CDR Agenda

- The Design
- Project Status and Goals
The Design
We will track the hands using ultrasonic trilateration.

Using the trilateration coordinates, software will display the hand in real time on a computer screen.
Ultrasonic Tracking System

Ultrasonic Transmitter
(will probably have 4 around the corners of the screen)

Ultrasonic receivers on hand

Mike
Ultrasonic Sensor System

Base Station

Glove

Host Computer

Wireless Data Transfer

Ultrasonic Transmissions

Trilateration data

Visualization of data
The Glove

- 6 ultrasonic receivers
  - One on each of 5 finger tips, one on back of hand
  - Can extrapolate the rest of the hand with this data

- 7 microprocessors
  - One for each receiver and one control

- RF Transceiver
  - Used to sync timing and transmit data to base station

- Battery powered
Glove Block Diagram

Ultrasonic Receivers (up to 6)

Receiver

Analog Conditioning

Timing μP

RF Transceiver

Battery

+3.9V

Power

Control μP

SPI

100 – 200mV sine

Receiver

Analog Conditioning

Timing μP

Serial (115200, 8N1)
One interrupt line

Power to all electronics

±3.3V

0 – 3.3V square

Mike
Glove Schematic
Glove Schematic

Control

RF Transceiver

Receiver Circuitry

Power System

Mike
Glove Power System

- Glove will be powered by rechargeable, 3.9V Li-Polymer battery

- Regulated output voltages: +3.3V, -3.3V

- +3.3V Boost Regulator
  - SP6641B
    - 0.9V to 4.5V input range
    - Rated 500mA

- -3.3V Regulator
  - LTC660
    - Input range 1.5V to 5.5V
    - Rated 100mA
# Glove Max Power Estimate

<table>
<thead>
<tr>
<th>Part Name</th>
<th>Part Number</th>
<th>Qty</th>
<th>+3.3 (mA)</th>
<th>-3.3 (mA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparator/Schmitt Trigger</td>
<td>ADCMP356</td>
<td>6</td>
<td>0.02</td>
<td>0.09</td>
</tr>
<tr>
<td>Op-Amp (dual)</td>
<td>LM6132BIM-ND</td>
<td>6</td>
<td>0.98</td>
<td>5.88</td>
</tr>
<tr>
<td>&quot;+Voltage Regulator&quot;</td>
<td>1016-1211-ND</td>
<td>1</td>
<td>0.07</td>
<td>0.07</td>
</tr>
<tr>
<td>&quot;-Voltage Regulator&quot;</td>
<td>LTC660CS8#PBF-ND</td>
<td>1</td>
<td>3.00</td>
<td>3.00</td>
</tr>
<tr>
<td>RF Transceiver</td>
<td>RFM12</td>
<td>1</td>
<td>30.00</td>
<td>30.00</td>
</tr>
<tr>
<td>Timing uProcessor</td>
<td>PIC12(L)F1840</td>
<td>6</td>
<td>3.40</td>
<td>20.40</td>
</tr>
<tr>
<td>Control uProcessor</td>
<td>PIC32MX3XX/4XX</td>
<td>1</td>
<td>75.00</td>
<td>75.00</td>
</tr>
<tr>
<td>Ultrasonic Receiver</td>
<td>SPM0404UD5</td>
<td>1</td>
<td>0.25</td>
<td>0.25</td>
</tr>
</tbody>
</table>

Total current supplied by battery: 140.57 mA

Totals: 134.69 mA, 444.5 mW, 5.88 mA, -19.4 mW
Glove Control

- Control microprocessor
  - PIC32MX695F512H
    - 32 bit, 80 MHz
    - Six hardware UART modules
    - 64 pins

- Six timing microprocessors
  - PIC12LF1840
    - 8 bit, 32 MHz precision internal oscillator
    - 8 pins
    - One hardware UART

- FTDI USB Bridge
  - Allows us to start with wired communication
Glove Receiver Circuitry

Ultrasonic Ping

Signal amplified and high-pass filtered

Fully Rectified

Comparator Output (threshold = 0.6V)

Ultrasonic Ping

Comparator OUT

LPF Cutoff Frequency = 723k

Mike
Ultrasonic Receiver

Specifications

- Directivity: Omni-directional
- Sensitivity (S): @ 1kHz (dB-1V/Pa) min=-46dB, nom=-42dB, max=-38dB
- Output Impedance = 300Ω
- Current Consumed across 1.5 to 3.6 volts = 250μA
- Supply Voltage = 1.5 to 3.6 V
RF Transceiver

- RFM22 from Sparkfun
  - Low power consumption
  - Software programmable frequency
  - Frequency range
    - 433/470/868/915MHz ISM bands
  - $12.00
Glove Electronics Layout
Base Station

Ultrasonic Transmitters
- Use four transmitters labeled A, B, C, and D; need a minimum of 3.
- Ping one at a time in a preset sequence
- All transmit at same frequency

RF Transceiver
- Used to sync timing between transmitters and receivers
- Receives trilateration data from Glove

Micro-Processor
- Used to control ultrasonic transmitters and sync time with glove electronics
- Prepares 3D information for software on host computer.
Base Station Block Diagram

Ultrasonic Transmitters (up to 4, minimum of 3)

0 – 20V square wave

Base Station

RF Transceiver

Control μP

Analog Amplification

Analog Amplification

Analog Amplification

Ultrasonic Transmitter

Ultrasonic Transmitter

Ultrasonic Transmitter

Power

SPI

SPI

SPI

SPI

+20V

+3.3V

+3.3V

DC from wall converter

Serial (115200, 8N1)

USB

USB

USB

0 - 3.3V 40 kHz square wave

Host Computer

Eric
Base Station Schematic
Base Station Power System

- Base station will be powered by DC wall adapter and USB
- +3.3V supplied from USB
- +20V regulator
  - LM317HVT
    - 1.2V to 57V output range
    - Rated 1.5A
## Base Station Max Power Estimate

<table>
<thead>
<tr>
<th>Part Name</th>
<th>Part Number</th>
<th>Qty</th>
<th>+20V</th>
<th>+3.3V</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>(mA)</td>
<td>(mA)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Each</td>
<td>Total</td>
</tr>
<tr>
<td>Op-Amp (dual)</td>
<td>LM6132BIM-ND</td>
<td>4</td>
<td>0.98</td>
<td>3.92</td>
</tr>
<tr>
<td>+20_Voltage Regulator</td>
<td>LM317HVT</td>
<td>1</td>
<td>4.00</td>
<td>4.00</td>
</tr>
<tr>
<td>RF Transceiver</td>
<td>RFM12</td>
<td>1</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Control uProcessor</td>
<td>PIC32MX3XX/4XX</td>
<td>1</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Ultrasonic Transmitter</td>
<td>SPM0404UD5</td>
<td>4</td>
<td>6.25</td>
<td>25.00</td>
</tr>
<tr>
<td><strong>Totals:</strong></td>
<td></td>
<td></td>
<td>32.92</td>
<td>658.4</td>
</tr>
</tbody>
</table>

|                               |                  |     | (mA)     | (mW)      | (mA)  | (mW)  |
|                               |                  |     | Each     | Total     | Each  | Total |

Eric
Base Station Control

- Control microprocessor
  - PIC32MX695F512H
    - 32 bit, 80 MHz
    - Six hardware UART modules
    - 64 pins

- USB Interface with Host Computer
  - FTDI USB Bridge
Amplified Signal

Output from microcontroller
Ultrasonic Transmitter

Specifications

- Sound Pressure Level (SPL): 117dB (0dB=0.2n bar)
- Sensitivity (SEN): –60dB (0dB 1V/u bar)
- Impedance: 1K Ohm
- Operating Frequency: 38KHz to 42KHz
- Drive Voltage: 20V RMS maximum (60V peak to peak maximum @10% duty)
RF Transceiver

- RFM22 from Sparkfun
  - Low power consumption
  - Software programmable frequency
  - Frequency range
    - 433/470/868/915MHz ISM bands
  - $12.00
Base Station Layout
Host Software

Layers architecture to separate interface, control, computation, and data aspects of application.
Host Software

<<Interface>>
BaseStationInterrupt

+int UpdateModel(string FormattedData)

-string ParseData(string BufferData)

-string ReceivedData

-string FormattedData

Application Layer
This layer contains a session controller which synchronizes data calculations, model updates, and image rendering.
Host Software

<table>
<thead>
<tr>
<th>&lt;&lt;Control&gt;&gt;</th>
<th>SessionController</th>
</tr>
</thead>
<tbody>
<tr>
<td>+void Run()</td>
<td></td>
</tr>
<tr>
<td>-int ExtrapolateDataPoints()</td>
<td></td>
</tr>
<tr>
<td>-int UpdateObjectPositions()</td>
<td></td>
</tr>
<tr>
<td>-void RenderImages()</td>
<td></td>
</tr>
</tbody>
</table>

**Interface Layer**

Used to communicate with the base station, this layer contains a class that interrupts the application any time new data is available.
Host Software

**Domain Layer**
- Object Interaction Processor
- Inverse Kinematics Processor
- Image Renderer

<table>
<thead>
<tr>
<th><strong>&lt;&lt;Utility&gt;&gt;</strong></th>
<th>RenderImages</th>
</tr>
</thead>
<tbody>
<tr>
<td>+void UpdateCanvas()</td>
<td></td>
</tr>
<tr>
<td>-CoordinateSet GetHandData()</td>
<td></td>
</tr>
<tr>
<td>-void RenderHandModel(CoordinateSet Data)</td>
<td></td>
</tr>
<tr>
<td>-PrimitiveObject GetObjectData(int Index)</td>
<td></td>
</tr>
<tr>
<td>-void RenderObject(PrimitiveObject Object)</td>
<td></td>
</tr>
<tr>
<td>-Image CurrentImage</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>&lt;&lt;Utility&gt;&gt;</strong></th>
<th>ObjectInteractionProcessor</th>
</tr>
</thead>
<tbody>
<tr>
<td>+void UpdateObjectPositions()</td>
<td></td>
</tr>
<tr>
<td>-CoordinateSet GetHandPosition()</td>
<td></td>
</tr>
<tr>
<td>-PrimitiveObject GetGraphicsObject(int Index)</td>
<td></td>
</tr>
<tr>
<td>-PrimitiveObject DetermineInteraction(CoordinateSet HandPosition, PrimitiveObject GraphicsObject)</td>
<td></td>
</tr>
<tr>
<td>-void UpdateObjectPosition(PrimitiveObject GraphicsObject)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>&lt;&lt;Utility&gt;&gt;</strong></th>
<th>InverseKinematicsProcessing</th>
</tr>
</thead>
<tbody>
<tr>
<td>+void UpdateIKPoints(CoordinateSet IKPoints)</td>
<td></td>
</tr>
<tr>
<td>-CoordinateSet GetAbsoluteCoordinates()</td>
<td></td>
</tr>
<tr>
<td>-CoordinateSet CalculateIKPoints()</td>
<td></td>
</tr>
<tr>
<td>-CoordinateSet IKPoints</td>
<td></td>
</tr>
</tbody>
</table>
## Host Software

**Interface Layer**

Used to communicate with the base station, this layer contains a class that interrupts the application any time new data is available.

### Data Management Layer

Stores model data representing joint positions as well as virtual primitives such as boxes, spheres, and images.

<table>
<thead>
<tr>
<th><strong>ModelData</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>+CoordinateSet GetHandCoordinates()</td>
</tr>
<tr>
<td>+PrimitiveObject GetGraphicsObject(int Index)</td>
</tr>
<tr>
<td>+RotationSet GetHandOrientation()</td>
</tr>
<tr>
<td>+void UpdateAbsoluteCoordinates(CoordinateSet NewCoordinates)</td>
</tr>
<tr>
<td>+void UpdateExtrapolatedCoordinates(CoordinateSet NewCoordinates)</td>
</tr>
<tr>
<td>-CoordinateSet AbsoluteCoordinates</td>
</tr>
<tr>
<td>-CoordinateSet ExtrapolatedCoordinates</td>
</tr>
<tr>
<td>-RotationSet HandOrientation</td>
</tr>
<tr>
<td>-PrimitiveObjects[] GraphicObjects</td>
</tr>
</tbody>
</table>
Software programs will be written for all microprocessors

- Base firmware
- Glove firmware
# Basic Timing Diagram

## Goal of 20 Hz Refresh

<table>
<thead>
<tr>
<th>Step #</th>
<th>Time</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1</strong></td>
<td>0</td>
<td>Send “Start Timer” signal from Base Station to Glove via RF transceiver</td>
</tr>
<tr>
<td><strong>2</strong></td>
<td>T1</td>
<td>Timers on six microprocessors on Glove will start</td>
</tr>
<tr>
<td><strong>3</strong></td>
<td>T2</td>
<td>Send Ultrasonic Ping from Transmitter (A the first time, B the second time, etc)</td>
</tr>
<tr>
<td><strong>4</strong></td>
<td>T3</td>
<td>Ping received on six ultrasonic receivers; timers stopped</td>
</tr>
<tr>
<td><strong>5</strong></td>
<td>T4</td>
<td>Distance calculated for each receiver</td>
</tr>
<tr>
<td><strong>6</strong></td>
<td>T5</td>
<td>Data transmitted to Control microprocessor on Glove</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>REPEAT Steps 1-6 FOR 3 OTHER TRANSMITTERS</strong></td>
</tr>
<tr>
<td><strong>24</strong></td>
<td>T7</td>
<td>Perform trilateration calculations in Glove Control Microprocessor</td>
</tr>
<tr>
<td><strong>25</strong></td>
<td>T8</td>
<td>Transmit 3D Location Data from Glove to Base Station</td>
</tr>
<tr>
<td><strong>26</strong></td>
<td>T9</td>
<td>Base Station Checks Data Integrity and sends to Host Computer</td>
</tr>
<tr>
<td><strong>27</strong></td>
<td>T10</td>
<td>Redraw Hand in Software</td>
</tr>
<tr>
<td><strong>28</strong></td>
<td>T11</td>
<td>Wait X ms before restarting process</td>
</tr>
</tbody>
</table>
Software-Base State Diagram

User turns on base station

No response

Base scans for an active glove

Glove sends acknowledge message

Base and glove handshake

Handshake failed

Handshake successful

No response

Base scans for host application

Application sends acknowledge message

Repeat for each emitter

Base sends RF synchronization signal to glove

Glove starts timers

Base generates 1ms ultrasonic pulse

Receivers detect pulse and stop timers

Glove saves time value

Data corrupted

Glove calculates 3D coordinates

Values sent to base over RF

Base calculates checksums of received data

Checksums correct, values sent to host

Application renders graphics

Visual representation of the Ultrasonic Tracking System—State Diagram.
Base Station Primary Software Components

- UART Interface
- Session Controller
- SPI Interface
- Signal Generation
Base Station State Diagram

- Send handshake signal to glove
  - ACK
  - NACK
  - NACK

- Send handshake signal to host
  - ACK

- Delay
  - Still Connected
  - Connection Lost
  - No response

- Send RF Synchronization

- Generate 40KHz Signal

- Get data from glove

- Send data to host application

- Response from glove
Glove Primary Software Components

- UART Interface
- SPI Interface
- Slave Controller
- Data Processor
- Session Controller
Glove State Diagram

- **Send handshake signal to base**
- **Scan for synchronization signal**
- **Raise slave interrupt lines**
- **Wait for all slaves to send data**
- **Send data to base station**

Connections:
- **Connection Lost**
- **Send error signal**
- **Loop 4 Times**
- **Timeout**
- **ACK**
Slave Processor Primary Components

- UART Interface
- Session Controller
- Control Interrupt
- Timing System
Slave Processor State Diagram

- **Scan for timer start signal**: Sent
- **Start Timer**: Wait for comparator interrupt
- **Send timer value to controller and clear registers**: Received
- **Comparator activated**: Timeout

Chris
Project Status and Goals
Project Status

- **Glove Electronics**
  - Revision A schematic and layout completed
  - Receiver circuitry tested

- **Base Station Electronics**
  - Revision A schematic and layout completed
  - Ultrasonic transmission circuitry tested
  - Prototype board fabricated and populated
  - Control microprocessor up and running
  - UART data to Host functioning

- **Host**
  - Software architecture defined
Goals

- **Low**
  - Track one point and display point in a virtual environment

- **Mid**
  - Track two points and display in a virtual environment. Use tracked points to interface with virtual environment (i.e. rotate, zoom)

- **High**
  - Track entire hand and display in a virtual environment.
Milestone Goals

Milestone 1
- Send characters from Base Station to Host and display
- Generate signal on Base Station Control Processor, observe 20V output signal across emitter
- Generate ultrasonic pulses and measure square wave at output of comparator
- Send characters from timing processor to control processor every time interrupt is generated

Milestone 2
- Send characters between Base Station and Glove via RF
- Demonstrate battery powered operation for Glove
- Generate trilateration coordinates for one point
- Display virtual point
Capstone Expo Goal

- Track entire hand and display in a virtual environment.
Schedule Overview

- Analog circuitry for transmitters and receivers
- Transmitter and receiver timer synchronization, data communication, etc (using wires, not RF)
- Implement three transmitters and one receiver to track position of one point
- Track point and draw virtually
  - *Low level goal met*

- Go wireless (i.e. add RF, add battery)
- Add one more ultrasonic receiver
- Use two tracked points to interface with virtual environment (i.e. rotate, zoom)
  - *Mid level goal met*

- Add all receivers (6 total)
- Improve software to track multiple points
- Improve software to draw hand (extrapolate from 6 points)
  - *Capstone expo goal completed*
Schedule Overview

- Analog circuitry for transmitters and receivers
- Transmitter and receiver timer synchronization, data communication, etc (using wires, not RF)
- Implement three transmitters and one receiver to track position of one point
- Track point and draw virtually
  - *Low level goal met*
  - Go wireless (i.e. add RF, add battery)
  - Add one more ultrasonic receiver
  - Use two tracked points to interface with virtual environment (i.e. rotate, zoom)
    - *Mid level goal met*
- Add all receivers (6 total)
- Improve software to track multiple points
- Improve software to draw hand (extrapolate from 6 points)
  - *Capstone expo goal completed*

Jake
# Parts Received

<table>
<thead>
<tr>
<th>Expense</th>
<th>Quantity</th>
<th>Cost per</th>
<th>Total Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Op Amp, Dual</td>
<td>11</td>
<td>$3.50</td>
<td>$38.50</td>
</tr>
<tr>
<td>Bridge Rectifier</td>
<td>5</td>
<td>$0.70</td>
<td>$3.50</td>
</tr>
<tr>
<td>Voltage Regulators</td>
<td>5</td>
<td>$5.50</td>
<td>$27.50</td>
</tr>
<tr>
<td>Ultrasonic Transducer</td>
<td>2</td>
<td>$6.35</td>
<td>$12.70</td>
</tr>
<tr>
<td>RFM22B Wireless Transceiver</td>
<td>2</td>
<td>$13.93</td>
<td>$27.86</td>
</tr>
<tr>
<td>Ultrasonic Sensors</td>
<td>5</td>
<td>$7.90</td>
<td>$39.50</td>
</tr>
<tr>
<td>SMD Resistors, Capacitors and Crystals</td>
<td>1</td>
<td>$60.02</td>
<td>$60.02</td>
</tr>
<tr>
<td>USB to UART Bridge - FT232R</td>
<td>2</td>
<td>$4.00</td>
<td>$8.00</td>
</tr>
<tr>
<td>MPLAB PICkit 3 Programmer</td>
<td>1</td>
<td>$50.00</td>
<td>$50.00</td>
</tr>
<tr>
<td>USB Mini-B SMD Connector</td>
<td>2</td>
<td>$1.50</td>
<td>$3.00</td>
</tr>
<tr>
<td>Power Inductors</td>
<td>2</td>
<td>$2.50</td>
<td>$5.00</td>
</tr>
<tr>
<td>PIC MCU 32-Bit</td>
<td>2</td>
<td>$10.75</td>
<td>$21.50</td>
</tr>
<tr>
<td>PIC MCU 8-Bit</td>
<td>6</td>
<td>$1.50</td>
<td>$9.00</td>
</tr>
<tr>
<td>Diodes</td>
<td>8</td>
<td>$0.50</td>
<td>$4.00</td>
</tr>
<tr>
<td>Potentiometers</td>
<td>2</td>
<td>$3.75</td>
<td>$7.50</td>
</tr>
<tr>
<td>Transistors</td>
<td>6</td>
<td>$0.56</td>
<td>$3.36</td>
</tr>
</tbody>
</table>

**TOTAL EXPENSES:** $320.94
## Still to be purchased...

<table>
<thead>
<tr>
<th>Expense</th>
<th>Quantity</th>
<th>Cost per</th>
<th>Total Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIC MCU 32-Bit</td>
<td>3</td>
<td>$10.75</td>
<td>$32.25</td>
</tr>
<tr>
<td>PIC MCU 8-Bit</td>
<td>10</td>
<td>$1.50</td>
<td>$15.00</td>
</tr>
<tr>
<td>Crystal Oscillators</td>
<td>4</td>
<td>$5.00</td>
<td>$20.00</td>
</tr>
<tr>
<td>RFM22B Wireless Transceiver</td>
<td>3</td>
<td>$13.93</td>
<td>$41.79</td>
</tr>
<tr>
<td>Printed Circuit Board - BatchPCB.com</td>
<td>40</td>
<td>$2.50</td>
<td>$110.00</td>
</tr>
<tr>
<td>Wall Wart - 24VDC Power Supply</td>
<td>1</td>
<td>$15.00</td>
<td>$15.00</td>
</tr>
<tr>
<td>Lithium Ion Battery 850mAh</td>
<td>2</td>
<td>$9.00</td>
<td>$18.00</td>
</tr>
<tr>
<td>Ultrasonic Sensors</td>
<td>8</td>
<td>$7.90</td>
<td>$63.20</td>
</tr>
<tr>
<td>Ultrasonic Transducers</td>
<td>4</td>
<td>$6.35</td>
<td>$25.40</td>
</tr>
<tr>
<td>SMD Resistors, Capacitors, LEDs, etc.</td>
<td>1</td>
<td>$100.00</td>
<td>$100.00</td>
</tr>
<tr>
<td><strong>TOTAL EXPENSES:</strong></td>
<td></td>
<td></td>
<td><strong>$440.64</strong></td>
</tr>
</tbody>
</table>

Jake
### Updated Budget

**Total Budget Expense**

<table>
<thead>
<tr>
<th>Parts Received</th>
<th>$320.94</th>
</tr>
</thead>
<tbody>
<tr>
<td>Future Purchases</td>
<td>$440.64</td>
</tr>
<tr>
<td><strong>Grand Total:</strong></td>
<td><strong>$761.58</strong></td>
</tr>
<tr>
<td><strong>UROP</strong></td>
<td>Up to $1000</td>
</tr>
</tbody>
</table>
Risks

- Sensors can not distinguish small enough movements to accurately represent hand movement such as entire arm or body.
  - Modify project goal to sense large object movement such as entire arm or body.

- Time
  - Work hard to stay on schedule.
Risks

- Ambient noise on RF and acoustic signals
  - Change RF transceiver frequency (can be done in software)
  - Strongly filter acoustic input signals

- Extrapolating the hand accurately
  - We are hardware sided group
  - Consult people with experience
Our Gauge for Success

- How accurately can we track the individual points on the hands?
  - Have we improved from what has been built in the past?

- How well can we extrapolate finger position and hand orientation in the 3D model?
  - Measured with visual observation

- How well does the entire system perform?
  - Meaning how quickly and smoothly we can render the hands such that it appears instantaneous
Questions?

Ultrasonic Transmitter
(will probably have 4 around the corners of the screen)

Ultrasonic receivers on hand