RB3

Report the computed amplitude and cross-correlation bearings. Must follow P or S command.

Host → ROBOEAR	[b/B]
ROBOEAR → Host	(ACK), set BUSY-RB3 or (NAK)
	{send bearings to host on one line in the following format:

Amplitude bearing,Cross-correlation bearing}
ROBOEAR → Host clear BUSY-RB3

The M command sets the minimum dynamic range (amplitude) required for each channel before the data will be used to calculate a bearing. This is the minimum used in step 4 of the flowchart shown for the P command in section 3.4. The higher the minimum value the louder the sound must be before the P command will return bearing angles. The M command has no effect on the E or T commands. Following the M character the value for the minimum must be sent as three ascii decimal characters, for example: “M048".

Set the minimum channel amplitude required for processing via the P command.
Host → ROBOEAR [m/M]###
ROBOEAR → Host (ACK), set BUSY-RB3 or (NAK)
{set the min_rng value}
ROBOEAR → Host clear BUSY-RB3

The H, D, A and G commands are used to set the numeric format to report raw data and computed bearings. At power on all formats default to decimal. Decimal data is always sent with leading zeros suppressed. Hexadecimal data is sent as 2 characters for 8-bit data and three characters for 10-bit data.

Internally, the ROBOEAR stores all raw data values from the channels in binary, so it takes less time to convert them to ASCII-hexadecimal than it does to ASCII-decimal representations. It is also faster for the host computer to convert ASCII-hex back to binary. Thus, transfer of raw data in ASCII-hex format may be the best choice if processing time is critical.

The ROBOEAR stores bearing angles in 8-bit Binary Angle Measure format. In BAM the msb represents 180 degrees and each lesser bit half of the higher bit. See the table at the right for the 8-bit weights. BAM values are sent in hexadecimal format. Reporting the bearings in decimal format involves multiplying, dividing and converting binary to bcd, so if speed is an issue BAM format will be quicker.

Bit	BAM Weight
7	180°
6	90°
5	45°
4	22.5°
3	11.25°
2	5.625°
1	2.8125°
0	1.40625°

Set hexadecimal as the numeric format for the R command to report raw data.
Host → ROBOEAR [h/H]
ROBOEAR → Host (ACK), set BUSY-RB
{set raw data format flag}
ROBOEAR → Host clear BUSY-RB3

Set decimal as the numeric format for the R command to report raw data.
Host → ROBOEAR [d/D]
ROBOEAR → Host (ACK), set BUSY-RB3
{clear raw data format flag}
ROBOEAR → Host clear BUSY-RB3

Set BAM as the numeric format for the B command to report bearing angles.

Host → ROBOEAR	[a/A]
ROBOEAR → Host	(ACK), set BUSY-RB3
	{set raw data format flag}
ROBOEAR → Host	clear BUSY-RB3

Set decimal as the numeric format for the B command to report bearing angles.

Host → ROBOEAR	[g/G]
ROBOEAR → Host	(ACK), set BUSY-RB3
	{clear raw data format flag}
ROBOEAR → Host	clear BUSY-RB3

The last commands are C, ESCape, and @.

Put the ROBOEAR in computer control mode.

Host → ROBOEAR	[c/C]
ROBOEAR → Host	(ACK), set BUSY-RB3
	{set computer control mode flag}
ROBOEAR → Host	clear BUSY-RB3

Put the ROBOEAR in human mode.

Host → ROBOEAR	[ESC]
ROBOEAR → Host	(ACK), set BUSY-RB3
	{clear computer control mode flag}
ROBOEAR → Host	clear BUSY-RB3

Do a software reset

Host → ROBOEAR	[@]
	{execute software reset}
ROBOEAR → Host	set BUSY-RB3

4.3 Error Messages

The ROBOEAR does not send error messages as such. Sometimes noise on the channels or other conditions will prevent the ROBOEAR from computing a valid bearing angle. The B command will return 0xFF for invalid bearings in BAM format and 358 for invalid bearings in decimal format.

5.0 ROBOEAR Circuit Board Assembly

5.1 Preparation

You will need the following tools:

> A low wattage soldering pencil, approximately 10 to 20 Watts.
> Flux core solder wire, organic flux core preferred.
> Lead benders.
> Lead/wire clippers.

Before beginning assemble, carefully check the ROBOEAR circuit board for shorted or incomplete traces and confirm all parts against the list in Appendix A. Known problems with circuit boards are shown in Appendix C, ROBOEAR circuit board errata.

5.2 Assembly Checklist

Check the value/type of each part as you assemble the board. Clip excess lead length from each component after it's soldered. See Appendix H for parts placement.

Insert and solder the low-profile sockets. Do not put the chips in the sockets.

____	U1	18-pin DIP
____	U10	8-pin DIP
____	U11	8-pin DIP
____	U20	8-pin DIP
____	U21	8-pin DIP
____	U30	8-pin DIP
____	U31	8-pin DIP
____	U40	16-pin DIP

Insert and solder the following components.

____	R10	2.2K, 1/8W, 5%	(red-red-red-gold)
____	R20	2.2K, 1/8W, 5%	(red-red-red-gold)
____	R30	2.2K, 1/8W, 5%	(red-red-red-gold)
____	R11	10K, 1/8W, 5%	(brown-black-orange-gold)
____	R12	10K, 1/8W, 5%	(brown-black-orange-gold)
____	R21	10K, 1/8W, 5%	(brown-black-orange-gold)
____	R22	10K, 1/8W, 5%	(brown-black-orange-gold)
____	R31	10K, 1/8W, 5%	(brown-black-orange-gold)
____	R32	10K, 1/8W, 5%	(brown-black-orange-gold)
____	R13	1M, 1/8W, 5%	(brown-black-green-gold)
____	R23	1M, 1/8W, 5%	(brown-black-green-gold)
____	R33	1M, 1/8W, 5%	(brown-black-green-gold)
____	R14	3.3K, 1/8W, 5%	(orange-orange-red-gold)
____	R24	3.3K, 1/8W, 5%	(orange-orange-red-gold)
____	R34	3.3K, 1/8W, 5%	(orange-orange-red-gold)
____	R15	33K, 1/8W, 5%	(orange-orange-orange-gold)
____	R25	33K, 1/8W, 5%	(orange-orange-orange-gold)
____	R35	33K, 1/8W, 5%	(orange-orange-orange-gold)

____	R40	1.1K, 1/85W, 5%	(brown-brown-red-gold)
____	R41	1.1K, 1/85W, 5%	(brown-brown-red-gold)
____	RN1	10k, 10-pin SIP, pin 1 goes in the square pad	

____	C1	0.1 uFd, radial, 0.1" lead spacing
____	C3	0.1 uFd, radial, 0.1" lead spacing
____	C4	0.1 uFd, radial, 0.1" lead spacing
____	C5	0.1 uFd, radial, 0.1" lead spacing
____	C13	0.1 uFd, radial, 0.1" lead spacing
____	C23	0.1 uFd, radial, 0.1" lead spacing
____	C33	0.1 uFd, radial, 0.1" lead spacing
____	C44	0.1 uFd, radial, 0.1" lead spacing
____	C2	100 uFd, axial, positive lead toward square pad
____	C6	100 nFd, radial, 0.2" lead spacing
____	C12	0.001 uFd, radial, 0.2" lead spacing
____	C22	0.001 uFd, radial, 0.2" lead spacing
____	C32	0.001 uFd, radial, 0.2" lead spacing
____	C10	1 uFd, radial, 0.2" lead spacing
____	C11	1 uFd, radial, 0.2" lead spacing
____	C20	1 uFd, radial, 0.2" lead spacing
____	C21	1 uFd, radial, 0.2" lead spacing
____	C30	1 uFd, radial, 0.2" lead spacing
____	C31	1 uFd, radial, 0.2" lead spacing
____	C40	1 uFd, radial, 0.2" lead spacing
____	C41	1 uFd, radial, 0.2" lead spacing
____	C42	1 uFd, radial, 0.2" lead spacing
____	C43	1 uFd, radial, 0.2" lead spacing

____	J1	DC power jack
____	J2	DB9 female connector
____	J3	3x1 jumper header
____	J4	3x1 jumper header
____	J5	2x2 jumper header
____	J6	2x1 jumper header
____	J7	6x1 jumper header
____	J8	4x2 jumper header

____	U2	TLE2426, TO-92 package
____	U3	78M05, TO-220 package

The last construction step is to clean the board. If you used organic core solder just rinse the board in warm water. If you used acid core solder try scrubbing it with an old toothbrush and rubbing alcohol.

5.3 Circuit Board Checkout

You will need a multimeter to check out the ROBOEAR circuitry. Place the ROBOEAR circuit board on an insulating surface. DO NOT install the integrated circuits yet.

Attach the negative lead of your voltmeter to a ground point, such as the negative side of C2. Plug-in your 9V wall transformer and connect it to J1 on the ROBOEAR. The supply voltage should measure at least 8 VDC on the back side of J1. Measure the 5 VDC supply at the +5V test point or the positive lead of C2. Measure the virtual ground voltage at the +2.5V test point. If there is a problem, disconnect the wall transformer and inspect the ROBOEAR circuit board. Be sure the 78M05 and TLE2426 regulators are not backwards.

With the wall transformer disconnected install the integrated circuits in their sockets. Be sure to note the position of pin 1.

____ U1 PIC16F1847
____ U10 MCP602
____ U11 MCP602
____ U20 MCP602
____ U21 MCP3001
____ U30 MCP3001
____ U31 MCP3001
____ U40 MAX232

6.0 Installation

6.1 Prepare and Mount the Microphones

The three electret microphones connect to the ROBOEAR at J10, J20 and J30. Pin 1 on each connector is the positive input and pin 2 is ground. On most electret microphones the positive output is pin 1 and the ground reference is pin 2. Pin 1 of the microphone should be wired to pin 1 of its respective connector and pin 2 of the microphone to pin 2 of the connector. Be sure to make the connecting wires long enough to arrange the microphones in the required pattern.

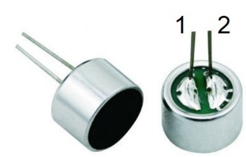


Figure 6.1

The microphones need to be arranged in the correct order, in the form of an equilateral triangle 10 centimeters on a side. This can also be thought of as mounting them 120 degrees apart on a circle with a radius of 5.7 centimeters, see Appendix D. The order of the microphones (numbered according to the analog channel to which they are wired) is critical to determining the direction to a sound source. Triangles of other dimensions are possible if you are processing the raw data on an external processor. The diagram at the right shows how the detected direction corresponds to the proper arrangement of the microphones. The plane of the microphones should be parallel to the floor.

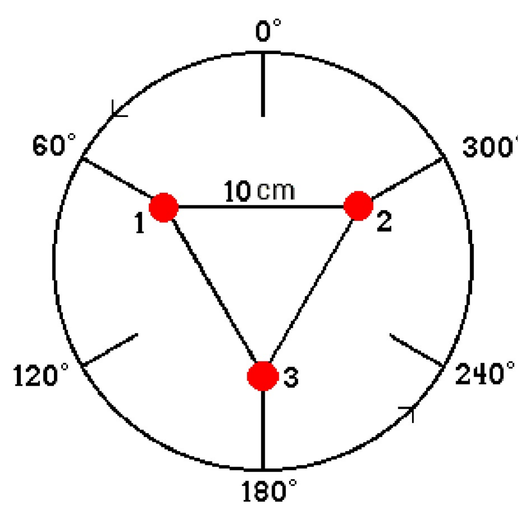


Figure 6.2

7.0 Customization

The ROBOEAR circuit board does not need to be installed in the CM5-125 case suggested by this manual. Mounting of the ROBOEAR and microphones is entirely up to the user. If your application has 5 VDC available you can wire that directly to the circuit board and eliminate J1, C1 and the 5V regulator (U3). If you communicate with the host via SPI (J8) or TTL level RS232 (J6) then you can eliminate J2, U40 and all the 4X numbered resistors and capacitors.

If you use the ROBOEAR as the data acquisition system for a computationally powerful host you can try lots of things with the raw data supplied by the ROBOEAR. You could:

- Change the size (microphone baseline) of the equilateral triangle.
- Change the arrangement of the microphones, for example a right triangle.
- Experiment with ways to improve the calculation of bearing angle.

However, as long as you use the ROBOEAR firmware you will be limited to time series 127 bytes in length sampled at 20 kHz.

The assembly source code for the ROBOEAR comes with the kit. Feel free to modify it in any way you like, to make the ROBOEAR do anything the hardware capable of. In particular, if you plan to use the ROBOEAR only as an acquisition system, you should be able to modify the code to change the number of samples and sample rate, perhaps even acquire samples in real-time.

APPENDIX A

ROBOEAR-1 CIRCUIT BOARD PARTS LIST

Quantity	Part	Reference	
8	0.1uFd, 0.1" LS	C1, C3, C4, C5, C13, C23, C33, C44	
1	100uFd, axial	C2	
1	100nFd, 0.2" LS	C6	
10	1.0uFd, 0.2" LS	C10, C11, C20, C21, C30, C31, C40-43	
3	0.001uFd, 0.2" LS	C12, C22, C32	
3	2.2K, 1/8W, 5%	R10, R20, R30	(red-red-red-gold)
6	10K, 1/8W, 5%	R11, R12, R21, R22, R31, R32	(brown-black-orange-gold)
3	1M, 1/8W, 5%	R13, R23, R33	(brown-black-green-gold)
3	3.3K, 1/8W, 5%	R14, R24, R34	(orange-orange-red-gold)
3	33K, 1/8W, 5%	R15, R25, R35	(orange-orange-orange-gold)
2	1.1K, 1/8W, 5%	R40, R41	(brown-brown-red-gold)
1	10K, 10 pin SIP	RN1	
1	PIC16F1847	U1	
1	TLE2426CLP	U2	
1	LM78M05	U3	
3	MCP602	U10, U20, U30	
3	MCP3001	U11, U21, U31	
1	MAX232ACPE	U40	
1	18-DIP socket	U1	
1	16-DIP socket	U40	
6	8-DIP socket	U10, U11,U20, U21, U30, U31	
1	DC power jack	J1	
1	DB9 female connector	J2	
6	2 pin header	J3, J4, J6, J10, J20, J30	
1	4 pin header, 2x2	J5	
1	6 pin header, 1x6	J7	
1	8 pin header, 2x4	J8	
1	ROBOEAR-1 circuit board		
4	Self-tapping screws for mounting the printed circuit board in a plastic case.		
3	Jumpers		
3	Electret microphones		

APPENDIX B

ROBOEAR FULL SYSTEM PARTS

The ROBOEAR-1 kit sold by Lucid Technologies includes all the parts necessary for a fully functional ROBOEAR system with the exception of wiring. The list below shows both required and optional parts.

Required for all ROBOEAR systems:

Wiring to the three microphones.

Required if on-board 5V regulator is used:

Wall transformer, 9VDC, 100 mA or more, coaxial power plug, 5.5mm O.D., 2.1mm I.D., center positive, 1 required

Jameco Electronics	www.jameco.com	800-831-4242
	Part# 100845	
Mouser Electronics	www.mouser.com	800-346-6873
	Part# 553-WDU9-100	

Optional parts:

Plastic case (optional) 1 required

PacTec	www.pactecenclosures.com	610-361-4200
	Model CM5-125	
Simco	www.simcobox.com	800-780-9090
	Model 150X5, Challenger series	
Lucid Technologies	http://www.lucidtechnologies.info/case-10.htm	
	CASE-001	

APPENDIX C

ROBOEAR CIRCUIT BOARD ERRATA

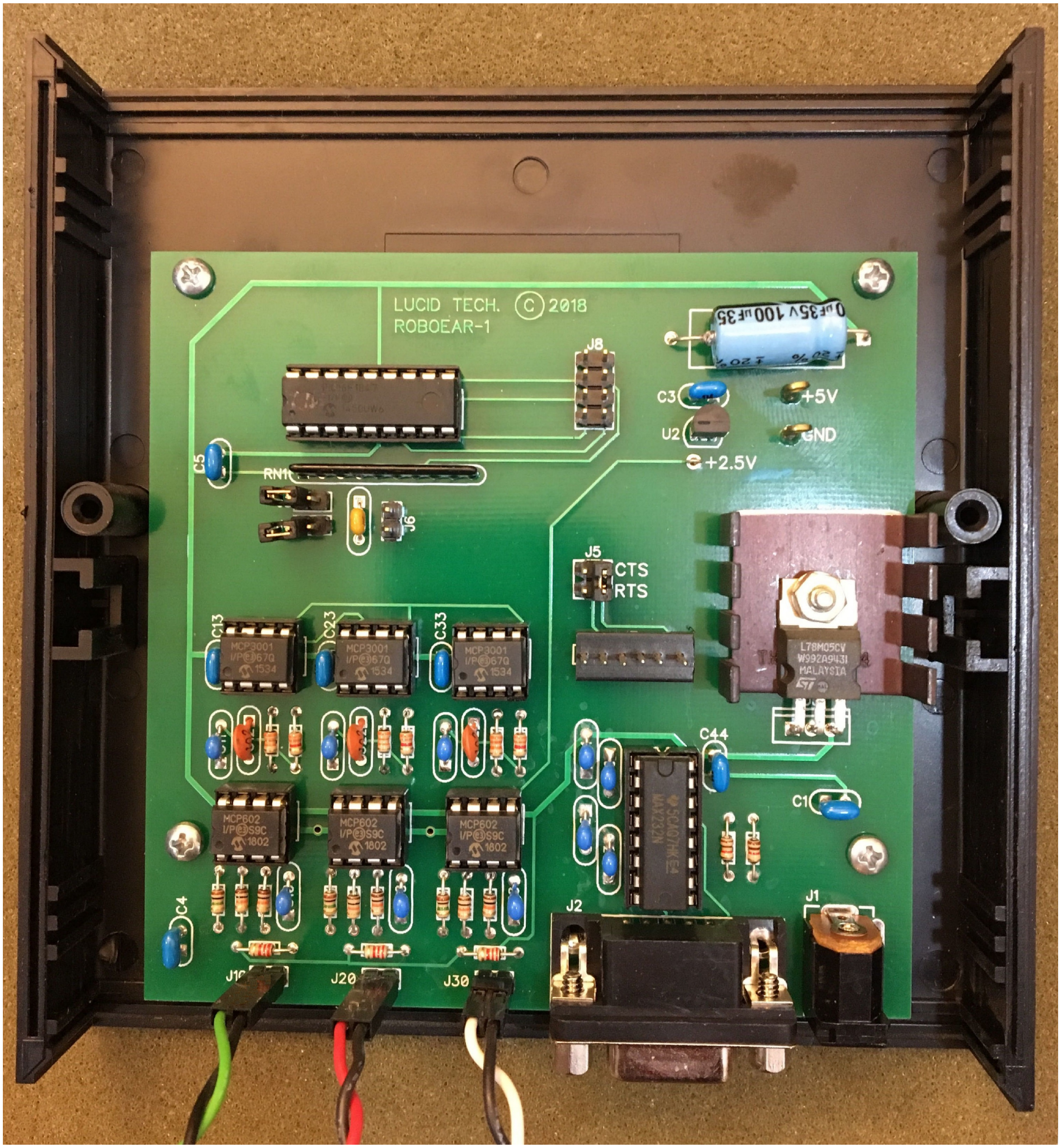
ROBOEAR-1

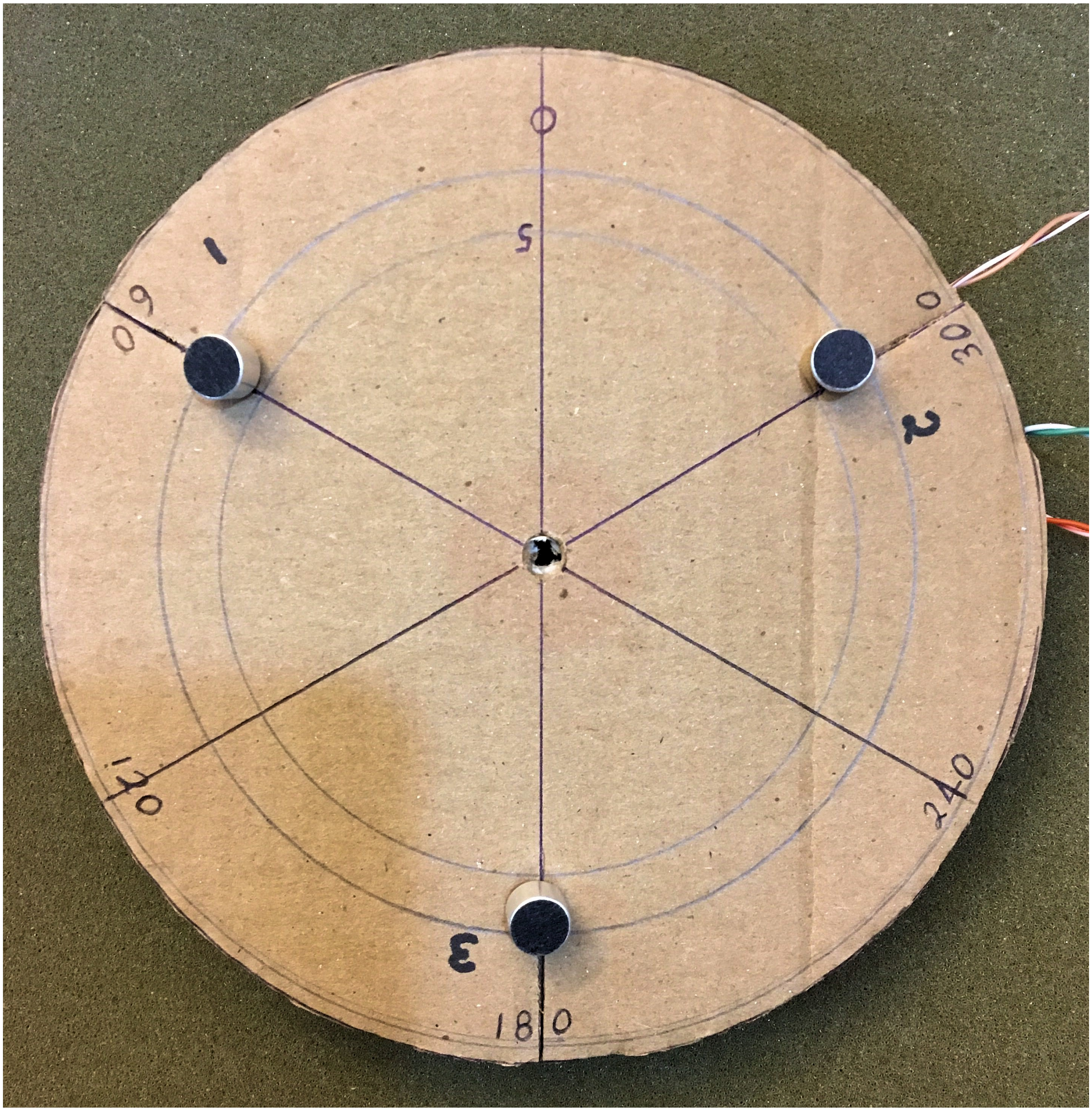
The GND test point seen on the top of the board near the upper-right corner is not connected to anything. It should be connected to the ground plane on the bottom of the board.

APPENDIX D

CHASSIS DETAILS

This picture shows the ROBOEAR-1 mounted in the recommended CM5-125 case.





The prototype ROBOEAR mounted the microphones on a cardboard disk.

APPENDIX E

RS-232 SERIAL INTERFACE CONNECTOR

The table below shows the most commonly implemented signals and their pin assignments in accordance with RS-232D. Host computers are usually DTE (Data Terminal Equipment) and modems are DCE (Data Communications Equipment). Note that circuits are named from the point of view of the DTE. For example, circuit BB (receive data) is actually data transmitted by the DCE. DCE devices originally used a 25 pin, female, D connector.

Pin	**Circuit**	**Description**	**Direction**
1	AA	Protective ground, PG	n/a
2	BA	Transmit data, TD	to DCE
3	BB	Receive data, RD	from DCE
4	CA	Request to send, RTS Ready to receive, RTR	to DCE
5	CB	Clear to send, CTS	from DCE
6	CC	Data set ready, DSR	from DCE
7	AB	Signal ground, SG	n/a
8	CF	Data carrier detect, DCD	from DCE
20	CD	Data Terminal ready, DTR	to DCE
22	CE	Ring indicator, RI	from DCE

In recent years all personal computer have migrated to the use of a 9 pin, male, D connector instead of the 25 pin connector. The pin assignments for such a DTE device are shown below.

Pin	**Circuit**	**Description**	**Direction**
1	CF	Data carrier detect, DCD	from DCE
2	BB	Receive data, RD	from DCE
3	BA	Transmit data, TD	to DCE
4	CD	Data Terminal ready, DTR	to DCE
5	AB	Signal ground, SG	n/a
6	CC	Data set ready, DSR	from DCE
7	CA	Request to send, RTS Ready to receive, RTR	to DCE
8	CB	Clear to send, CTS	from DCE
9	CE	Ring indicator, RI	from DCE

All Lucid Technologies products are designed as DCE devices. They use a 9 pin, female, D connector that is directly compatible with 9 pin COM ports found on personal computers. The pin assignments for this connector are shown below.

Pin	Circuit	Description	Direction
1	CF	Data carrier detect, DCD	from Roboear
2	BB	Receive data, RD	from Roboear
3	BA	Transmit data, TD	to Roboear
4	CD	Data Terminal ready, DTR	to Roboear
5	AB	Signal ground, SG	n/a
6	CC	Data set ready, DSR	from Roboear
7	CA	Request to send, RTS Ready to receive, RTR	to Roboear
8	CB	Clear to send, CTS	from Roboear
9	CE	Ring indicator, RI	from Roboear

RS-232 logic and voltage levels

Data circuits	Circuits status	RS-232 Voltage
0 (space)	Asserted = ON	+3 to +15 V
1 (mark)	Deasserted = OFF	-15 to -3 V

APPENDIX F

RS-232 COMMUNICATIONS SETUP

For the ROBOEAR to communicate via UART (RS-232) jumper J6 must be open.

Newer PCs may not possess a COM port but communication is still possible via a USB-to-Serial adapter. Connector J7 is a 6 pin header set up to accept an FTDI TTL-232R-5V, USB-to-Serial adapter. If you are interfacing the RoboEar with a Raspberry or Arduino you can use J7 to directly access the 5V UART signals.

Table 1. J7 pinout

1	2	3	4	5	6
GND	CTS	N/C	TXD	RXD	RTS

Pin 3 on the FTDI TTL-232R-5V USB-to-Serial adapter is VCC. Because J7 pin 3 is not connected to the ROBOEAR VCC bus the ROBOEAR can not be powered from the adapter unless the board is modified by connecting pin 3 to the ROBOEAR +5V bus.

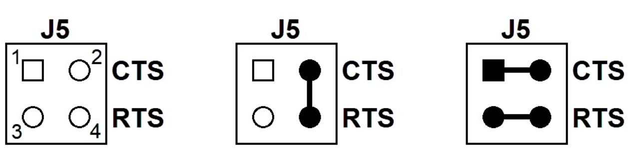
The PIC16F1847 UART is set to operate with 8 data bits, no parity and one stop bit. Baud rate is selected via jumpers J3 and J4.

Table 2. Baud rate jumper settings

J4 (RA4)	**J3 (RA3)**	**Baud Rate**
Open	Open	9600
Open	Jumpered	19200
Jumpered	Open	38400
Jumpered	Jumpered	76800

The RS-232 control lines for the Roboear are configured as follows:

- DTR is not connected and thus is ignored.
- DSR and DCD are hard-wired to the ON condition at all times.
- Depending on how J5 is jumpered:
  - RTS may be received, buffered, and looped back to the host as CTS - thus CTS tracks RTS.
  - RTS may be received and sent to RB4, but is ignored by the PIC.
  - CTS may be jumpered to the BUSY signal (RB3) on J5-1. BUSY=1 indicates ROBOEAR is acquiring or processing data and is not ready for a command from the host. BUSY=0 indicates the Roboear is waiting to receive a command from the host. Use this option if the host implements flow control.



A B C

A = J5 pinout
B = CTS follows RTS
C = RTS to RB4 and CTS from RB3.

APPENDIX G

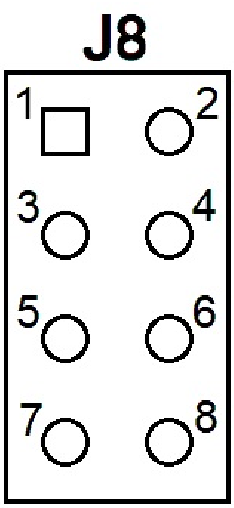
SERIAL PERIPHERAL INTERFACE (SPI) COMMUNICATIONS

For the ROBOEAR to communicate via SPI jumper J6 must be present. In SPI communication mode the expected clock polarity is controlled CKP, which is determined by jumper J3; if J3 is open then CKP=1. The expected active edge of the clock pulse is set by CKE, which is determined by jumper J4; If J4 is open then CKE=1. See section 25.2.4 SPI SLAVE MODE of the PIC16F1847 data sheet for details on four SPI modes determined by CKP and CKE.

SPI architecture requires a Master and one or more Slaves in an SPI bus. The ROBOEAR is configured as a Slave device so the host must be configured as a Master. The Master controls all data transfers, including, via the SCK signal, the transfer rate. Because the SPI bus gives the Slave has no way to indicate it is ready to receive or transmit, other signal lines must be used for those purposes.

RB3 is the BUSY output. BUSY=1 indicates ROBOEAR is acquiring or processing data and is not ready for a command from the host. BUSY=0 indicates the ROBOEAR is waiting to receive a command from the host. Any communications sent by the host while BUSY is high will be ignored. BUSY is pin 7 on J8.

RB0 is the Byte ReaDY (BRDY) output. BRDY=1 indicates the ROBOEAR has a data byte ready to transmit to the host. When the host sees this and initiates a data transfer the ROBOEAR will see the Slave Select go low and return BRDY to zero. BRDY is pin 8 on J8.



J8 pin	Signal
1	MISO = Master In, Slave Out data, from ROBOEAR
2	Not connected on ROBOEAR, usually Vcc
3	SCK = Serial clock signal, from host
4	MOSI = Master Out, Slave In data, from host
5	SS = Slave Select (active low) from host
6	Ground
7	BUSY (RB3), from ROBOEAR
8	BRDY (RB0), from ROBOEAR

APPENDIX H

REFERENCES

Texas Instruments (www.ti.com)
TLE2426 Data Sheet, SLOS098D, rail-splitter, precision virtual ground
MAX323 Data Sheet, SLLS047M, dual EIA-232 driver/receiver

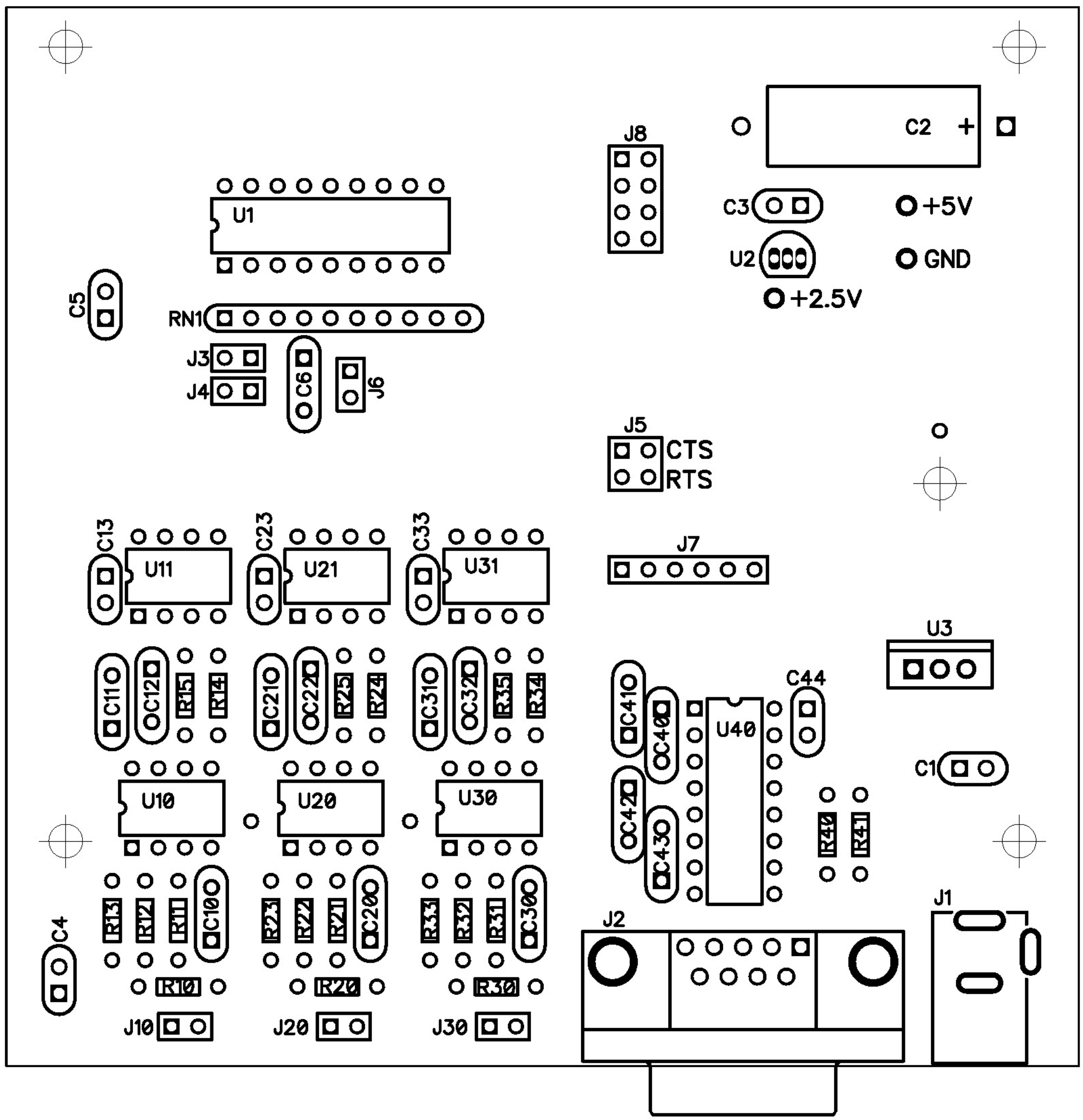
Microchip (www.microchip.com)
PIC16F184 Data Sheet, DS400011453E, enhanced mid-range 8-bit microcontroller
MCP602 Data Sheet, DS21314F, dual single-supply op-amp
MCP3001 Data Sheet, DS21293C, 10-bit A/D converter with SPI interface

Explanations of correlation
https://www.youtube.com/watch?v=_r_fDlM0Dx0
https://www.youtube.com/watch?v=L6YJqhbsuFY
https://www.youtube.com/watch?v=RO8s1TrElEw

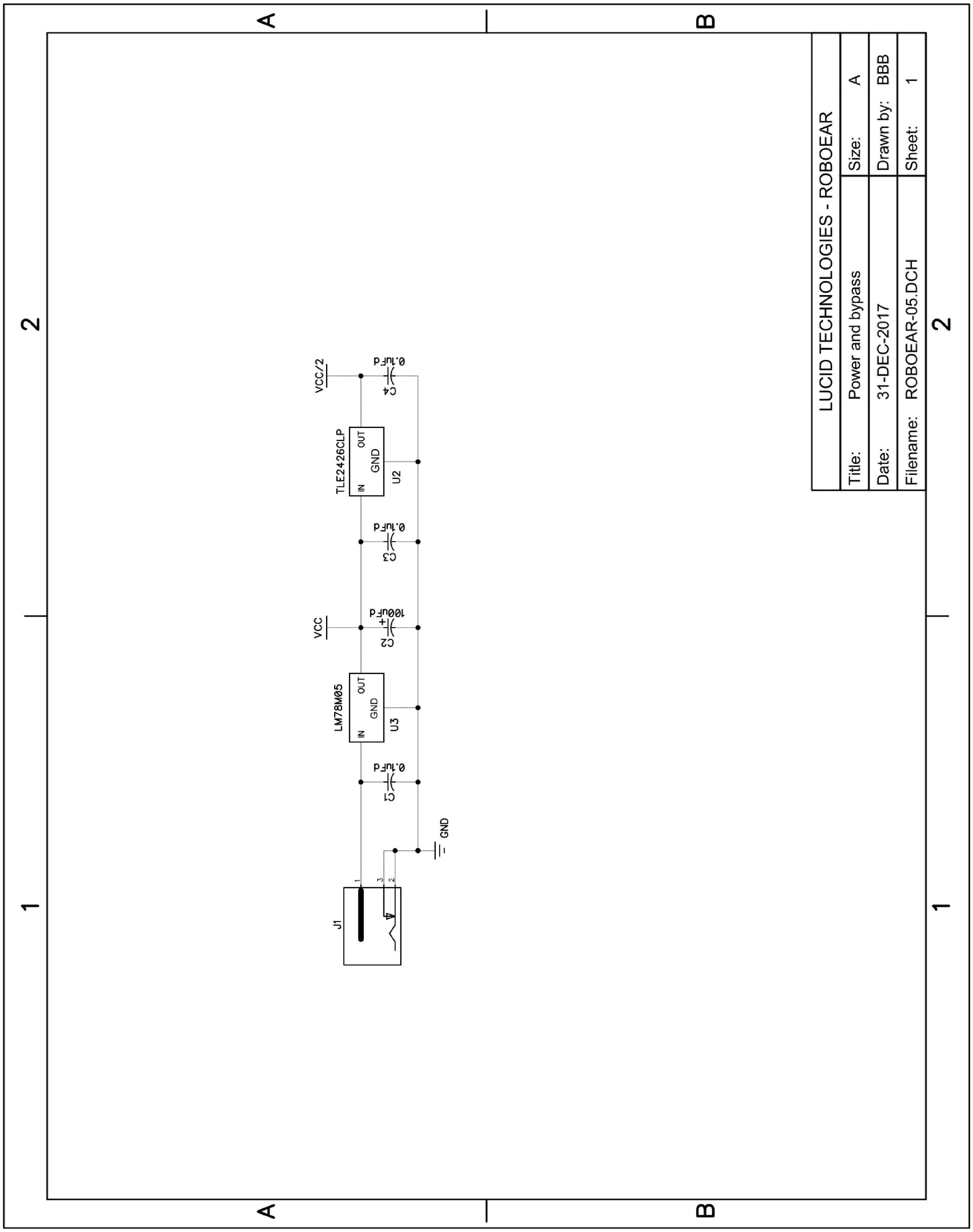
Lucid Technologies' home page
http://www.lucidtechnologies.info/index.htm

APPENDIX I

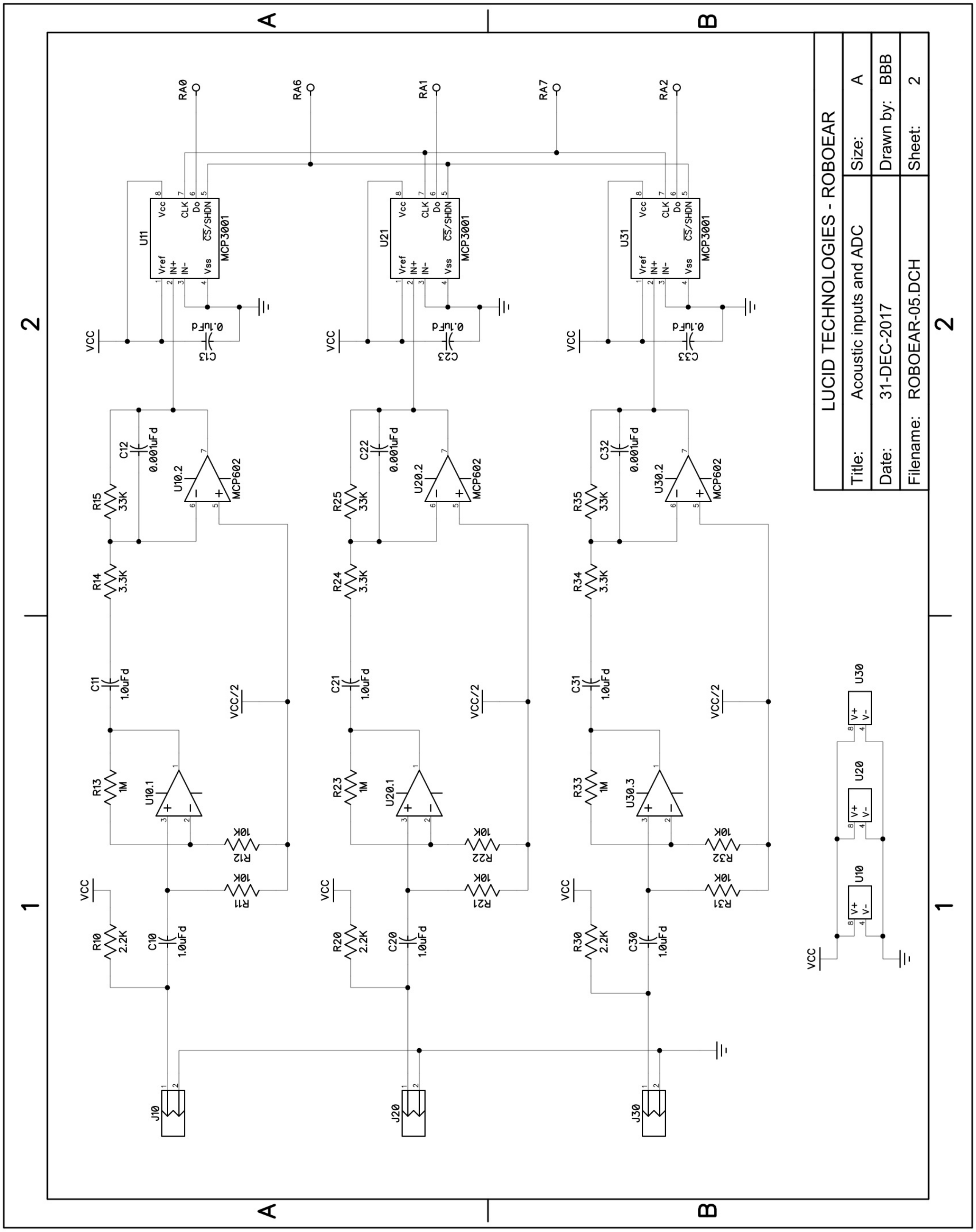
ROBOEAR-1 CIRCUIT BOARD



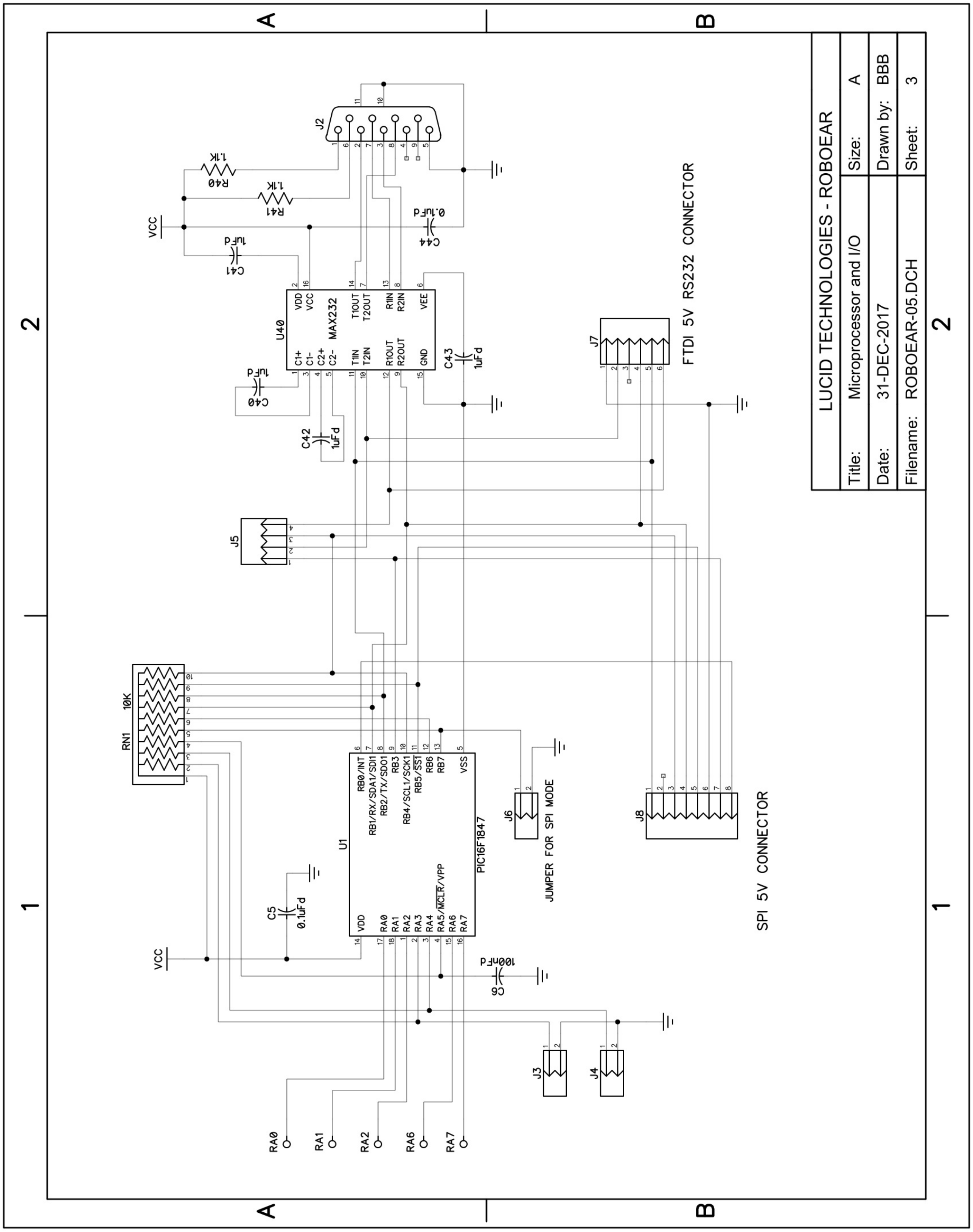
APPENDIX J


LUCID TECHNOLOGIES - ROBOEAR
Title: Power and bypass
Size: A
Date: 31-DEC-2017
Drawn by: BBB
Filename: ROBOEAR-05.DCH
Sheet: 1
J1
GND
C1 0.1uFd
LM78M05
U3
IN GND OUT
VCC
C2 100uFd
C3 0.1uFd
TLE2426CLP
U2
C4 0.1uFd
VCC/2
A
B
1
2

APPENDIX J


LUCID TECHNOLOGIES - ROBOEAR
Title: Acoustic inputs and ADC
Date: 31-DEC-2017
Filename: ROBOEAR-05.DCH
Size: A
Drawn by: BBB
Sheet: 2
RA0
RA6
RA1
RA7
RA2
U11
U21
U31
MCP3001
U10.2
U20.2
U30.2
U10.1
U20.1
U30.3
MCP602
J10
J20
J30
VCC
VCC/2
U10
U20
U30

APPENDIX J


LUCID TECHNOLOGIES - ROBOEAR
Title: Microprocessor and I/O
Size: A
Date: 31-DEC-2017
Drawn by: BBB
Filename: ROBOEAR-05.DCH
Sheet: 3
FTDI 5V RS232 CONNECTOR
SPI 5V CONNECTOR
JUMPER FOR SPI MODE
U40
MAX232
U1
PIC16F1847
J2
J3
J4
J5
J6
J7
J8
RN1 10K
R40 1.1K
R41 1.1K
C40 1uFd
C41 1uFd
C42 1uFd
C43 1uFd
C44 0.1uFd
C5 0.1uFd
C6 100nFd
VCC
RA0
RA1
RA2
RA6
RA7