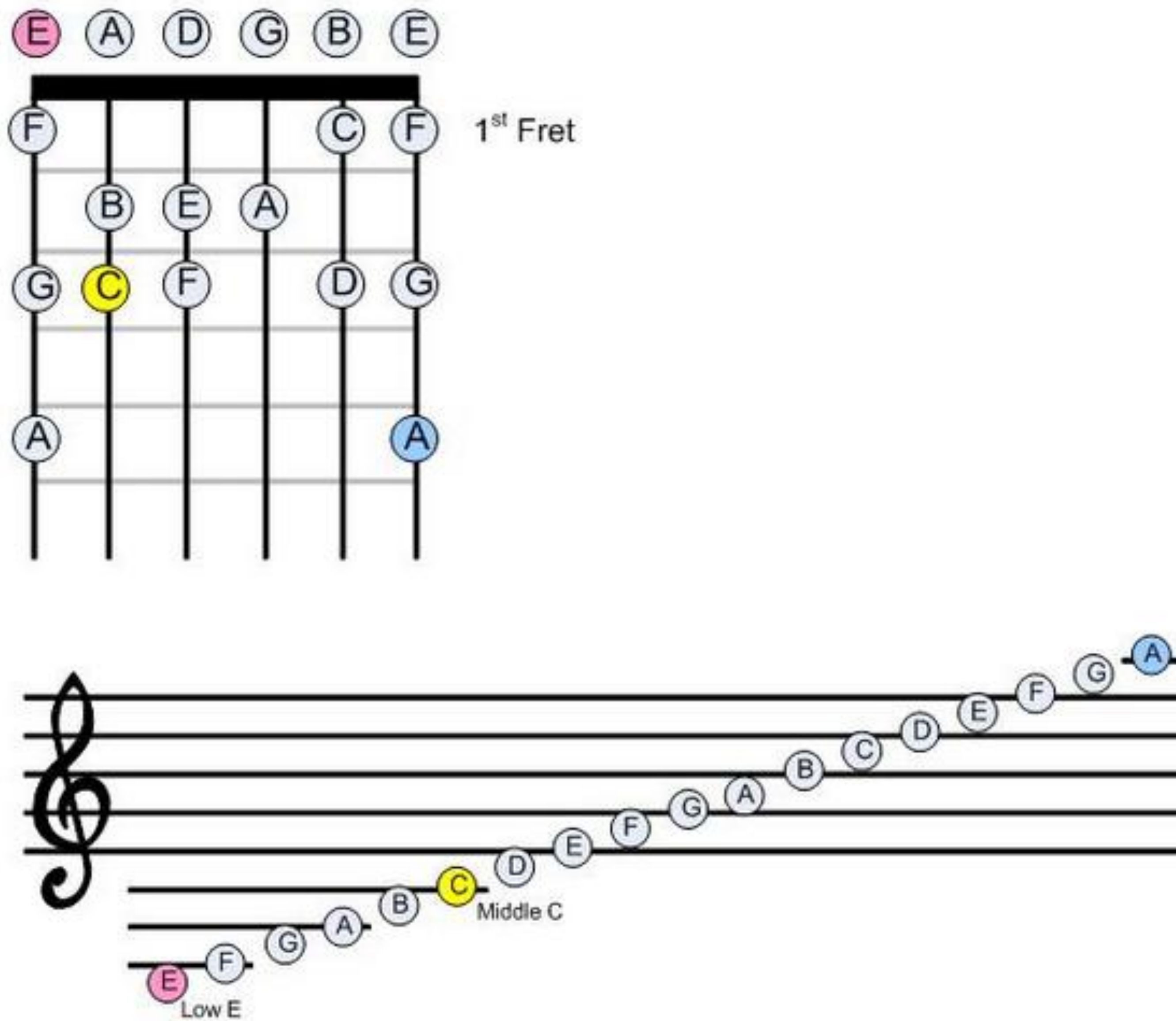
IDR: Smart Guitar

By: Smart Guitar Group

Basic Notes



E A D G B E
F C F
1st Fret
B E A
G C F D G
A A
E F G A B C D E F G A B C D E F G A
Middle C
Low E

Note	Frequency (Hz)	Wavelength (cm)
C_0	16.35	2100.
$C^{\#}_0/D^{b}_0$	17.32	1990.
D_0	18.35	1870.
$D^{\#}_0/E^{b}_0$	19.45	1770.
E_0	20.60	1670.
F_0	21.83	1580.
$F^{\#}_0/G^{b}_0$	23.12	1490.
G_0	24.50	1400.
$G^{\#}_0/A^{b}_0$	25.96	1320.
A_0	27.50	1250.
$A^{\#}_0/B^{b}_0$	29.14	1180.
B_0	30.87	1110.
C_1	32.70	1050.
$C^{\#}_1/D^{b}_1$	34.65	996.
D_1	36.71	940.
$D^{\#}_1/E^{b}_1$	38.89	887.
E_1	41.20	837.
F_1	43.65	790.
$F^{\#}_1/G^{b}_1$	46.25	746.
G_1	49.00	704.
$G^{\#}_1/A^{b}_1$	51.91	665.
A_1	55.00	627.
$A^{\#}_1/B^{b}_1$	58.27	592.
B_1	61.74	559.
C_2	65.41	527.
$C^{\#}_2/D^{b}_2$	69.30	498.
D_2	73.42	470.
$D^{\#}_2/E^{b}_2$	77.78	444.
E_2	82.41	419.
F_2	87.31	395.
$F^{\#}_2/G^{b}_2$	92.50	373.
G_2	98.00	352.
$G^{\#}_2/A^{b}_2$	103.83	332.
A_2	110.00	314.
$A^{\#}_2/B^{b}_2$	116.54	296.
B_2	123.47	279.
C_3	130.81	264.
$C^{\#}_3/D^{b}_3$	138.59	249.
D_3	146.83	235.
$D^{\#}_3/E^{b}_3$	155.56	222.
E_3	164.81	209.
F_3	174.61	198.
$F^{\#}_3/G^{b}_3$	185.00	186.
G_3	196.00	176.
$G^{\#}_3/A^{b}_3$	207.65	166.
A_3	220.00	157.
$A^{\#}_3/B^{b}_3$	233.08	148.
B_3	246.94	140.
C_4	261.63	132.
$C^{\#}_4/D^{b}_4$	277.18	124.
D_4	293.66	117.
$D^{\#}_4/E^{b}_4$	311.13	111.
E_4	329.63	105.
F_4	349.23	98.8
$F^{\#}_4/G^{b}_4$	369.99	93.2
G_4	392.00	88.0
$G^{\#}_4/A^{b}_4$	415.30	83.1
A_4	440.00	78.4
$A^{\#}_4/B^{b}_4$	466.16	74.0
B_4	493.88	69.9
C_5	523.25	65.9
$C^{\#}_5/D^{b}_5$	554.37	62.2

- The basic formula for the frequencies of the notes of the equal tempered scale is given by
  $f_n = f_0 * (a)^n$
  where
  f_0 = the frequency of one fixed note which must be defined. A common choice is setting the A above middle C (A_4) at f_0 = 440 Hz.
  n = the number of half steps away from the fixed note you are. If you are at a higher note, n is positive. If you are on a lower note, n is negative.
  f_n = the frequency of the note n half steps away.
  $a = (2)^{1/12}$ = the twelth root of 2 = the number which when multiplied by itself 12 times equals 2 = 1.059463094359...

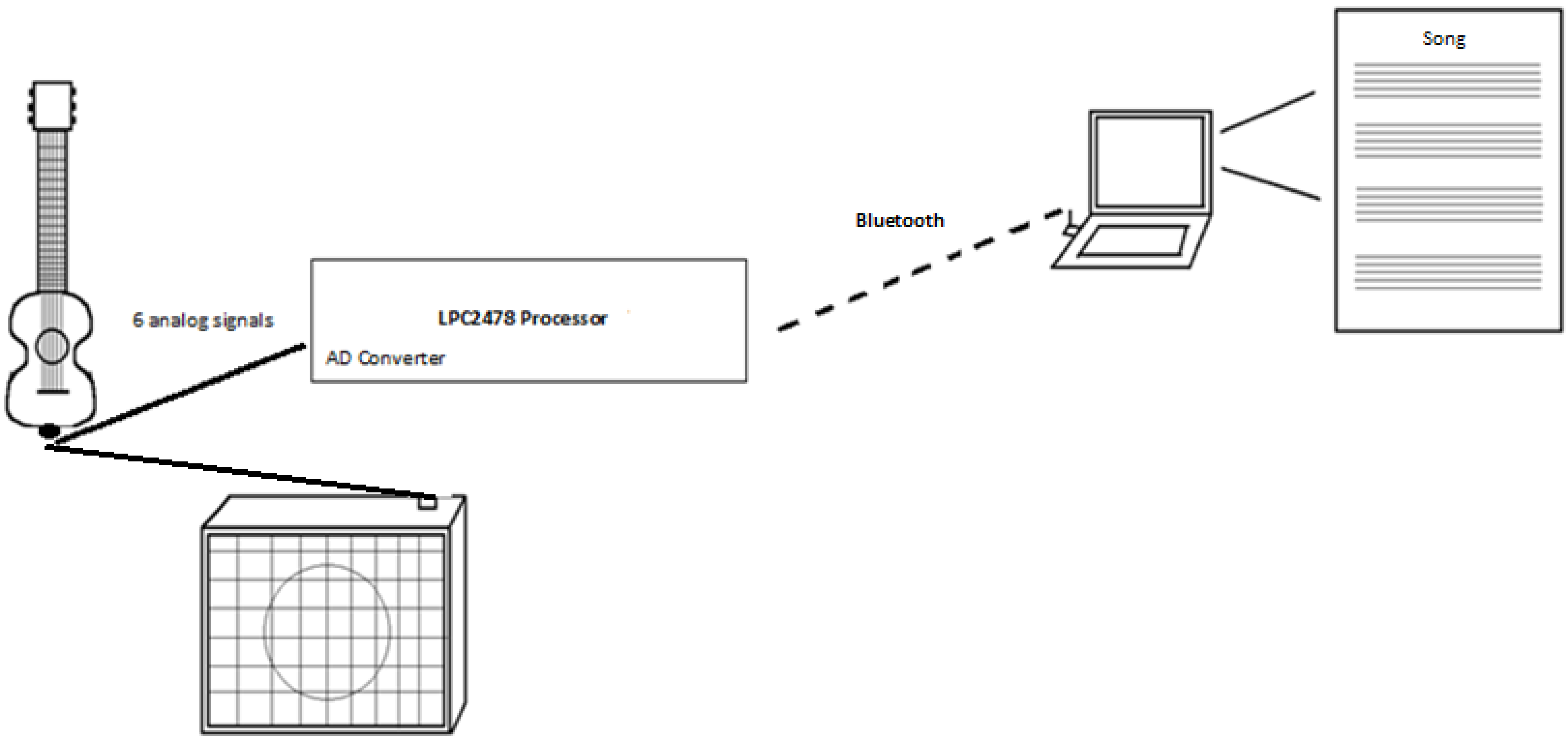
  The wavelength of the sound for the notes is found from
  $W_n = c/f_n$
  where W is the wavelength and c is the speed of sound. The speed of sound depends on temperature, but is approximately 345 m/s at "room temperature."

  Examples using A_4 = 440 Hz:

- C_5 = the C an octave above middle C. This is 3 half steps above A_4 and so the frequency is
  $f_3 = 440 * (1.059463..)^3 = 523.3$ Hz
  If your calculator does not have the ability to raise to powers, then use the fact that
  $(1.059463..)^3 = (1.059463..)*(1.059463..)*(1.059463..)$
  That is, you multiply it by itself 3 times.

  Middle C is 9 half steps below A_4 and the frequency is:
  $f_{-9} = 440 * (1.059463..)^{-9} = 261.6$ Hz
  If you don't have powers on your calculator, remember that the negative sign on the power means you divide instead of multiply. For this example, you divide by

Large Scale Diagram



Product Purposes

- To make a useful tool to enable guitarists to write there music without having to be classically trained.
- A faster more convenient tool for professional guitarists to be able to record their work and distribute it more readily.
- It acts as a teaching tool for beginner guitarists to be able to see visually the notes that they are playing and demonstrates the proper way to write music.
- Since music scores are how most music is written around the world, it takes away the unconventional need for tabs , and places guitar on the same spectrum of any other instrument.

Function

- Our project will take a note or notes from a guitar and convert the 6 analog signals from our Roland pickup to 6 digital signals by using the analog to digital converter on our processor.
- At the processor we will be interpreting the 6 digital signals using FFT algorithms and sending through a Bluetooth module where there will be two modes. A streaming mode that will take the signal and one can see the notes being played as the user plays them. The other mode will be a file transfer mode. Where after the user is finished playing the song a file will be transferred to the PC that can be opened. The Bluetooth module will communicate with the PC application where the operation mode will be chosen.
- The Bluetooth will be picked up by the PC and there will be a software program that will take the data package and transmit this into readable notes on a music score.

Critical Elements

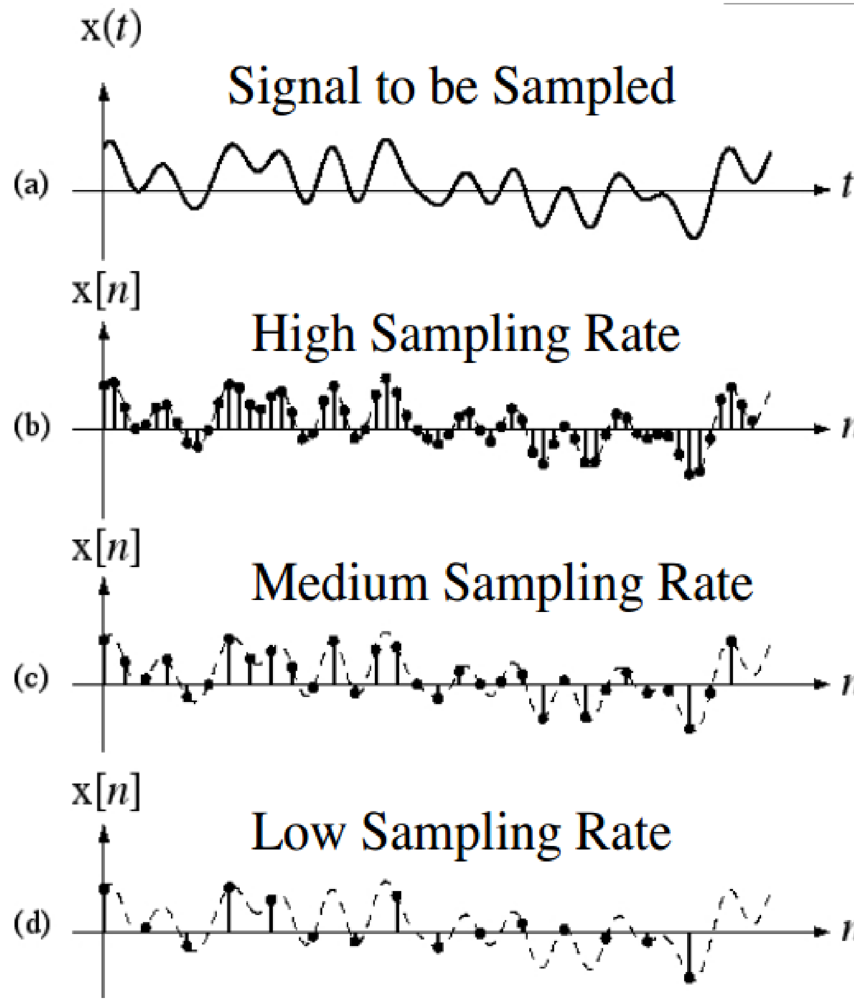
- One of the critical elements of our project is the actual interpreting the 6 signals that are coming from the guitar. If the FFT algorithm is incorrect or are algorithm of corresponding the signal to a readable note is off then the whole project is a bust.
- We might also have issues when it comes to sampling and how fast we can sample. Can the user play faster than the time it takes to sample?
- Another crucial part of our design is the Bluetooth connection. If we are not able to transmit the digital signals as a data package through our Bluetooth module than there will be no way for the signals to be written as notes.

Discrete Fourier Transform

- Analog Signal

↓

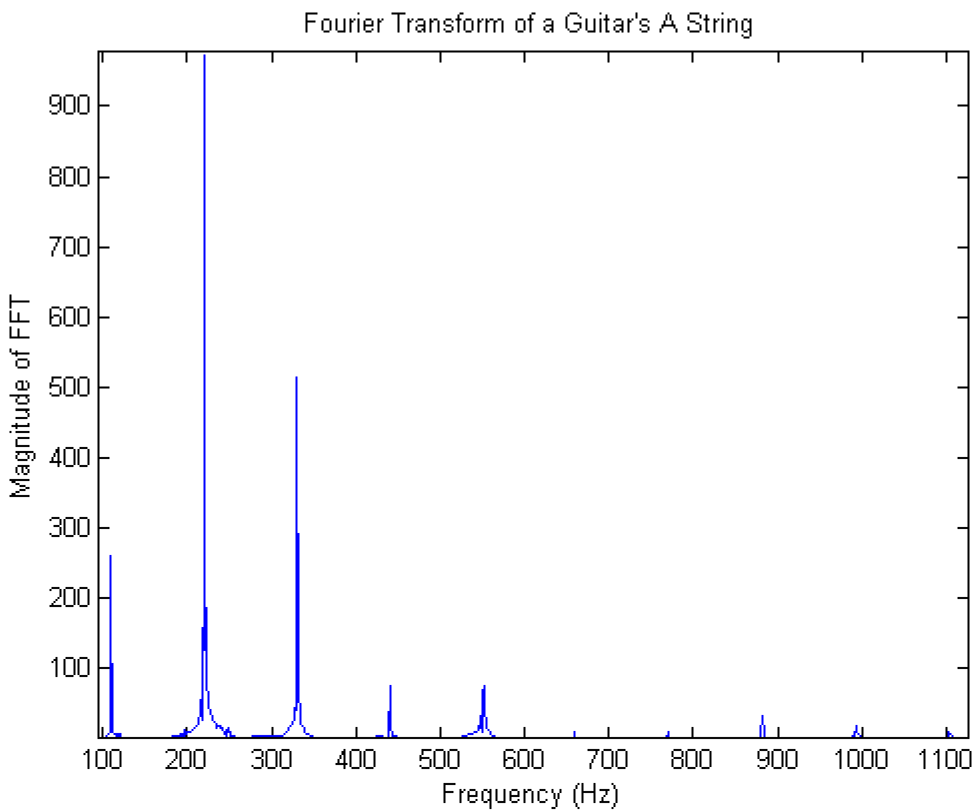
- Note name/duration



Implementation (c++)

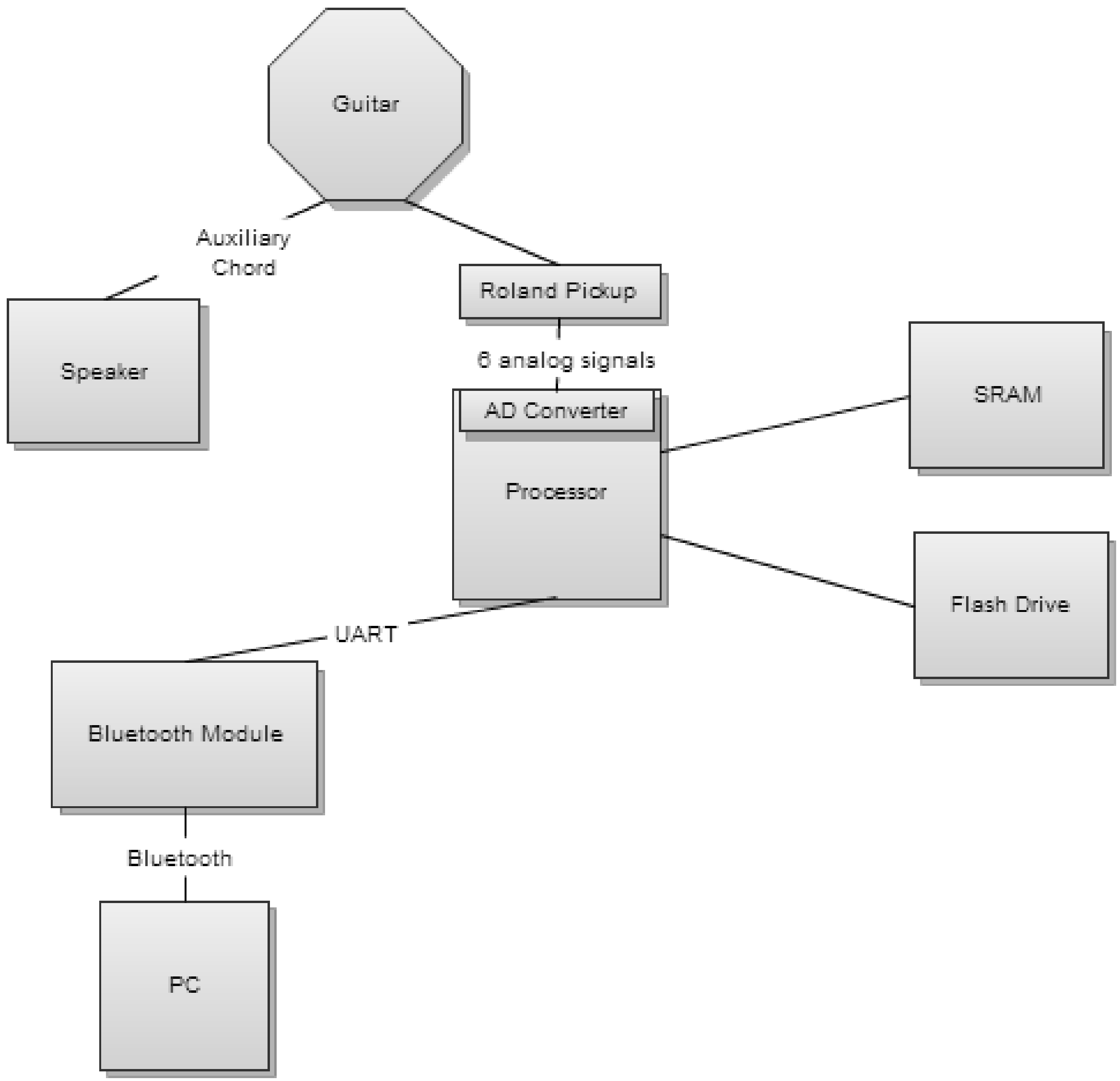
```
/*
 * Computes the discrete Fourier transform (DFT) of the given vector.
 * All the array arguments must have the same length.
 */
void compute_dft(double inreal[], double inimag[], double outreal[], double outimag[], int n) {
        int k;
        for (k = 0; k < n; k++) {  /* For each output element */
                double sumreal = 0;
                double sumimag = 0;
                int t;
                for (t = 0; t < n; t++) {  /* For each input element */
                    sumreal +=  inreal[t]*cos(2*M_PI * t * k / n) + inimag[t]*sin(2*M_PI * t * k / n);
                    sumimag += -inreal[t]*sin(2*M_PI * t * k / n) + inimag[t]*cos(2*M_PI * t * k / n);
                }
                outreal[k] = sumreal;
                outimag[k] = sumimag;
        }
}
```

DFT Result



Fourier Transform of a Guitar's A String
Magnitude of FFT
Frequency (Hz)
100
200
300
400
500
600
700
800
900
1000
1100

Annotated Block Diagram



Projective Schedule

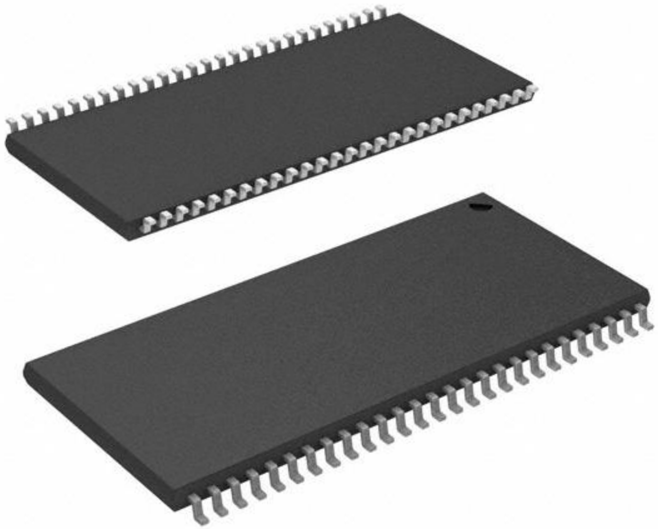
Mile Stone	Date	Task	Start Date	End Date	Person Responsible
Project Idea and team formation	09/30/12				
Refined project	10/07/12				
Initial design review	10/24/12				
		Processor Selection	10/14/12	10/24/12	Team
		Interface Design	10/14/12	10/24/12	Alex
		Bluetooth Design	10/14/12	10/24/12	Tang
		Power Supply	10/14/12	10/24/12	Larry
		Software Design	10/14/12	10/24/12	Tang
		Algorithm Design	10/14/12	10/24/12	Jeff
system level design	10/31/12	Purchase Parts	10/14/12	11/07/12	Alex
preliminary design review		Schematic Capture			
detailed design		PCB layout			
critical design review					
Implement					

Parts and Costs

- SRAM(16-Mb) – CY7C1061DV33…….$51.93
- Bluetooth Module…………………………$15.95
- Flash Drive (SST49LF080A334CNHE)..$5.93
- Roland GK2-AH……………………………Reused
- Processor – Philips LPC2214……………Free ☺

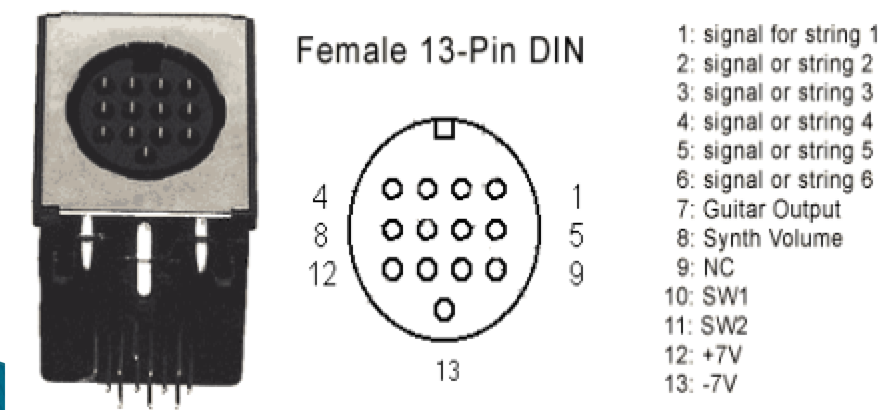
Rough Total=$73.81

SRAM (CY7C1061DV33)



- 16 MB CY7C1061DV33
- Supply Voltage 3.3V
- 54 pin package
- Access Time= 10ns
- Parallel Interface
- Max Clock Frequency: 100 MH
- Cost: $51.93

Roland GK2-AH



- Has 6 pickups that output 6 analog signals, one for each string.
- Supply Voltage: +/- 7V

Flash Drive (SST49LF080A334CNHE)

- Memory Type: NAND
- Memory Size: 8Mbit
- Access Time: 33ns
- Interface type: LPC
- Supply Voltage: 3.3V
- Bus Width: 8 bit
- Package: PLCC
- Operating Temp: +/- 70C
- Cost: $5.93

Bluetooth Module (RN42-I)

- Class 2 Bluetooth 2.1 + EDR module
- Supports Bluetooth data link to iPhone
- 128 bit encryption
- Built in error correction
- UART local and over the air RF configuration
- Supply Voltage: 3.3V
- Size: .52” x 1” x 0.07”
- Cost: $15.95

Processor (LPC2478)

- 512 kB on-chip flash memory
- 98 kB on-chip SRAM
- 10-bit ADC: mulitplex among 8 pins
- ADC 10-bit conversion time >= 2.44 μs
- ADC frequency 4.5 MHz
- UART 16B receive and transmit FIFO's, standard baud rate 115200 bit per second
- 3.3V power supply
- 4MHz internal clock

Member Specialities

- Alex Paige – Team Leader/Hardware/Music
- Zhiyuan Tang – Software/Signals
- Jeff Hanna – Power/OS
- Larry Zhao – CAD/Hardware

The End

- Questions
- Comments
- Suggestions