Automated Bio-Medical Diagnostic Collection System

CECS Senior Design Group 12:
Jesse Easterling   Jonathan Adams
Jimmy Nguyen      Zachary Levy
What is Doc Box?

- Saliva Test Collection System
- Allow for more convenient & effective way to get tested
- Saliva Can Tell: Vitamin Deficiencies, Pregnancies, STD’s, Drugs ...
Goals & Objectives

• Design Experience: Embedded Systems, Printed Circuit Boards, Motor Control Methods, Circuitry
• Teamwork in Engineering projects
• Provide inexpensive and easy solution to track personal health
Requirements

- Reliable
- User Friendly
- Easily Maintainable
- Easily incorporated with current saliva testing procedures
Specifications

• Size of general kiosk (~ 10 - 50 cubic feet)
• Temperature controlled sample storage environment (2° - 4° C)
• Returned Samples must be stored in plastic bags
• Test Tubes 0.44 cubic inches
• Shall hold 1,000 test tubes at maximum capacity
System Overview

User Taps "Begin"

- Open Bag Dispenser
  - Need Plastic Bag?
    - No
    - Open Collector Door
    - 10 Second Delay
    - Bag Collected?
      - No
        - Confirmation & Expected Results Time
      - Yes
        - Yes
          - Barcode Read?
            - Yes
              - Dispense Test Tube to User
            - No
              - No
                - Discard Test Tube
  - Yes
    - Purchase Test?
      - Yes
        - Choose Test
          - Enter Email & Purchase Test
          - Dispense Test Tube
          - Scan Barcode
          - Barcode Read?
            - Yes
              - Dispense Test Tube
            - No
              - No
                - Discard Test Tube
      - No
        - Exit Screen

Legend

- Optical Sensor:
- Solenoid:
- PCB:
- DC Motor:

Need Plastic Bag?

3.0" x 1.0"

Tongue

Collector Chute

Dispensing Chute

Discard Bin

Motor & PCB Mount

Touchscreen CPU

Boot

Roller

Collector Chute

Bag Dispenser

11"

20"

5.5"

47"

13"

1.5"

2.3"

1"

2.3"

Holder

Exit Screen
System Block Diagram

Legend:
- **Responsible Member**
  - Jesse
  - Jimmy
  - Jon
  - Zach

**Abbreviation**
- T.T. – Test Tube
- F.F. – Front Face

- CPU Test
- Microcontroller Test
- Tilter
- Barcode Scanner
- Web Database Design
- Touchscreen Interface Test
- Card Reader Acquire
- Dispensing Sensor Test
- Tilter Sensor Test
- Bi-Directional Motor Driver Test
- Temp. Sensor Test
- Dispensing Sensor Test
- A/C Controller Design
- Opto-Isolated Motor Enable Test
- Solenoid Lock (F.F. Collector Door) Test
- Capacity Sensor Test

Legend:
- **Responsible Member**
  - Jesse
  - Jimmy
  - Jon
  - Zach

Abbriviation:
- T.T. – Test Tube
- F.F. – Front Face

- Motor Voltage Regulator Test
- Motor Voltage Test
- Scan Roller Motor Test
- Pacman Motor Test
- Tilter Motor Test
- Tilter Motor Test
- Solenoid Lock (F.F. Collector Door) Test
- Capacity Sensor Test
- Solenoid Lock (F.F. Collector Door) Test
- Temp. Sensor Test
- A/C Controller Design
- T.T. Collector
- T.T. Collector

- Serial Bus
- T.T. Dispenser
- T.T. Holder
- Capacity Sensor Design
- Solenoid Lock (F.F. Collector Door) Test
- T.T. Holder
- Temp. Sensor Test
- A/C Controller Design
- Bag Dispenser
- Bag Dispenser
- Solenoid Lock (F.F. Collector Door) Test
- Capacity Sensor Test
- Solenoid Lock (F.F. Collector Door) Test
- Temp. Sensor Test
- A/C Controller Design
- T.T. Holder
- Capacity Sensor Design
- Solenoid Lock (F.F. Collector Door) Test
- T.T. Collector
- T.T. Collector
- Serial Bus

- System Block Diagram
Dispenser Functional Diagrams

T.T. Dispenser

- Boot Capacity Sensor Test
- Tilter Sensor Test
- Dispensing Sensor Test
- Bi-Directional Motor Driver Test
- Barcode Scanner Acquire
- Tilter Motor Test
- Scan Roller Motor Test
- Motor Voltage Regulator Test
- Pacman Motor Test

Legend
- Test Tube: ○
- DC Motor Shaft: ●

- Doc Box Outer Wall
- Pacman Roller
Motor Choice

- Low RPM
- Compact housing
- Used for Rollers and Tilter
- Bi directional

**DC Motor:**
- Linear Voltage-RPM relationship
- Little frequency effect
- Simple to control
Motor Choice

Sayama 12SM-AT3

• 58 RPM$_{\text{max}}$
• 12 VDC$_{\text{max}}$
• 20mA (no-load)
• 12mm dia. x 35 mm len.
Roller Motor Control

Variable Speed Motor Driver with Opto-Isolated Embedded Control

Truth Table

<table>
<thead>
<tr>
<th>MSP_out</th>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Motor</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 V</td>
<td>On</td>
<td>Off</td>
<td>On</td>
<td>On</td>
</tr>
<tr>
<td>0 V</td>
<td>Off</td>
<td>On</td>
<td>Off</td>
<td>Off</td>
</tr>
</tbody>
</table>

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rb1</td>
<td>7.5kΩ</td>
</tr>
<tr>
<td>R2</td>
<td>100Ω</td>
</tr>
<tr>
<td>R3</td>
<td>100Ω</td>
</tr>
<tr>
<td>R4</td>
<td>500Ω</td>
</tr>
<tr>
<td>R5</td>
<td>7.5kΩ</td>
</tr>
<tr>
<td>R6</td>
<td>1kΩ</td>
</tr>
<tr>
<td>R1</td>
<td>200Ω</td>
</tr>
<tr>
<td>Motor</td>
<td>600Ω</td>
</tr>
</tbody>
</table>
Motor Voltage Regulation

Parameters:

\[ \begin{align*}
C_1 &= 10 \mu F \\
C_2 &= 10 \mu F \\
R_5 &= 7.5 \Omega \\
R_v &= 10 k \Omega \text{ Pot} \\
R_b &= 100 \Omega \\
R_z &= 141 \Omega
\end{align*} \]
## Tilter Motor Control

<table>
<thead>
<tr>
<th>Motor Driver Considerations</th>
<th>L298N</th>
<th>SN754410NE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Driver</td>
<td>L298N</td>
<td>SN754410NE</td>
</tr>
<tr>
<td>Mounting</td>
<td>Through Hole</td>
<td>Through Hole</td>
</tr>
<tr>
<td>Pins</td>
<td>15</td>
<td>16</td>
</tr>
<tr>
<td>Max Op. Temp.</td>
<td>130 C</td>
<td>85 C</td>
</tr>
<tr>
<td>Max Cont. Current</td>
<td>2A</td>
<td>1.1A</td>
</tr>
<tr>
<td>Output Voltage Range</td>
<td>4.8V - 46V</td>
<td>4.5V - 36V</td>
</tr>
<tr>
<td>Price</td>
<td>4.67</td>
<td>1.95</td>
</tr>
</tbody>
</table>
Optical Sensors

<table>
<thead>
<tr>
<th>Component</th>
<th>Value/Model</th>
<th>Effective Wavelength</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value/Model</td>
<td>500Ω</td>
<td>850nm</td>
</tr>
<tr>
<td>Effective Wavelength</td>
<td>2.5x10³ Ω</td>
<td>700nm – 1000nm</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Component</th>
<th>R1</th>
<th>R2</th>
<th>LED</th>
<th>Phototransistor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value/Model</td>
<td>500Ω</td>
<td>2.5x10³ Ω</td>
<td>SFH4550</td>
<td>OFT-5301</td>
</tr>
<tr>
<td>Effective Wavelength</td>
<td>--</td>
<td>--</td>
<td>850nm</td>
<td>700nm – 1000nm</td>
</tr>
</tbody>
</table>

The diagram shows a circuit with the components labeled as follows:
- uController Output Buffer
- R1
- R2
- LED
- Phototransistor

The graph displays the uController Input and R2 Voltage with the following data points:

<table>
<thead>
<tr>
<th>Value</th>
<th>Mean</th>
<th>Min</th>
<th>Max</th>
<th>Std Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.00 V</td>
<td>3.28</td>
<td>3.28</td>
<td>3.28</td>
<td>0.00</td>
</tr>
<tr>
<td>2.00 V</td>
<td>120mV</td>
<td>120m</td>
<td>120m</td>
<td>0.09</td>
</tr>
</tbody>
</table>

The diagram also includes traces labeled 1 and 2.
Collector

Collector Return / Capacity Sensor with Enable
Author: Jesse Easterling

Parameters

<table>
<thead>
<tr>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
<th>R6</th>
<th>R7</th>
<th>R8</th>
</tr>
</thead>
<tbody>
<tr>
<td>40Ω</td>
<td>1kΩ</td>
<td>10kΩ Pot</td>
<td>1kΩ</td>
<td>51kΩ</td>
<td>18kΩ</td>
<td>51kΩ</td>
<td>5kΩ</td>
</tr>
</tbody>
</table>

Truth Table

<table>
<thead>
<tr>
<th>CO_E</th>
<th>X1</th>
<th>CLT_S</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Diagram of the Collector Return / Capacity Sensor with Enable.
Bag Dispenser

Objective
• Provide required storage for sample vials

Goals
• Accessible via electronic lock
• Actuate Lock many times without Overheating
• Feedback to the CPU when out of bags which will disable dispense and return processes.

Specifications

<table>
<thead>
<tr>
<th>Specifications</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max Unlock time</td>
<td>~20s</td>
</tr>
<tr>
<td>Min Lock off time</td>
<td>~3s</td>
</tr>
<tr>
<td>Unlocking Duty Cycle</td>
<td>&gt;87%</td>
</tr>
</tbody>
</table>
Bag Dispenser

- spring plate
- bags
- linear solenoid
- hatch door
- spring hinge
- outer wall
- empty supply switch
Door Lock Schematic

- V12bus (12V)
- V5bus (5V)
- R1 (100Ω)
- K1
- Flywheel (1N4149)
- Q2 (Lock Relay)
- D1 (1N4149)
- K2
- 2N2222A

Connections:
- MSP430_Signal
- msp430
- 3.3V Key = Space
- R5 (1.8kΩ)
- R2 (10kΩ)
- 0.74 H 579 Ω
- 0.74 H 2.88 Ω

Solenoid
Holder

Objective
• Provide temperature controlled environment for sample storage.

Solution
• Repurpose Mini-Fridge as the Cooling Unit
• Designed an adjustable thermostat

Goals
• Adjustable set temperature
• Precise holding temperature
• Accurate temperature reading

Specifications

<table>
<thead>
<tr>
<th>Specification</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintain Temperature of</td>
<td>~7°C (44.6°F)</td>
</tr>
<tr>
<td>Adjustable Set Temperature</td>
<td>0°C to ~20°C</td>
</tr>
<tr>
<td>Temp. Reading Accuracy</td>
<td>&lt;1°C</td>
</tr>
<tr>
<td>Holding Temp. Accuracy</td>
<td>~±1°C</td>
</tr>
</tbody>
</table>
# Temperature Sensor

<table>
<thead>
<tr>
<th></th>
<th>LM35CAZ</th>
<th>LM50BIM3</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sensor Temp. Range</strong></td>
<td>±0.2°C</td>
<td>±2°C</td>
</tr>
<tr>
<td><strong>Accuracy Range</strong></td>
<td>-55°C to +135°C</td>
<td>-40°C to +125°C</td>
</tr>
<tr>
<td></td>
<td>10mV/1°C</td>
<td></td>
</tr>
<tr>
<td><strong>Supply Range</strong></td>
<td>4V to 30V</td>
<td>4V to 10V</td>
</tr>
<tr>
<td><strong>Price</strong></td>
<td>~$5.00</td>
<td>~$1.50</td>
</tr>
</tbody>
</table>

- Linear Centigrade Output makes reading from easy – no calibration.
- Both have range within our means.
- Choose LM35 for better accuracy.
Temperature Control

- Powers on compressor when temperature is above high threshold and cuts power off when below the low threshold.
- Once power is cutoff, will remain off for about 5 minutes before being able to be power on again to prevent damage to the compressor.
**Comparator**

- Chose LM293 Precision Rail-to-rail comparator with 5V supply voltage.
- Avoid chatter with external Hysteresis.
- For 20mV (~2°C) Hysteresis we use below equations to calculate R2/R1 ratio of about 250.

\[
V_{TH} = \frac{(R_1 + R_2)V_{REF} - R_1V_L}{R_2}
\]

\[
V_{TL} = \frac{(R_1 + R_2)V_{REF} - R_1V_H}{R_2}
\]

Hysteresis = \(V_{TH} - V_{TL}\)
Choosing an Embedded Processor

• RS-232 Interface
• Sufficient I/O
• Low-Power
• Low-Cost
# Embedded Hardware Control Specifications

<table>
<thead>
<tr>
<th>Component</th>
<th>I/O Function(s)</th>
<th>Pins Needed</th>
<th>Component</th>
<th>I/O Function(s)</th>
<th>Pins Needed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dispenser Wheel Motor</td>
<td>General Output</td>
<td>1</td>
<td>Tilt Motor Sensor</td>
<td>General Input</td>
<td>1</td>
</tr>
<tr>
<td>Roller Motor</td>
<td>General Output</td>
<td>1</td>
<td>Bag Quantity Sensor</td>
<td>General Input</td>
<td>1</td>
</tr>
<tr>
<td>Tilt Motor</td>
<td>General Output</td>
<td>3</td>
<td>Return / Cap. Sensor</td>
<td>General Input</td>
<td>1</td>
</tr>
<tr>
<td>Collect Unlock Solenoid</td>
<td>General Output</td>
<td>1</td>
<td>Temperature Sensor</td>
<td>ADC Input</td>
<td>1</td>
</tr>
<tr>
<td>Bag Unlock Solenoid</td>
<td>General Output</td>
<td>1</td>
<td>Dispense Sensor Enable</td>
<td>General Output</td>
<td>1</td>
</tr>
<tr>
<td>T.T. Quantity Sensor</td>
<td>General Input</td>
<td>1</td>
<td>Return Sensor Enable</td>
<td>General Output</td>
<td>1</td>
</tr>
<tr>
<td>Vend Sensor</td>
<td>General Input</td>
<td>1</td>
<td>RS-232 Serial Interface</td>
<td>UART</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Total General I/O: 14</th>
<th>Total ADC Inputs: 1</th>
<th>Total Serial I/O: 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total I/O: 17</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Texas Instruments MSP430 G2553IPW28R

- 28-pin TSSOP package
- Three 8-bit I/O ports
- 10-bit ADC
- Supports external 32kHz crystal oscillator
- Supports in-system programming
- Free Samples Available
5V - 3.3V Inverting Level Translator

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_b$</td>
<td>64.5,k\Omega</td>
</tr>
<tr>
<td>$R_1$</td>
<td>1,k\Omega</td>
</tr>
<tr>
<td>$h_{FE1}$</td>
<td>100</td>
</tr>
<tr>
<td>$h_{FE2}$</td>
<td>100</td>
</tr>
</tbody>
</table>

Truth Table

<table>
<thead>
<tr>
<th>Input</th>
<th>Q1</th>
<th>$V_i$</th>
<th>$I_c$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 V</td>
<td>On</td>
<td>113 mV</td>
<td>2.9 mA</td>
</tr>
<tr>
<td>0 V</td>
<td>Off</td>
<td>3.3 V</td>
<td>46 pA</td>
</tr>
</tbody>
</table>
3.3v to 5v Non-Inverting Output Level Translator

Truth Table

<table>
<thead>
<tr>
<th>MSP_out</th>
<th>Q1</th>
<th>Q2</th>
<th>V01</th>
<th>V02</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3 V</td>
<td>On</td>
<td>Off</td>
<td>158mV</td>
<td>5V</td>
</tr>
<tr>
<td>0 V</td>
<td>Off</td>
<td>On</td>
<td>4.94 V</td>
<td>169 mV</td>
</tr>
</tbody>
</table>

Parameters:

- Rb1 = 7.5kΩ
- R1 = 200Ω
- R2 = 190Ω
- Rb2 = 13kΩ
Serial Communication

- Opto-Isolated Signals
- Inverts RS-232 logic for input to MSP430
- Converts RS-232 signals voltages to valid MSP430 signal voltages
- Converts MSP430 signal voltages to valid RS-232 voltages
- Additional IC not needed (ie. MAX232)

### RS-232 Signals Used

<table>
<thead>
<tr>
<th>Signal</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 GND</td>
<td>Signal Ground</td>
</tr>
<tr>
<td>4 DTR</td>
<td>Data Terminal Ready</td>
</tr>
<tr>
<td>3 TXD</td>
<td>Transmit Data</td>
</tr>
<tr>
<td>2 RXD</td>
<td>Receive Data</td>
</tr>
</tbody>
</table>
Tx/Rx Signal Voltages

RS232 Signal
- From computer
- Tx line ± 8.5v

Isolated MSP430 input
- Rx pin 0 - 3v

MSP430 output
- Tx pin 0 – 3v

Isolated RS232 Signal
- To computer
- Rx line ± 6v

<table>
<thead>
<tr>
<th></th>
<th>Value</th>
<th>Mean</th>
<th>Min</th>
<th>Max</th>
<th>Std Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Max</td>
<td>5.40</td>
<td>5.40</td>
<td>5.40</td>
<td>0.00</td>
</tr>
<tr>
<td>2</td>
<td>Min</td>
<td>-6.20</td>
<td>-6.18</td>
<td>-6.20</td>
<td>60.3m</td>
</tr>
<tr>
<td>1</td>
<td>Max</td>
<td>8.50</td>
<td>8.45</td>
<td>8.50</td>
<td>93.4m</td>
</tr>
<tr>
<td>1</td>
<td>Min</td>
<td>-8.50</td>
<td>-8.50</td>
<td>-8.50</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>Visual Studio</td>
<td>GTK+</td>
<td>Qt</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------</td>
<td>------------------</td>
<td>------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Licensing</td>
<td>Proprietary</td>
<td>Open Source</td>
<td>Open Source/ Commercial</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Documentation</td>
<td>Good</td>
<td>Mediocre</td>
<td>Excellent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Language</td>
<td>C++/C#/.NET</td>
<td>C/C++</td>
<td>C++ extended w/MOC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cross-Platform</td>
<td>No</td>
<td>Yes</td>
<td>Inherently</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Community Support</td>
<td>Yes</td>
<td>Mediocre</td>
<td>Unsurpassed</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Application Flow

- Each box represents a form the user interacts with.
- A single user decision/input is done per form much like a wizard application.
- Application consists of three paths starting and returning to the main window.
  1. “How it Works”
  2. “Return Sample”
  3. “New Sample”
Logic Classes

**QSqlDatabase**
- QSqlDatabase()
- ~QSqlDatabase()
- +addDatabase(QString)
- close()
- databaseName()
- isOpen()
- lastError()
- open()
- primaryIndex()
- record()
- setDatabaseName(QString)
- setHostName(QString)
- setPassword(QString)
- setUserName(QString)
- tables()

**QSqlQuery**
- QSqlQuery(QString)
- ~QSqlQuery()
- bindValue(int pos, const QVariant & value)
- first()
- last()
- next()
- previous()
- record()
- seek()
- setForwardOnly()

**QIODevice**
- close()
- open(QIODevice::OpenMode)
- portName()
- setBaudRate(int)
- setDataBits(int)
- setFlowControl(Qt::FlowControl)
- setParity(Qt::Parity)
- setPortName(QString)
- setStopBits(int)

**QSerialPort**
- QSerialPort()
- ~QSerialPort()
- readData()
- writeData()
- enum BaudRate
- enum DataBits
- enum FlowControl
- enum Parity
- enum StopBits

**QSerialPortInfo**
- availablePorts()
- isBusy()
- portName()

**QSqlDatabase**
- QSqlDatabase()
- ~QSqlDatabase()
- +addDatabase(QString)
- close()
- databaseName()
- isOpen()
- lastError()
- open()
- primaryIndex()
- record()
- setDatabaseName(QString)
- setHostName(QString)
- setPassword(QString)
- setUserName(QString)
- tables()

**QSqlQuery**
- QSqlQuery(QString)
- ~QSqlQuery()
- bindValue(int pos, const QVariant & value)
- first()
- last()
- next()
- previous()
- record()
- seek()
- setForwardOnly()

**Customer**
- int id
- QString email
- Customer()
- ~Customer()
- int id()
- QString email()
- setEmail()
- setId()
GUI Classes

QWidget

Base Class

Wizard

int: pageCount
QList<QWidget>: pages
QStackedWidget: stack

Wizard()
~Wizard()
addPage(&Page)
doNext(int)
doPrevious(int)
goToPage(int)

QWidget

PanelPage

QPushButton
QLabel
QListView
Etc.

RequestTest

QWidget: PanelPage
QWidget: Options
QWidget: CheckoutReview
QWidget: EmailInput
QWidget: Payment
QWidget: Dispensing
QStackedWidget: descriptions

RequestTest()
~RequestTest()
finished()

QWidget

Return

QWidget: CustomerNotFound
QWidget: EmailInput
QWidget: ReturnReview
QWidget: SlotOpen
QWidget: SampleReceived

Return()
~Return()
finished()

QWidget

MoreInfo

QWidget: Information

MoreInfo()
~MoreInfo()
finished()
Logistics

Customer information must be safely associated with sample tubes

- Sample tubes are marked with a unique barcode using Code128 symbology
- Customer identified through personal email address
- Upon order placement, sample tubes are scanned before being dispensed to the customer
- Customer and barcode information are stored in a database
- Results are only available via personal email or home address

*Taken from Wikimedia Commons*
## Power Requirements

<table>
<thead>
<tr>
<th>Component</th>
<th>+12V</th>
<th>-12V</th>
<th>+5V</th>
<th>+3.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Motor Interface X2</td>
<td></td>
<td></td>
<td></td>
<td>50 mA</td>
</tr>
<tr>
<td>Motor Driver</td>
<td>90 mA</td>
<td>11.4 mA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Collector Sensor</td>
<td>9 mA</td>
<td>9 mA</td>
<td>60 mA</td>
<td></td>
</tr>
<tr>
<td>Tilt Motor + Ctrl</td>
<td>120 mA</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lock Solenoid x2</td>
<td>3 A</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lock Ctrl</td>
<td>200 mA</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Temperature Circuit</td>
<td>110 mA</td>
<td>30 mA</td>
<td>40 mA</td>
<td></td>
</tr>
<tr>
<td>Input Buffer</td>
<td></td>
<td></td>
<td></td>
<td>18 mA</td>
</tr>
<tr>
<td>Output Buffer X4</td>
<td></td>
<td></td>
<td>200 mA</td>
<td></td>
</tr>
<tr>
<td>MSP430G2553 @ 16MHZ</td>
<td></td>
<td></td>
<td></td>
<td>5 mA</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>3.53 A</strong></td>
<td><strong>50.4 mA</strong></td>
<td><strong>350 mA</strong></td>
<td><strong>23 mA</strong></td>
</tr>
</tbody>
</table>
Power Supply

General CPU Power Supply

- +12V @ 18A
- -12V @ 0.8A
- +5V @ 25A
- +3.3V @ 25A
PCB Schematic Layout
PCB Software and Manufacturing

CadSoft EAGLE for design
• Easy to use
• Professional Version Provided

Advanced Circuits for Manufacturing
• $33 Per board (under 60 in²)
• Free Online DFM check
• Reputable & Decent Turnaround
### Project Budget

#### Projected Budget (Fall 2013)

<table>
<thead>
<tr>
<th>Description</th>
<th>Price</th>
<th>Qty.</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Circuit Components</td>
<td>42.25</td>
<td>-----</td>
<td></td>
</tr>
<tr>
<td>Power Supply</td>
<td>15.67</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Dual-Core PC</td>
<td>174.99</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Touch Screen Display</td>
<td>426.99</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Serial Card</td>
<td>14.99</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Tamper-proofing Software</td>
<td>19.95</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>58 RPM Brushed DC Motor</td>
<td>12.95</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Switch SPDT Roller Action</td>
<td>8.64</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Mechanical Systems Supplies</td>
<td>50</td>
<td>-----</td>
<td></td>
</tr>
<tr>
<td>System Enclosure</td>
<td>1200</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>4x8 Insulation Panel</td>
<td>25</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Refrigeration System</td>
<td>250</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

**Total Cost:** 2275.97

#### Current Budget (Spring 2014)

<table>
<thead>
<tr>
<th>Description</th>
<th>Price</th>
<th>Qty.</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Circuit Components</td>
<td>70</td>
<td>-----</td>
<td></td>
</tr>
<tr>
<td>PCB</td>
<td>33</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Power Supply</td>
<td>20</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Dual-Core PC</td>
<td>174.99</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Touch Screen Display</td>
<td>50</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Serial Card</td>
<td>14.99</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>58 RPM Brushed DC Motor</td>
<td>12.95</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Switch SPDT Roller Action</td>
<td>8.64</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Mechanical Systems Supplies</td>
<td>60</td>
<td>-----</td>
<td></td>
</tr>
<tr>
<td>System Enclosure</td>
<td>1200</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>4x8 Insulation Panel</td>
<td>25</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Refrigeration System</td>
<td>Free</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

**Total Cost:** 1737.11
Current Progress

- **Research**: 98% complete
- **Design**: 80% complete
- **Hardware Testing**: 50% complete
- **Software Testing**: 40% complete
- **% Complete**: 67% complete
Completion Plan

- Mechanical Systems Development
- PCB Testing
- Software Designing/Testing
- Cooling System
¿Tienen Preguntas?