Maestro™

SFU

(Sheet Music Scanner)

Harmony Innovations Inc.

Maestro Team: Sean Edmond
Nikola Cucuk
Veronica Cojocaru
Cris Panaitiu

December 14th 2010

Overview

1. Motivation
2. Market Research
3. Maestro Team
4. System Overview
5. Camera Module
6. Sound Module
7. Optical Character Recognition (OCR)
8. Challenges
9. Finances
10. What We Have Learned
11. Future Work
12. Acknowledgments / Questions
13. Live Demo

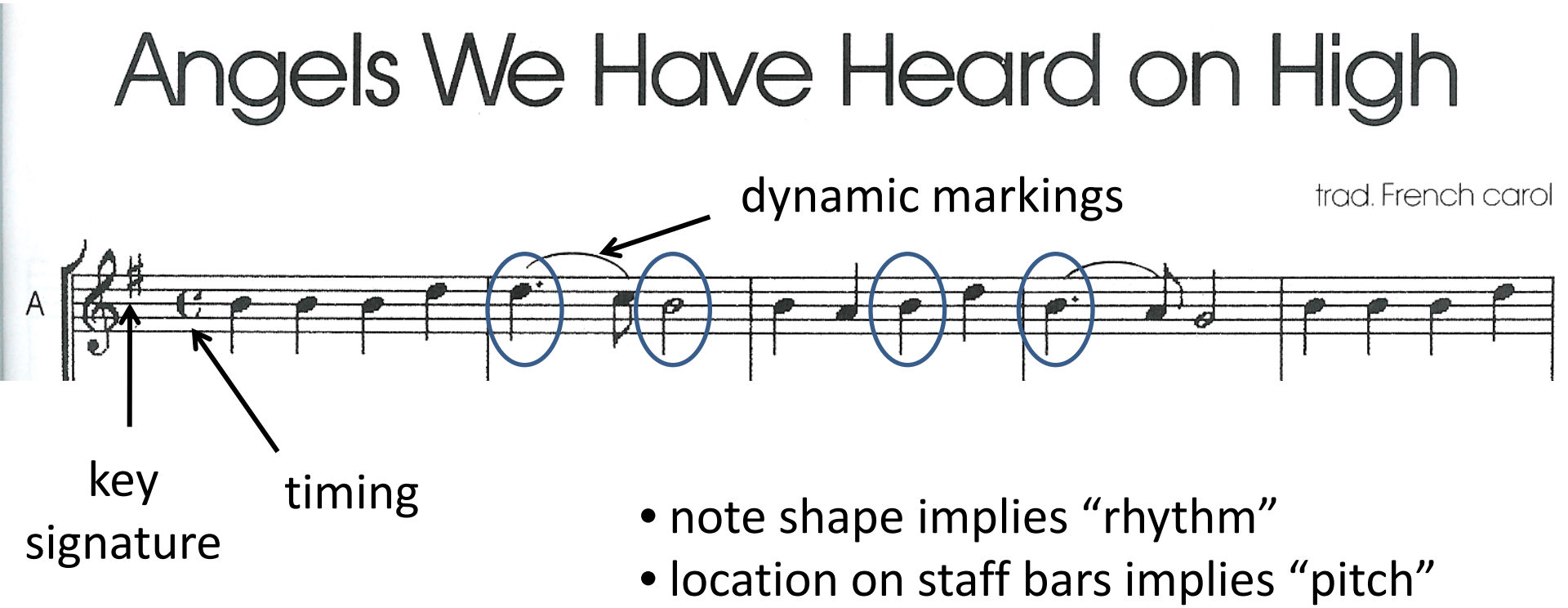
SFU

Motivation

- **How do novice musicians learn to play their instrument?**
- **Sheet music is complex**
- **Different learning styles**
- **Improvisation**

SFU

Sheet Music



- note shape implies “rhythm”
- location on staff bars implies “pitch”

Maestro™ Team

Sean Edmond

- 5th year Electronics Engineering student
- Was responsible for OCR software and sound module
- Lots of software and hardware experience in many languages
- Worked at MDA, Reconfigurable Computing Lab, PMC-Sierra

Nikola Cucuk

- 5th year Electronics Engineering student
- Was responsible for the camera module
- Experienced with embedded programming
- Worked at PMC-Sierra as HW designer and verifier

SFU

Maestro™ Team

Veronica Cojocaru

- 5th year Electronics Engineering student
- Responsible for MIDI driver, user interface, system integration
- Experienced with MatLab, C programming
- Worked at PMC-Sierra and Research at SFU

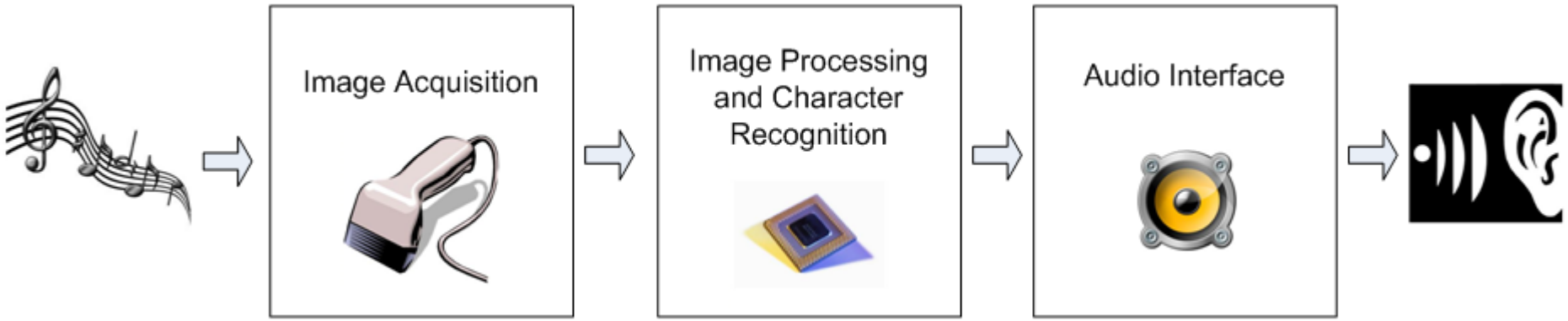
SFU

Market Research

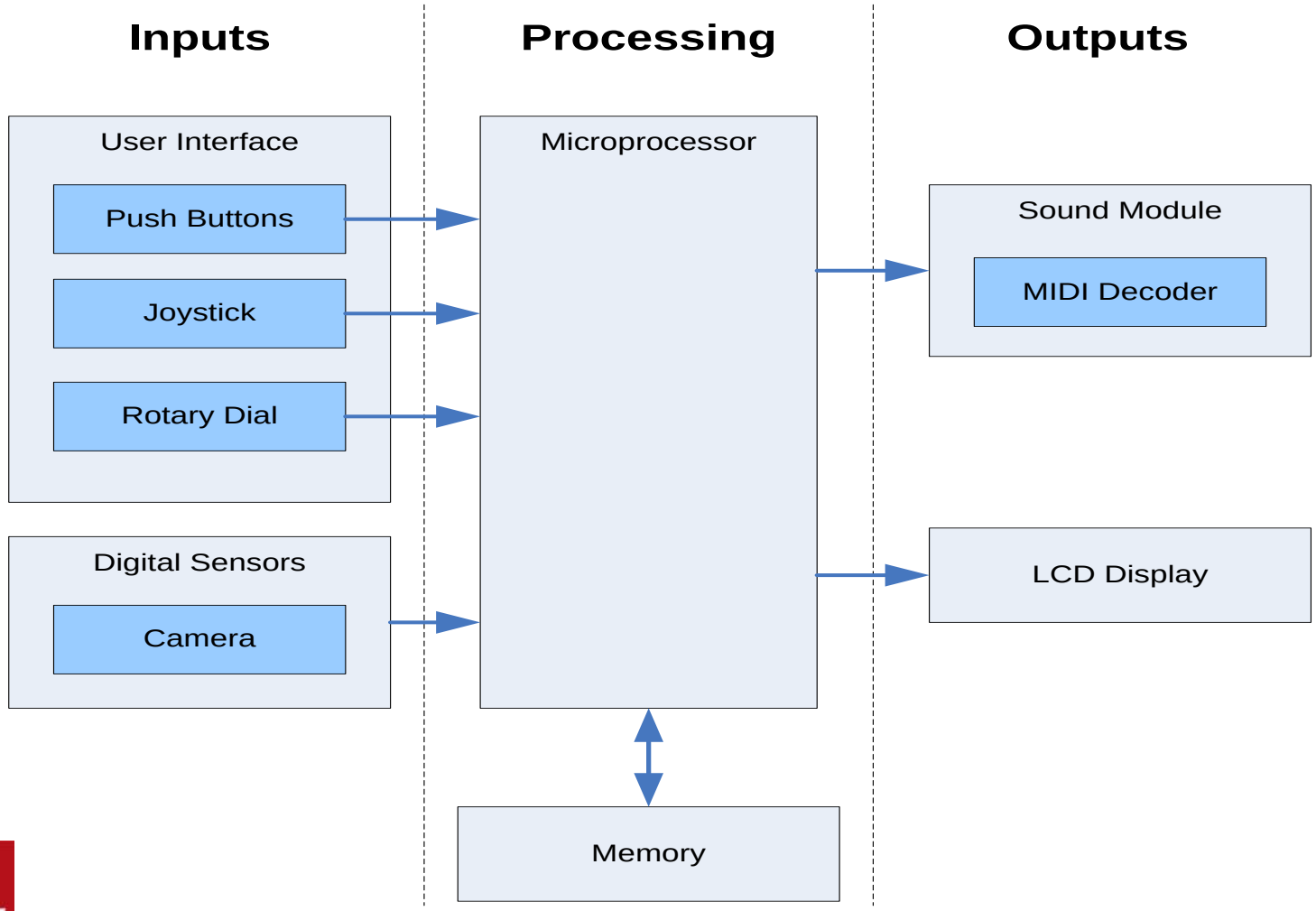
- Performed market research at music stores
- " Musicians are always looking for a way to skip learning sheet music "
- Sheet music packaged with CDs
- Software tools for computers
- Majority of string musicians purchase an electronic tuner
- MIDI is standard in electronic music devices
- Nothing like our device

SFU

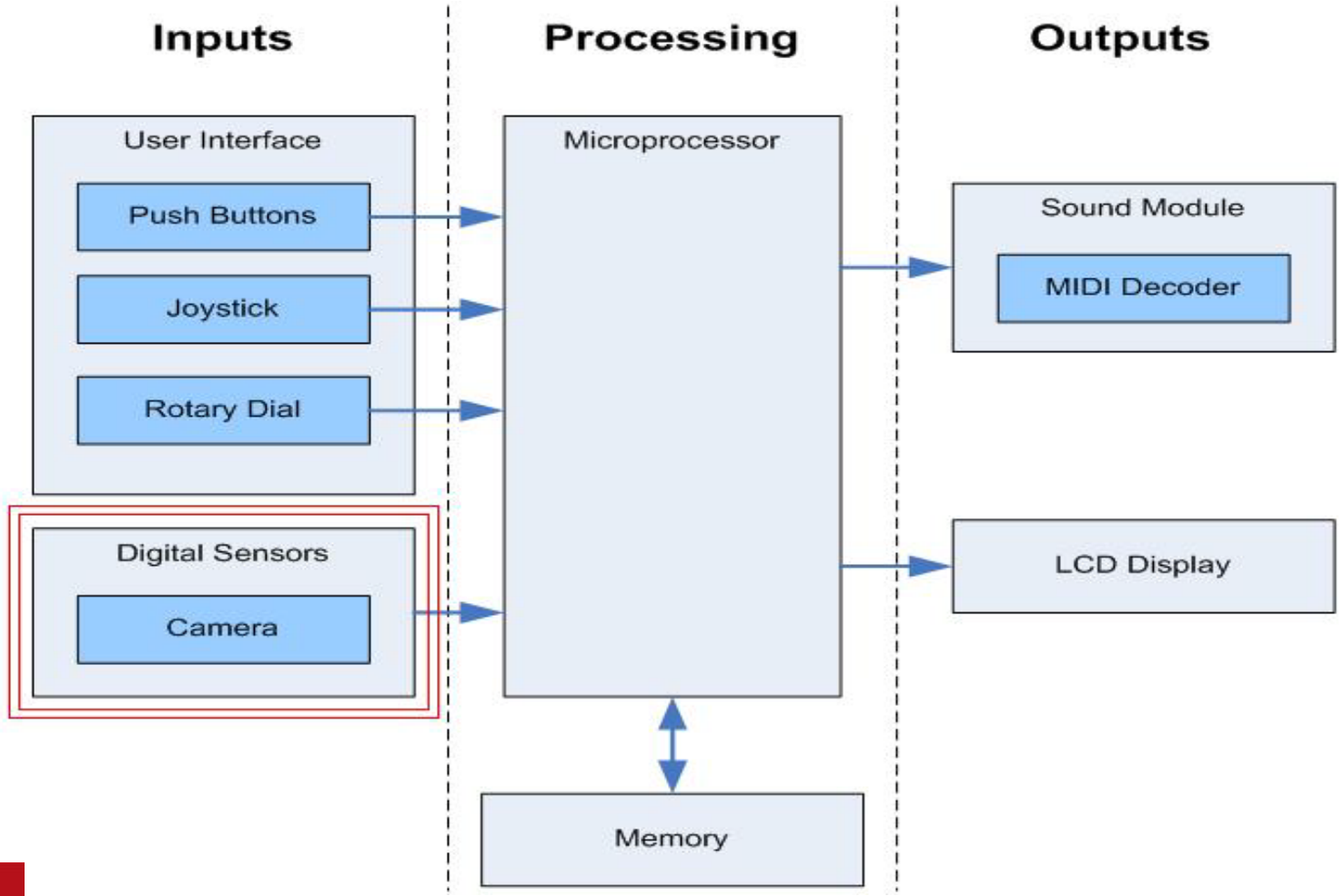
High-Level System Overview



High-Level System Overview



Camera Module



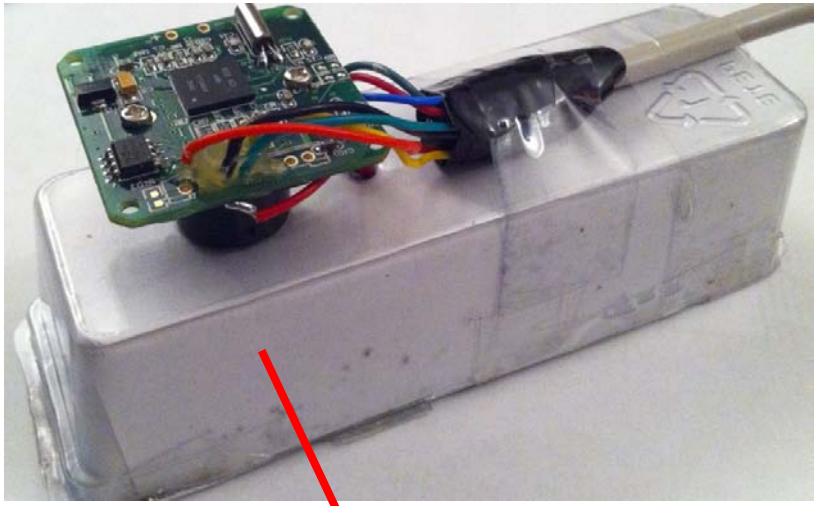
Camera Module (uCAM-232)

- CMOS Digital color camera
- RS232 interface
- Package size 28x32 mm
- 160x120 resolution
- Smallest pixel size 5.6um^2
- 2-bit gray colour conversion
- 0.4 fps
- 90° lens

SFU

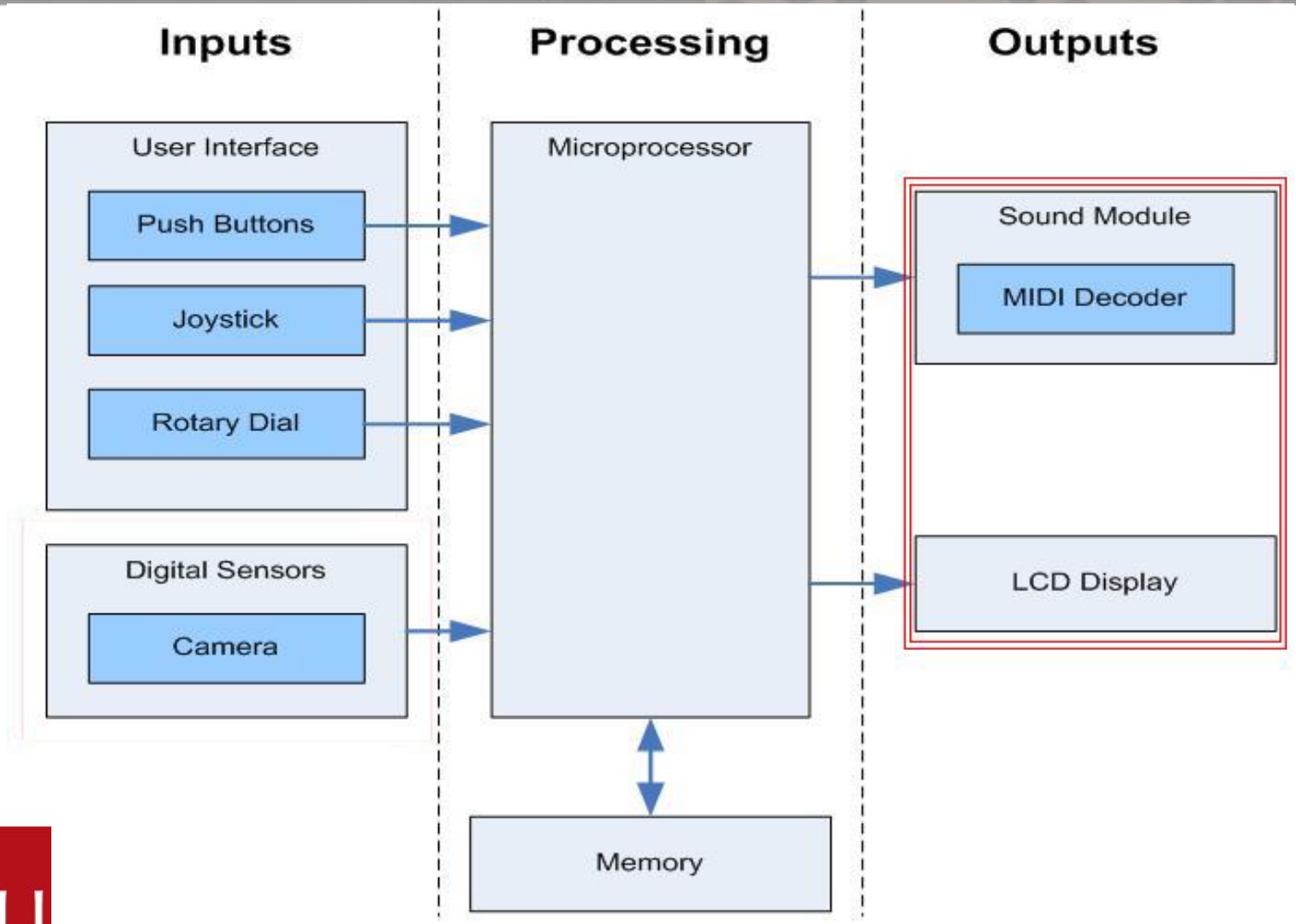
Camera Mudule Enclosure

- Separate mobile module
- Illumination box
  - Aluminum foil + white paper
  - 3x MAGlite bulbs
- Rectangular box for ease of straight line scan



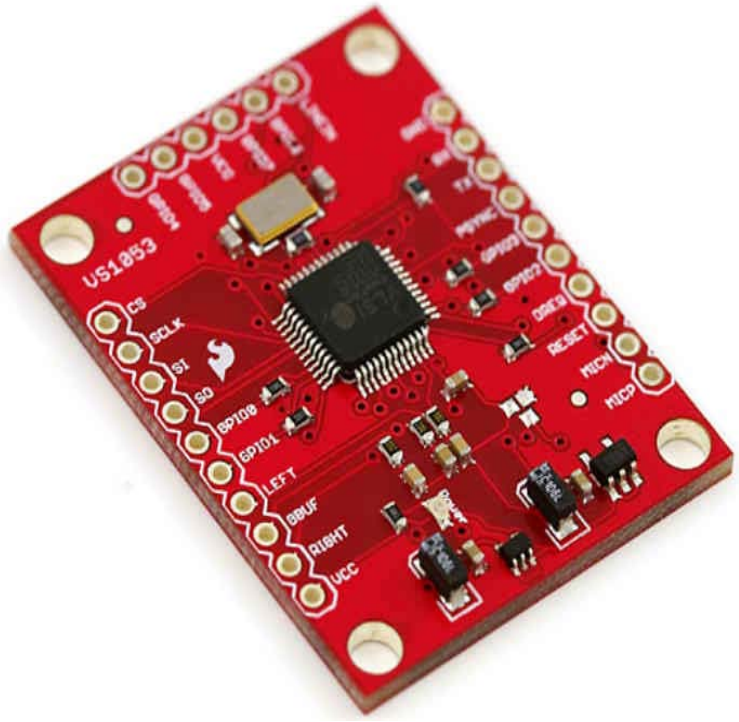
SFU

Sound Module



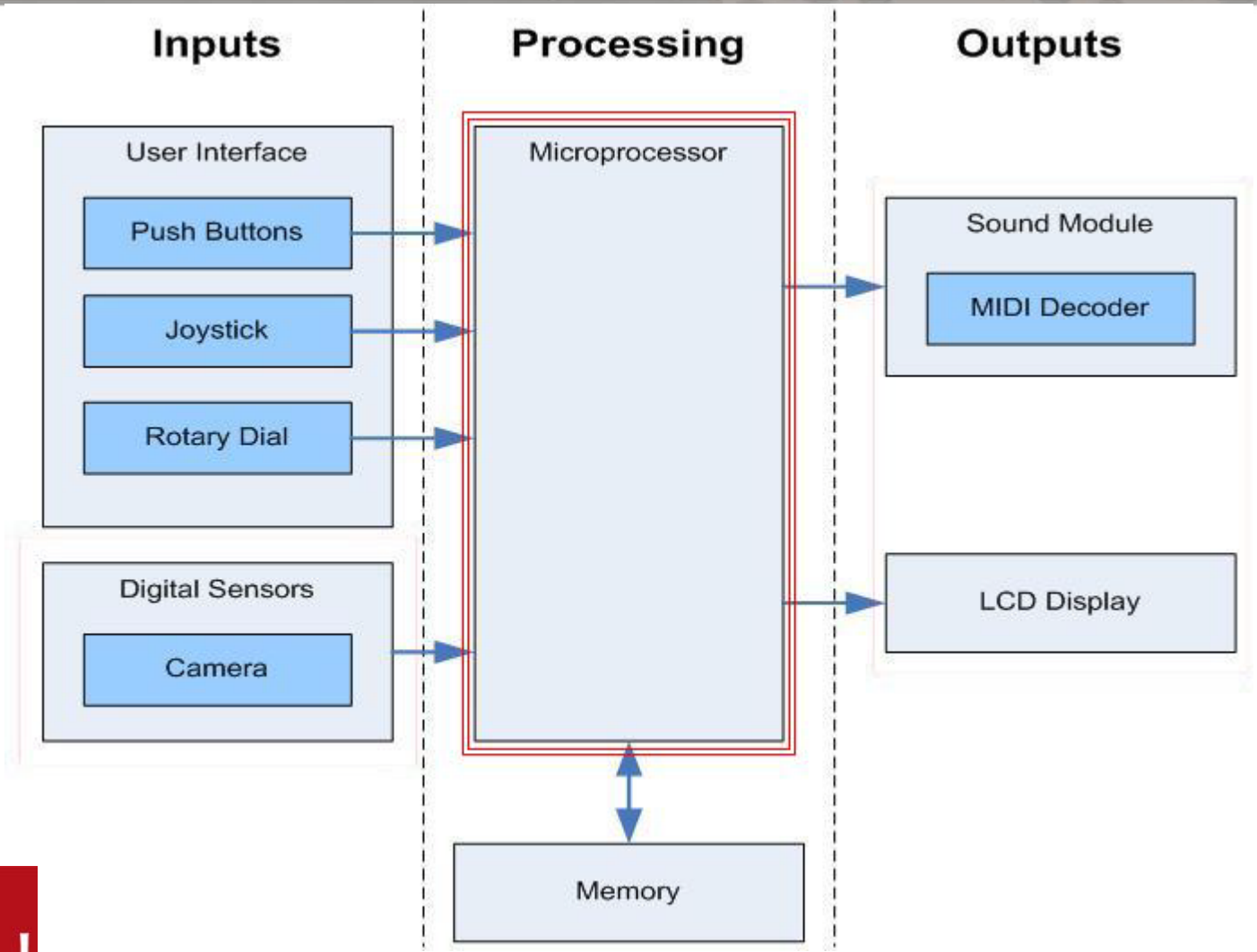
Sound Module

- Vs1053b packaged on break-out board
- MIDI decoder
- 2 x SPI interface
  - Serial Control interface
  - Serial Data Interface
- Sound output to standard headphone jack
- Package Size 7x7x1.4mm



SFU

Software


Inputs
Processing
Outputs
User Interface
Push Buttons
Joystick
Rotary Dial
Digital Sensors
Camera
Microprocessor
Memory
Sound Module
MIDI Decoder
LCD Display

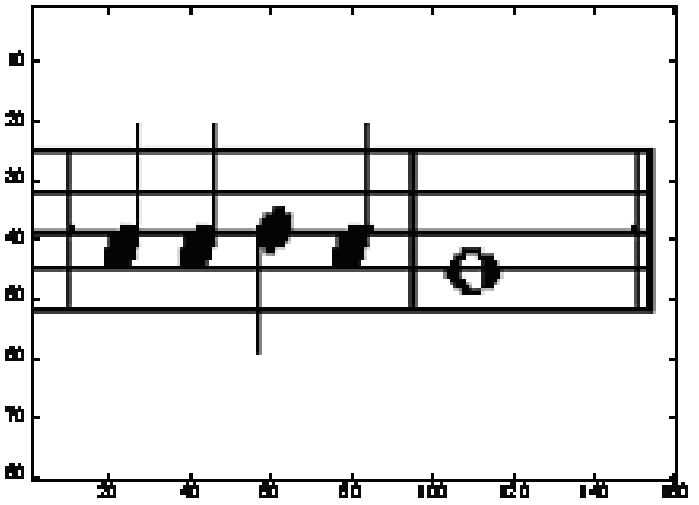
The Development Board

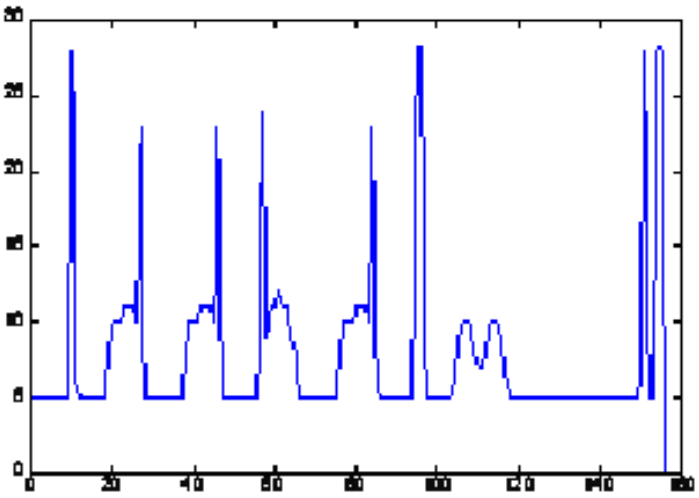
- EVK1100
  - Atmel 32-bit processor
  - SDRAM
  - LCD
  - Pushbuttons
  - joystick
- Drivers available for many standard interfaces
- Large online user community

SFU

Optical Character Recognition

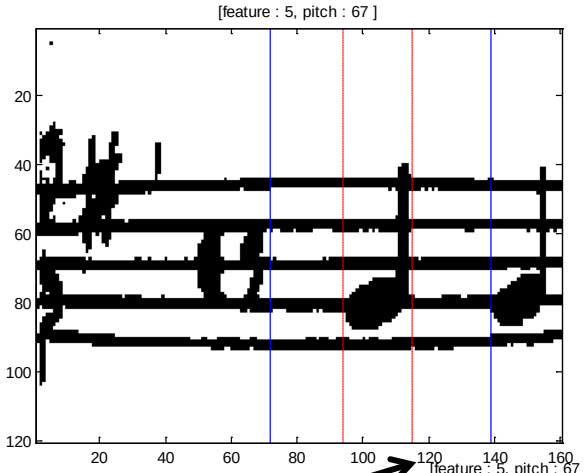
- Character recognition performed with “black pixel histogram” comparison
- Mosaic algorithm stitches images
- MIDI file created from detected features
- Algorithm was prototyped in Matlab





SFU

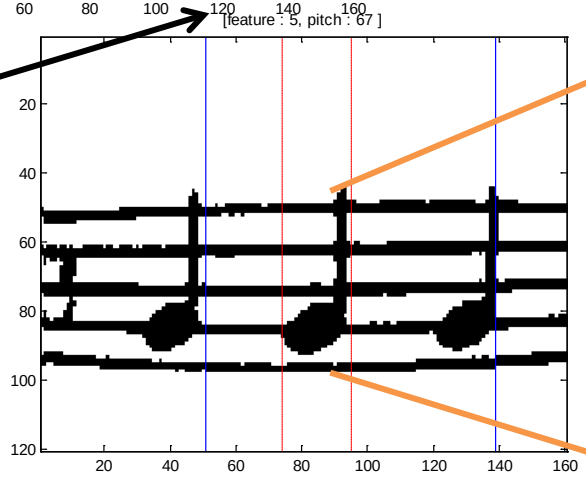
Sample Matlab Output

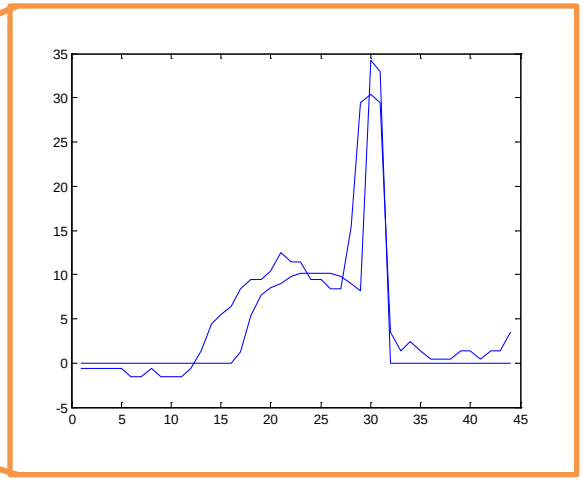


- Blue is the “search region”
- Red is feature edge detection

Sample “black histogram” comparison

Point of overlap determined by mosaicing algorithm





SFU

Finances

Equipment List	Cost
Microcontroller & EVBD	$129
Sound module	$44.93
Camera_A Camera_B1 Camera_B2	$60 $105 $105
Microcontroller programmer	free
Electronics	$160
Miscellaneous	$150
Total Cost	**$753.93**

SFU

Challenges

- Microcontroller couldn’t meet timing requirements of initial camera
- Second camera fried due to a short
- Tools were error prone
- MIDI specifications not readily available
- Resources shift required to complete sound module
- Scan area illumination challenges
- Ex-group member reporting false progress
- Processing speed

SFU

What We Have Learned

- Optical recognition software is challenging!!!
- Software development should employ “extreme programming” practices
- Project management
- Image processing requires a high speed processor
- Don’t rely on firmware to capture high speed interrupts use hardware instead

SFU

Future Work

- Is this viable?
- Dedicated hardware to accelerate firmware and algorithm
- Better camera with higher speed interface
- More compact design with PCB and custom parts
- Digital switch for illumination
- Improve algorithm
- Iphone app?

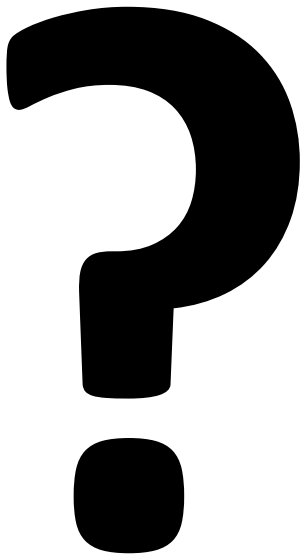
SFU

Acknowledgments

- Lakshman (Lucky) One
- Adrienne Jacobs (painting)
- Sean roomates ☺

SFU

Questions…



SFU