Critical Design Review:

By: Smart Guitar Group
Alex Paige
Larry Zhao
Henry Tang
Jeff Hanna
Table of Contents

1. Overview
2. Processor
   - Pin Configuration
   - Interface Specifications
   - Memory Mapping
   - Timing
3. Power Design
   - Analog Power
   - Digital Power
   - Guitar Pick-up Power
4. Components/Schematic
   - Bluetooth, SDRAM, SD card, Roland Pickup...
5. Bill of Materials
6. Software Design
Mode Overview

- **Tuning**
  - While in Tuning mode, the note being played is displayed on the LCD screen. Three LED's indicate whether the input is flat, sharp, or in tune.

- **Write to Memory**
  - While in this mode, the data produced by the FFT is stored in memory to be sent via bluetooth at another time.

- **Live**
  - This is the main operating mode. While in this mode FFT data is sent via bluetooth to a computer which will write the music score in live time.
Setting Time Signature and BPM

- The three digits can be incremented individually using the up buttons under them. If the input is unchanged for three seconds it will begin to blink, indicating that it is about to be set.

- After three blinks the Time signature has been set and the BPM is set using the same procedure. When the BPM blinks, it counts off the time signature at the current BMP rate and the FFT is started.
User Interface

Tuning: 0 0 0
Mode: 0 0 0
Power: 0 0 0
Power Design (Vdd & Gnd)

Power: 3.3V
  -Digital:
  15, 60, 71, 89, 112, 125, 146, 165, 181, 198, 26, 86, 174, 38
  -Analog: 20, 24

Gnd:
  -Digital:
  33, 63, 77, 93, 114, 133, 148, 169, 189, 200, 32, 84, 172
  -Analog: 22
Memory Mapping

OnBoard Flash:
To store program, 512k
Enough for our purpose

QPNano Statemachine
ARM C lib
FFT code
Sys Init & Peripheral Code

OnBoard RAM:
64K
Enough for running
program and in-place
1024 point FFT
Interface Specification:

UART0: Debug/On-Board Flash Programming Port
UART2: Bluetooth

Interrupt based: register: U0IIR - 0xE000 C008 U0IIR - 0xE000 C008

From Interrupt source(bit 3:1) to figure out which status register to read next

011 1 - Receive Line Status (RLS).
010 2a - Receive Data Available (RDA).
110 2b - Character Time-out Indicator (CTI).
001 3 - THRE Interrupt
Timing Concern:

Can the system run fast enough?
Our sampling rate $44$khz = $0.023$ms

Our target play rate is set at 5 times/sec per string, which equal to 30 times FFT calculation. $1/30=33$ms

Sampling takes about 11clock cycle per input channel, at 20mhz, it takes about 3microsec for all input channels.

According to following table, we can calculate 512 sample FFT at about $6$ms

Thus, ADC channels will have to use timed interrupt to take samples at sampling frequency, and use the rest of the time to calculate FFT.
## Timing Concern (Chart)

<table>
<thead>
<tr>
<th>Size of FFT N</th>
<th>40 MHz ARM7</th>
<th></th>
<th>40 MHz ARM7M</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time for one FFT</td>
<td>FFTs per second</td>
<td>Time for one FFT</td>
<td>FFTs per second</td>
</tr>
<tr>
<td>16</td>
<td>0.028ms</td>
<td>36,000</td>
<td>0.026ms</td>
<td>38,000</td>
</tr>
<tr>
<td>32</td>
<td>0.080ms</td>
<td>12,500</td>
<td>0.069ms</td>
<td>14,500</td>
</tr>
<tr>
<td>64</td>
<td>0.20ms</td>
<td>4,700</td>
<td>0.17ms</td>
<td>5,700</td>
</tr>
<tr>
<td>128</td>
<td>0.52ms</td>
<td>1,900</td>
<td>0.41ms</td>
<td>2,400</td>
</tr>
<tr>
<td>256</td>
<td>1.24ms</td>
<td>800</td>
<td>1.00ms</td>
<td>1,040</td>
</tr>
<tr>
<td>512</td>
<td>2.9ms</td>
<td>340</td>
<td>2.2ms</td>
<td>450</td>
</tr>
<tr>
<td>1024</td>
<td>6.6ms</td>
<td>150</td>
<td>4.9ms</td>
<td>200</td>
</tr>
<tr>
<td>2048</td>
<td>15ms</td>
<td>67</td>
<td>11ms</td>
<td>89</td>
</tr>
<tr>
<td>4096</td>
<td>33ms</td>
<td>30</td>
<td>25ms</td>
<td>41</td>
</tr>
<tr>
<td>8192</td>
<td>73ms</td>
<td>14</td>
<td>53ms</td>
<td>19</td>
</tr>
<tr>
<td>16384</td>
<td>160ms</td>
<td>6.3</td>
<td>115ms</td>
<td>8.7</td>
</tr>
</tbody>
</table>

*Table 1: Optimized timings*
Other software concern:

Note Time Stamp:
A threshold will be in place to detect string play, and the note will be time stamped. Even though the note will not be determined until later time due to FFT delay.
Schematic

- SD Card Reader
- 24-pin Display
- Rs232 (w/Level Shifter)
- 7 LED’s
- Bluetooth
- SDRAM
- Pushbutton
- 13-pin Socket
24-pin Display

Display

Processor

[Image of a circuit diagram showing connections between display and processor components, with labels and connections highlighted in various colors.]
RS232 and Level Shifter

CPU

Level Shifter

Rs-232
LED’s
Bluetooth

RXD3  176
TXD3  170

Bluetooth

RN-42-I/RM
RN42

UART_RX
UART_TX
SDRAM (Part 1)
SDRAM Connections
PushButton and Crystal
13-pin socket

13 pin Socket

CPU
Software Architecture

1. QPNano state machine
   - Open source, ARM compilable.
   - Three State: Tune, Record, Transmit

2. Optimized FFT
   - Fixed point FFT, and uses cos/sin lookup table.
   - Algorithm and code can be found online at:
     http://www.jjj.de/fft/fftpage.html

   No special math library needed.