**MINISTRY OF EDUCATION AND TRAINING**

**FPT UNIVERSITY**

Capstone Project Document

**Vietnamese Sign Language Recognition**

|  |  |
| --- | --- |
| **Group 03** | |
| **Group members** | Nguyễn Hữu Kỳ Long – Team leader – SE60984  Nguyễn Đình Tân – Team member – SE61115  Nguyễn Xuân Ý – Team member – SE60869  Lê Phương Bình – Team member – SE61049 |
| **Supervisor** | Mr. Đỗ Đức Minh Quân |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** | VSLR |

-Ho Chi Minh City, 18/06/2015-

*This page is intentionally left blank*

# Table of Contents

[Table of Contents 3](#_Toc426972944)

[List of Tables 3](#_Toc426972945)

[Definitions, Acronyms, and Abbreviations 4](#_Toc426972946)

[A. Report No. 4 Software Design Description 5](#_Toc426972947)

[1. Design Overview 5](#_Toc426972948)

[2. System Architectural Design 5](#_Toc426972949)

[3. Component Diagram 6](#_Toc426972950)

[4. Detailed Description 7](#_Toc426972951)

[4.1 Software Detailed Description 7](#_Toc426972952)

[1. User Interface Design 37](#_Toc426972953)

[1.1 Subtract Background Color 37](#_Toc426972954)

[1.2 Recognize Hand Sign Language 37](#_Toc426972955)

[1.3 Learn Hand Sign Language 38](#_Toc426972956)

[1.4 Select 38](#_Toc426972957)

[1.5 Hardware Detailed Description 39](#_Toc426972958)

[2. Algorithms 48](#_Toc426972959)

# List of Tables

[Table 1: Entity Data Dictionary 61](#_Toc417274779)

[Table 2: Detail Data Dictionary 63](#_Toc417274780)

# Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| **Name** | **Definition** |
| VSLR | Vietnamese Sign Language Recognition |
|  |  |
|  |  |
|  |  |

# Report No. 4 Software Design Description

## Design Overview

* This document describes the technical and user interface design of VSLR System. It includes the architectural design, the detailed design of common functions and business functions and the design of database model.
* The architectural design describes the overall architecture of the system and the architecture of each main component and subsystem.
* The detailed design describes static and dynamic structure for each component and functions. It includes class diagrams, class explanations and sequence diagrams for each use cases.
* The database design describes the relationships between entities and details of each entity.
* Document overview:
* Section 2: gives an overall description of the system architecture design.
* Section 3: gives component diagrams that describe the connection and integration of the system.
* Section 4: gives the detail design description, which includes class diagram, class explanation, and sequence diagram to details the application functions.
* Section 5: describe a fully attributed ERD.

## System Architectural Design

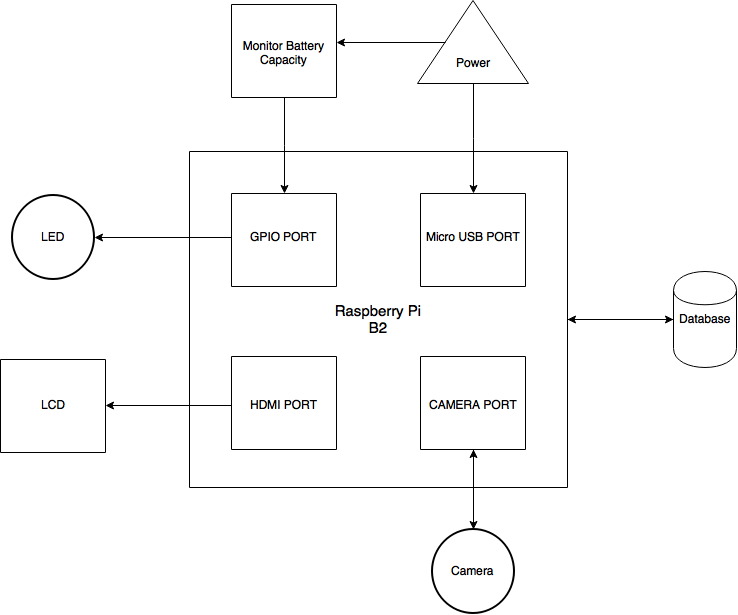


Figure 1: VSLR System Architectural



## Component Diagram

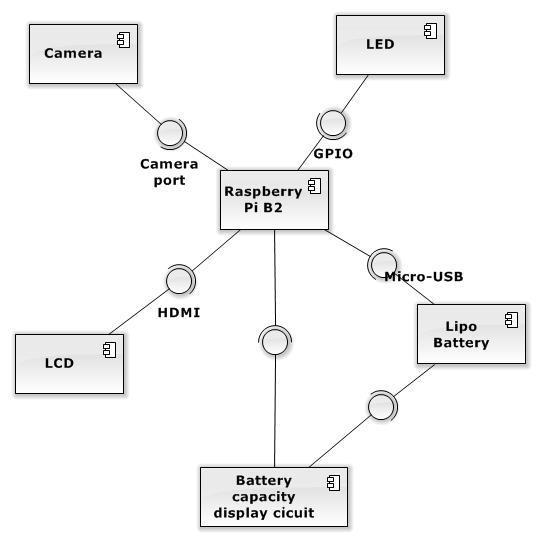


Figure 2: Component Diagram

## Detailed Description

### **Software Detailed Description**

#### Class Diagram



Figure 3: Class Diagram

|  |  |
| --- | --- |
| Class Dictionary : Class Description | |
| Class Name | Description |
| HandSignDTO | This is data transfer object class of table HandSign encapsulating HandSign attributes. |
| HandSigns | This class contains list of hand sign loaded from database and methods for getting attributes of a specific hand sign. |
| BackgroundTimerThread | This class inherits QThread class that is used for sending signals to notify, count down timer and move between steps in background color subtraction phase.  This thread will stop if the phase testing background color subtraction succeeds. |
| RetreivingFrameThread | This class inherits QThread class that is used for activing camera and then retrieving images captured from camera continuously.  This thread will run until the application stops. |
| ShowingImageThread | This class inherits QThread class that is used for getting image from RetreivingFrameThread when it finish , processing these image to subtract background color which will show on the interface.  This thread will run until the application stops. |
| ImageProcessingThread | This class inherits QThread class that receives image subtracted background color to detect hands, create the binary image containing hand features and then extract the features from these image.  This thread receives signals asking to perform image processing for testing background subtraction, learning function and recognition function.  This thread just is enable to process when it receives image from ShowingImageThread.  This thread will run until the application stops. |
| HandGesture | This class encapsulates attributes, methods to create binary images containing hand features and methods to output features related to height, hand palm and finger lines. |
| ExtractingBinaryImage | This class provides methods which outputs histogram features of a binary image. |
| SignRecognition | This class encapsulates attributes and methods to perform SVM algorithm for hand sign recognition. |
| Speech | This class encapsulates attributes and methods to initiate some parameters for using espeak library to read a content in sound via LCD speaker. |
| RecognitionContent | This class encapsulates attributes and methods to manage the whole recognition content for hand sign language recognition function. |
| RecognitionTimerThread | This class inherits QThread class that is used for implementing the real time timer to manage hand sign recognition function.  This thread will start when the application runs but it is just enable to process when the recognition function is selected.  This thread will stop until the application stops. |
| BatteryThread | This class inherits QThread class that will run during the application works to check battery capacity and send signals notifying user. |
| LowBatteryDialog | This class is used for initiating dialog interface which shows low battery announcement. |
| LowBatteryTimerThread | This class inherits QThread processes as real time timer to manage the time showing LowBatteryDialog. |
| MainWindow | This class is the main UI thread which manages the application interfaces and creates communications between thread objects. |

#### Class Diagram Explanation

##### HandSigns

Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| signsMap | QMap<int, HandSignDTO> | Private | This map contains hand sign records loaded from HandSign table. |

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| getAllSigns | QStringList | Private | Returns all records signsMap hold. |
| getLinkBySelectedIndex | QString | Private | Return record by index user select in combobox. |
| getMeanByID | QString | Private | Return record by recognition result returned from SVM. |

##### HandSignDTO

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | int | Private | Unique identifier of a word and recognition result number. |
| content | QString | Private | Content of a particular sign. |
| link | QString | Private | Path to directory contain image of hand sign. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| SetId | void | Public | Set value of attribute id |
| GetId | int | Public | Get attribute id value |
| SetLink | void | Public | Set value of attribute link |
| GetLink | QString | Public | Get attribute link value |
| SetContent | void | Public | Set value of attribute content |
| GetContent | QString | Public | Get attribute content value |

##### 

##### BackgroundTimerThread

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| STOP | bool | Public | This variable is used for to stop the thread before the application is closed. |
| mutex | QMutex | Private | It is to protect section of code so that only one thread can access at a time. |
| isEnableToCountDown | bool | Private | This variable is to check whether this thread continues counting down timer. |
| testingResult | bool | Private | This variable is to held background subtraction testing result and it is condition to stop this thread. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| run | void | Private | This is starting point for the thread and this method implements functions of the thread. |
| sendSignalChangingToHandDetectionMode | void | Private | This is signal which is sent Image Processing thread to move to hand detection step. |
| sendSignalChangingToTestingTime | void | Private | This is signal which is sent Image Processing thread to move to background subtraction testing step. |
| sendSignalFinishingColorSubtraction | void | Private | This is signal which is sent with testing result to notify main thread. |
| sendSignalGetTestingResult | bool | Private | This is signal which is sent Image Processing thread to get background subtraction testing result and stop testing step. |
| sendSignalChangingLabelNotice | void | Private | This is signal which is sent with notify content to main thread. |
| sendSignalFailTesingResult | void | Private | This is signal which is sent to notify main thread of testing result. |
| continueCountDown | void | Private | This is function that is called in response to signal which enables to continue counting down timer. |

##### 

##### SignRecognition

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| svmModel | struct svm\_model | Private | SVM will load the model file to this struct. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| getRecognitionResult | double | Public | This method receives the hand sign features, then uses SVM library to predict recognition result and return it. |

##### Speech

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| bufferLength | int | Private |  |
| options | int | Private |  |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| speak | void | Public |  |

##### RecognitionTimerThread

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| STOP | bool | Public | This variable is used for to stop the thread before the application is closed. |
| mutex | QMutex | Private | It is to protect section of code so that only one thread can access at a time. |
| isEnableWorking | bool | Private | This variable is to check whether the thread can process functions. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| enableWorking | void | Public | This is function that is called in response to signal which enables to perform functions. |
| run | void | Private | This is starting point for the thread and this method implements function counting down timer. |
| sendSignalUpdatingContent | void | Protected | This is signal which is sent to notify main thread of performing updating recognition content. |
| sendSignalCountDownRecognitionTime | void | Protected | This is signal which is sent main thread the real time timer. |

#### 

##### RetrievingFrameThread

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| STOP | bool | Public | This variable is used for to stop the thread before the application is closed. |
| frame | Mat | Private | This is object holding image captured from camera. |
| camera | VideoCapture | Private | This is object which actives camera working and retrieves image captured from camera to frame object. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| run | void | Private | This is starting point for the thread and this method implements retrieving images continuously. |
| changingFrame | void | Protected | This is signal which is sent to notify Showing Image thread that new image is retrieved. |

##### 

##### RecognitionContent

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| content | QString | Private | This is the whole recognition content will be updated every 3 seconds. |
| previousWord | QString | Private | This object holds the last updated hand sign. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| updateContent | QString | Public | This method receives new hand sign recognition result and then performs updating content function. |
| clear | void | Public | This method is used for clear the current content. |

##### LowBatteryTimerThread

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| STOP | bool | Public | This variable is used for to stop the thread before the application is closed. |
| mutex | QMutex | Private | It is to protect section of code so that only one thread can access at a time. |
| isEnableWorking | bool | Private | This variable is to check whether the thread can process functions. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| enableWorking | void | Public | This is function that is called in response to signal which enables thread can perform functions. |
| run | void | Private | This is starting point for the thread and this method implements retrieving images continuously. |
| sendSignalCountingDown | void | Protected | This is signal which is sent the real time timer to low battery dialog. |
| sendSignalClosingDialog | void | Protected | This is signal which is to notify main thread of closing current low battery dialog. |

#### 

##### BatteryThread

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| STOP | bool | Public | This variable is used for to stop the thread before the application is closed. |
| mutex | QMutex | Private | It is to protect section of code so that only one thread can access at a time. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| run | void | Private | This is starting point for the thread and this method implements retrieving images continuously. |
| sendSignalShowingBatteryCapacity | void | Protected | This is signal which is to notify main thread of current battery capacity every 5 minutes. |

#### 

##### LowBatteryDialog

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
|  |  |  |  |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| countDownTimer | void | Public | This method is used for changing the timer which is shown on the low battery dialog interface. |

##### HandGesture

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| contours | vector<vector<Point> > | Public | This holds sets of points as the contours of possible hands. |
| cMaxId | int | Public | This variable is to determines which contour is of right hand. |
| boundingRect | Rect | Public | This is rectangle bounding the contour of the right hand. |
| approxContour | vector<Point2f> | Public | This is contour of the hand in the form of set of float points after it is approximated polygonal curves. |
| biggestApporxContour | vector<Point > | Public | This is contour of the hand in the form of set of integer points after it is approximated polygonal curves. |
| hullI | vector<int> | Public | This is convex hull output. |
| defects | vector<Vec4i> | Public | This is convexity defects output. |
| correctDefects | vector<Point> | Public | This is convexity defects output after verifying which is correct. |
| inCircle | Point | Public | This is center location of hand palm. |
| inCircleRadius | double | Public | This is radius length of hand palm. |
| ratioHeightAndWidth | double | Public | This is height feature output of the hand sign. |
| isHand | bool | Public | This is output of detecting the hands. |
| nTotalLine | double | Private | This variable holds the number of finger lines. |
| totalLenLine | double | Private | This variable hold the total length of all finger lines. |
| n0To45Line | double | Private | This variable holds the number of finger lines belongs to 0 to 45 degree category. |
| totalLen0To45Line | double | Private | This variable hold the total length of all finger lines belongs to 0 to 45 degree category. |
| n46To90Line | double | Private | This variable holds the number of finger lines belongs to 46 to 90 degree category. |
| totalLen46To90Line | double | Private | This variable hold the total length of all finger lines belongs to 46 to 90 degree category. |
| n91To135Line | double | Private | This variable holds the number of finger lines belongs to 91 to 135 degree category. |
| totalLen91To135Line | double | Private | This variable hold the total length of all finger lines belongs to 91 to 135 degree category. |
| n136To180Line | double | Private | This variable holds the number of finger lines belongs to 136 to 180 degree category. |
| totalLen136To180Line | double | Private | This variable hold the total length of all finger lines belongs to 136 to 180 degree category. |
| nN1ToN45Line | double | Private | This variable holds the number of finger lines belongs to -45 to -1 degree category. |
| totalLenN1ToN45Line | double | Private | This variable hold the total length of all finger lines belongs to -45 to -1 degree category. |
| nN46ToN90Line | double | Private | This variable holds the number of finger lines belongs to -46 to -90 degree category. |
| totalLenN46ToN90Line | double | Private | This variable hold the total length of all finger lines belongs to -46 to -90 degree category. |
| nN91ToN135Line | double | Private | This variable holds the number of finger lines belongs to -91 to -135 degree category. |
| totalLenN91ToN135Line | double | Private | This variable hold the total length of all finger lines belongs to -91 to -135 degree category. |
| nN136ToN179Line | double | Private | This variable holds the number of finger lines belongs to -136 to -179 degree category. |
| totalLenN136ToN179Line | double | Private | This variable hold the total length of all finger lines belongs to -136 to -179 degree category. |
| whiteColor | Scalar | Private | White color is used to draw binary images. |
| blackColor | Scalar | Private | Black color is used to draw binary images. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| findBiggestContour | void | Public | This method is to find which contour is right. |
| detectIsHand | bool | Public | This method is to detect the biggest contour is contour of hand or not. |
| findInscribedCircle | void | Public | This method is to specify the hand palm. |
| drawPalmAndFingerLine | void | Public | This method is to draw two binary images such as hand palm and finger line. |
| getHeightFeatures | void | Public | This method returns the height feature of hand. |
| getRadiusFeatures | void | Public | This method calculates and returns radius features of hand palm. |
| getAngleFeatures | void | Public | This method calculates and returns angle features of finger lines. |
| getLinesFeatures | void | Public | This method calculate and returns classifier features of finger lines. |
| classifyLine | void | Public | This method classifies finger lines to 8 degree categories. |
| getSelectPoint | Point | Public | This method is to find the highest point of select hand sign and return it. |

#### 

##### ShowingImageThread

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| STOP | bool | Public | This variable is used for to stop the thread before the application is closed. |
| mutex | QMutex | Private | It is to protect section of code so that only one thread can access at a time. |
| BACKGROUND\_MODE | int | Private | This is a constant static variable holds value of background mode. |
| GETTING\_BACKGROUND\_MODE | int | Private | This is a constant static variable holds value of step sampling background. |
| HAND\_DETECTION\_MODE | int | Private | This is a constant static variable holds value of mode detecting hand. |
| isWorking | bool | Private | This variable is to check whether this thread is working. |
| frame | Mat | Private | This is image received from Retrieving Image thread. |
| blurMat | Mat | Private | This is image after blurring image process. |
| labMat | Mat | Private | This is image after converting BGR image into LAB image. |
| showMat | Mat | Private | This is image subtracted background and it is used for showing on the interface. |
| binMat | Mat | Private | This is binary image subtracted background. |
| mode | int | Private | This variable holds value of current mode. |
| IMAGE\_COLS | int | Private | This is constant static variable holds image’s width value. |
| IMAGE\_ROWS | int | Private | This is constant static variable holds image’s height value. |
| element | Mat | Private | This is structuring element is for morphological transformations. |
| LIGHT\_RANGE | double | Private | This is constant static variable holds range value of lightness. |
| WARM\_RANGE | double | Private | This is constant static variable holds range value of warm color-opponent. |
| COOL\_RANGE | double | Private | This is constant static variable holds range value of cool color-opponent. |
| whiteColor | Scalar | Private | White color is used to draw binary images. |
| blackColor | Scalar | Private | Black color is used to draw binary images. |
| lowerBoundArray | Scalar | Private | This array holds lower boundary of every single pixels. |
| upperBoundArray | Scalar | Private | This array holds upper boundary of every single pixels. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| setToDefaults | void | Public | This method sets this thread back to first step. |
| onChangingImage | void | Public | This is function that is called in response to signal, which receives new image captured from camera. |
| moveToHandDetectionMode | void | Public | This is function that is called in response to signal, which changes current mode to hand detection mode. |
| run | void | Private | This is starting point for the thread and this method implements processing these image to subtract background color which will show on the interface. |
| sendSignalShowingImage | void | Protected | This is signal which is to send images subtracted background color to main thread can show on the interface. |
| sendImageToExtractFeatures | void | Protected | This is signal which is to send images subtracted background color to Image Processing thread can extract features. |
| sendSignalEnableCountDown | void | Protected | This is signal which is to enable Timer thread to continue working. |

##### 

##### ExtractingBinaryImageFeatures

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
|  |  |  |  |

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return Type** | **Visibility** | **Descrition** |
| getFeature1of3VerticalAreas | void | Private | This method extracts 3 vertical area features of binary image. |
| getFeature1of3HorizontalAreas | void | Private | This method extracts 3 horizontal area features of binary image. |
| getFeatures4SquareAreas | void | Private | This method extracts 4 square area features of binary image. |
| getFeature4CornerAreas | void | Private | This method extracts 4 triangle area features of binary image. |
| getFeatures4x4 | void | Private | This method extract 16 square area features of binary image. |
| getFeatures3x3 | void | Private | This method extract 9 square area features of binary image. |

##### ImageProcessingThread

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| STOP | bool | Public | This variable is used for to stop the thread before the application is closed. |
| NO\_MODE | int | Public | This is a constant static variable holds value of no mode. |
| TESTING\_MODE | int | Public | This is a constant static variable holds value of testing background subtraction mode. |
| SELECTING\_MODE | int | Public | This is a constant static variable holds value of selecting function mode. |
| RECOGNITION\_MODE | int | Public | This is a constant static variable holds value of hand sign language recognition mode. |
| LEARNING\_MODE | int | Public | This is a constant static variable holds value of hand sign language learning mode. |
| SELECT\_SIGN | int | Public | This is a constant static variable holds recognition result of “select” hand sign. |
| SELECT\_SIGN\_SECOND | int | Public | This is a constant static variable holds recognition result of “select” hand sign. |
| END\_SIGN | int | Public | This is a constant static variable holds recognition result of “end” hand sign. |
| SPEAK\_SIGN | int | Public | This is a constant static variable holds recognition result of “speak” hand sign. |
| TEST\_SIGN | int | Public | This is a constant static variable holds recognition result of “test” hand sign. |
| mode | int | Private | This variable holds value of current mode. |
| testingResult | bool | Private | This is testing background color subtraction result at testing mode. |
| binaryMat | Mat | Private | This is hand binary image received from Showing Image thread. |
| frame | Mat | Private | This is image received from Showing Image thread which is subtracted background color. |
| croppedFrame | Mat | Private | This is hand images cropped from images subtracted background. |
| croppedHand | Mat | Private | This is cropped images which contains hand after adjusting size. |
| croppedBinHand | Mat | Private | This is cropped binary images which contains hand after adjusting size. |
| croppedInnerHand | Mat | Private | This is cropped binary images which contains hand palm after adjusting size. |
| croppedFingerLines | Mat | Private | This is cropped binary images which contains finger lines after adjusting size. |
| subROI | Mat | Private | This is region of hand image is to hold hand images cropped. |
| subInnerROI | Mat | Private | This is region of hand palm image is to hold hand palm cropped. |
| subFingerROI | Mat | Private | This is region of finger lines image is to hold finger line images cropped. |
| maskForFrame | Mat | Private | This is a mask which is used for cropping hand images. |
| maskForInner | Mat | Private | This is a mask which is used for cropping hand palm images. |
| maskForFingerLine | Mat | Private | This is a mask which is used for cropping finger line images. |
| cropImageSize | Size | Private | This is common size for every cropped images. |
| handGesture | HandGesture | Private | This is object processing images to create binary images and output features related to height, hand palm and finger lines. |
| extractingBinaryImageFeatures | ExtractingBinaryImageFeatures | Private | This is object processing binary images to output histogram features. |
| signRecogntion | SignRecognition | Private | This object is used for recognize hand sign. |
| isEnbaleProcessing | bool | Private | This variable is to check whether this thread can process images. |
| whiteColor | Scalar | Private | White color is to draw binary images containing features. |
| blackColor | Scalar | Private | Black color is to draw binary images containing features. |
| recognitionResultNumber | double | Private | This is recognition result predicted by SVM. |

##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| getRecognitionResult | double | Public | This method return hand sign recognition result. |
| recognizeSign | void | Private | This method implements steps to recognize hand sign. |
| changeToSelectingFunctionMode | void | Public | This method is to change current mode to selecting function mode. |
| changeToTestingMode | void | Public | This method is to change current mode to selecting function mode. |
| receiveBinaryImage | void | Public | This is function that is called in response to signal, which receives new image subtracted background color. |
| getTestingResult | bool | Public | This is function that is called in response to signal, which return testing background subtraction result. |
| run | void | Private | This is starting point for the thread and this method perform image processing for testing background subtraction, learning function and recognition function. |
| sendSignalToChangeLabelTestingResult | void | Protected | This is signal which is sent testing background color subtraction to main thread can update the interface. |
| sendSignalSelectingRecognition | void | Protected | This is signal which is sent notify main thread of changing current mode to recognition mode. |
| sendSignalSelectingLearning | void | Protected | This is signal which is sent notify main thread of changing current mode to learning mode. |
| sendSignalChangeToSelectingMode | void | Protected | This is signal which is sent notify main thread of changing current mode to selecting function mode. |
| sendSignalChangingRecognitionResult | void | Protected | This is signal which is sent notify main thread of outputting recognition result at recognition mode. |
| sendSignalMovingToUpperWord | void | Protected | This is signal which is sent notify main thread of “Lên” area is selected at learning mode. |
| sendSignalMovingToLowerWord | void | Protected | This is signal which is sent notify main thread of “Xuống” area is selected atlearning mode. |
| sendSignalChangingLearningResult | void | Protected | This is signal which is sent notify main thread of outputting recognition result at learning mode. |

#### 

##### MainWindow

##### Attribute

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| backgroundTimerThread | BackgroundTimerThread | Private | This object is to manage background color subtraction phase. |
| recognitionTimerThread | RecognitionTimerThread | Private | This object is to manage timer at recognition mode. |
| handSigns | HandSign | Private | This object is to output hand signs content. |
| recognitionSpeech | Speech | Private | This object is to speak recognition content. |
| recognitionContent | RecognitionContent | Private | This object is to manage the whole recognition content at recognition mode. |
| retrievingFrameThread | RetrievingFrameThread | Private | This is thread object which retrieves images captured from camera continuously. |
| showingImageThread | ShowingImageThread | Private | This is thread object which subtracts background color. |
| imageProccessingThread | ImageProccessingThread | Private | This is thread object which processes image to recognize hand sign during the application runs. |
| batteryThread | BatteryThread | Private | This is thread object which outputs battery capacity every 5 minutes. |
| lowBatteryTimerThread | LowBatteryTimerThread | Private | This object is to manage timer of Low Battery Dialog. |
| lowBatteryDialog | LowBatteryDialog | Private | This is UI thread of Low Battery Dialog. |

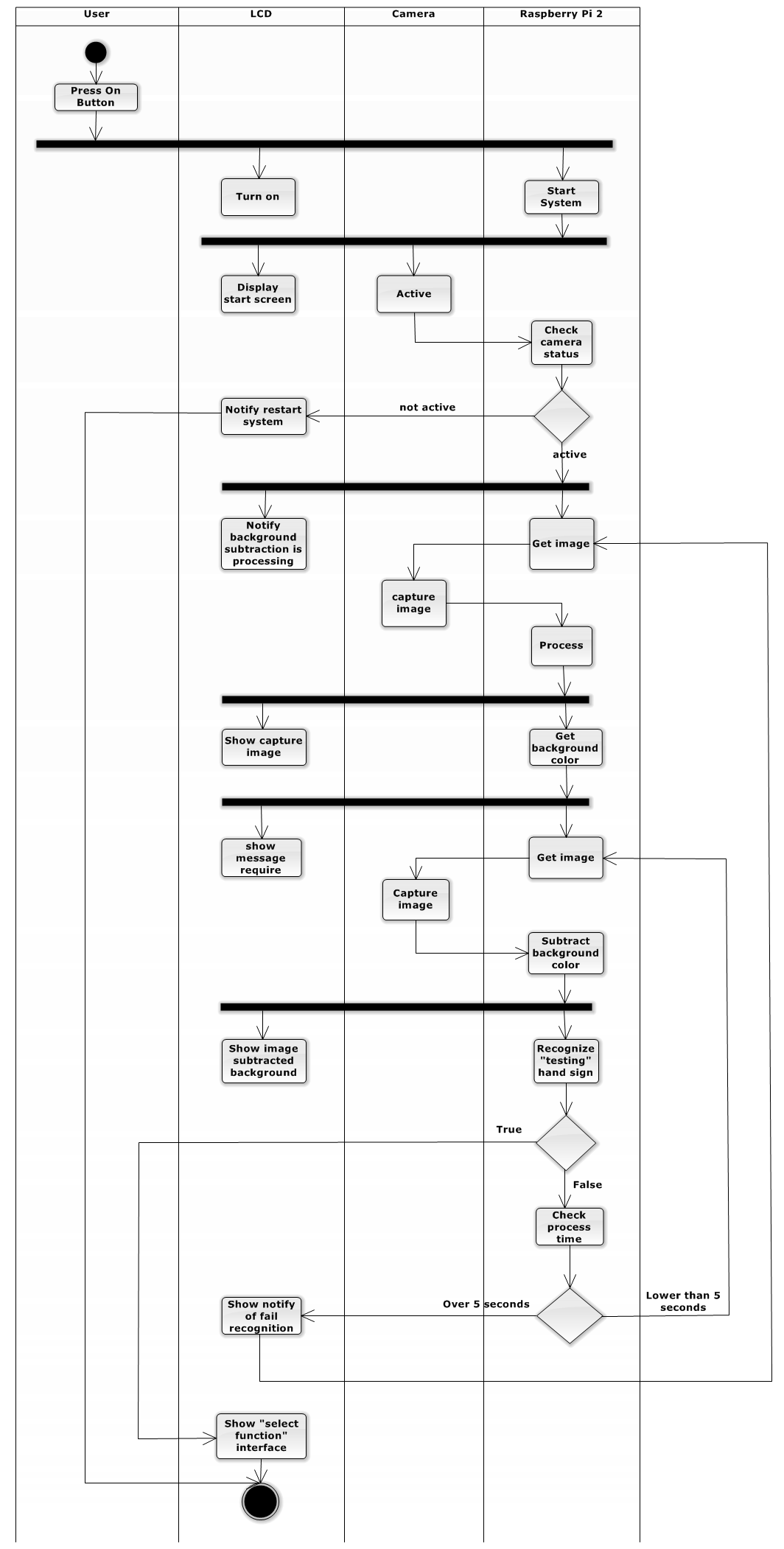
##### Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| onToShow | void | Private | This is function that is called in response to signal, which shows images on the interface. |
| changeLabelNotice | void | Private | This is function that is called in response to signal, which updates notify on the interface. |
| changeLabelTestingResult | void | Private | This is function that is called in response to signal, which updates testing background color subtraction on the interface. |
| countDownRecognitionTimer | void | Private | This is function that is called in response to signal, which updates real time timer on the interface at recognition mode. |
| changeRecognitionResult | void | Private | This is function that is called in response to signal, which updates recognition result on the interface at recognition mode. |
| updateRecognitionContent | void | Private | This is function that is called in response to signal, which updates new recognition result to the whole content and shows it on the interface at recognition mode. |
| changeToSelectingFunction | void | Private | This is function that is called in response to signal, which implements steps to change to selecting function mode. |
| changeToRecognitionFunciton | void | Private | This is function that is called in response to signal, which implements steps to change to recognition mode. |
| changeToLearningFunction | void | Private | This is function that is called in response to signal, which implements steps to change to learning mode. |
| onFinishingColorSubtraction | void | Private | This is function that is called in response to signal, which checks the testing background subtraction result. |
| moveToUpperWord | void | Private | This is function that is called in response to signal, which moves selection to upper word at learning mode. |
| moveToLowerWord | void | Private | This is function that is called in response to signal, which moves selection to lower word at learning mode. |
| changeLearningResult | void | Private | This is function that is called in response to signal, which updates recognition result on the interface at laerning mode. |
| showBatteryCapacity | void | Private | This is function that is called in response to signal, which shows battery capacity images on the interfaces. |
| closeLowBatteryDialog | void | Private | This is function that is called in response to signal, which closes low battery dialog. |
| initiateSelectingFunctionInterface | void | Private | This methods initiates components of selecting function mode on the interface. |
| initiateLearningInterface | void | Private | This methods initiates components of learning mode on the interface. |
| initiateRecognitionInterface | void | Private | This methods initiates components of recognition mode on the interface. |
| initiateColorSubtractionInterface | void | Private | This methods initiates components of selecting function mode on the interface. |
| changeImageByWordID | void | Private | This is function that is called in response to signal, which updates image of selected word on the interface at learning mode. |
| closeEvent | void | Protected | This is an override method which stops threads which is still working before closing application. |

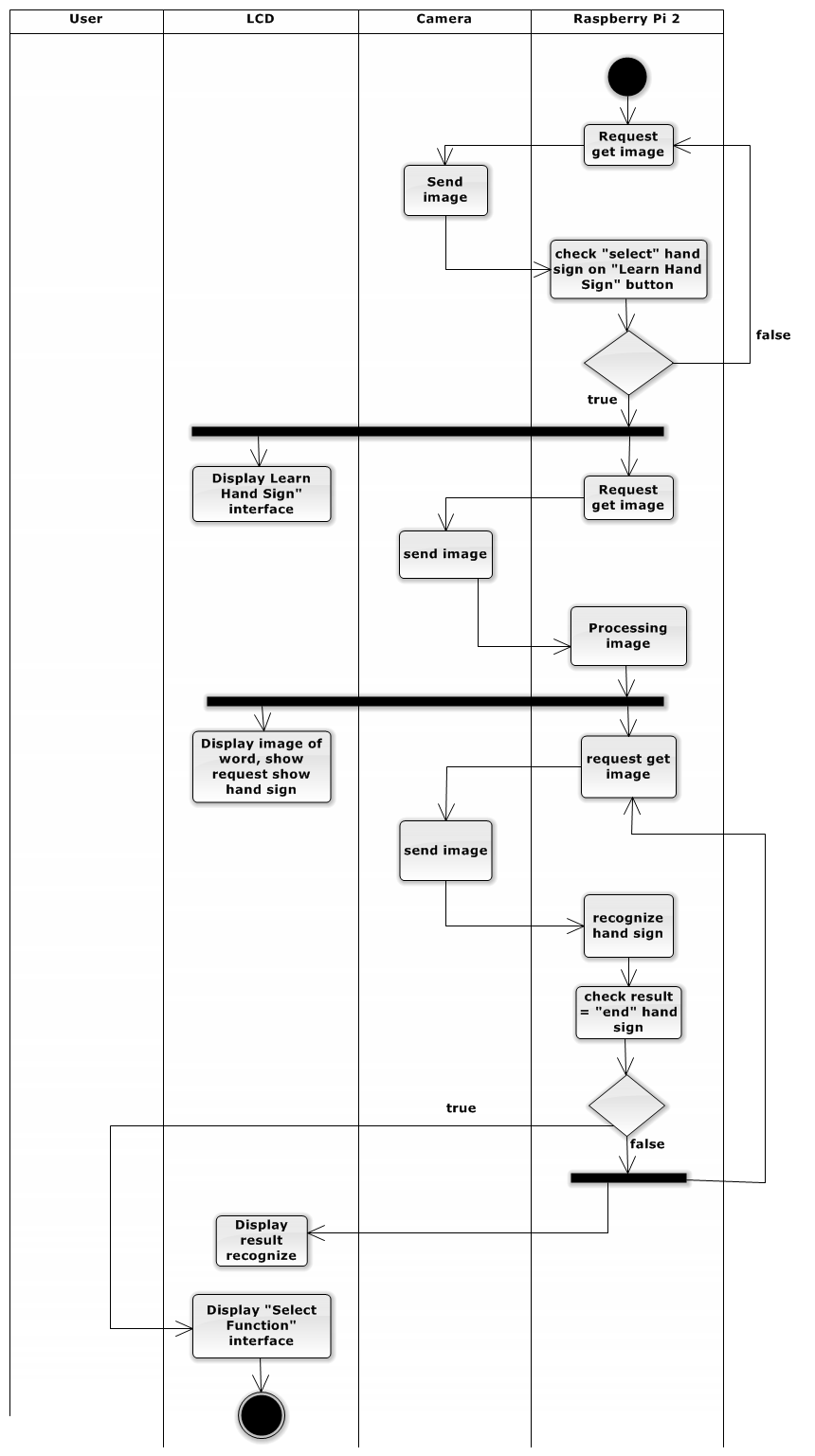
#### 

#### Activity Diagram

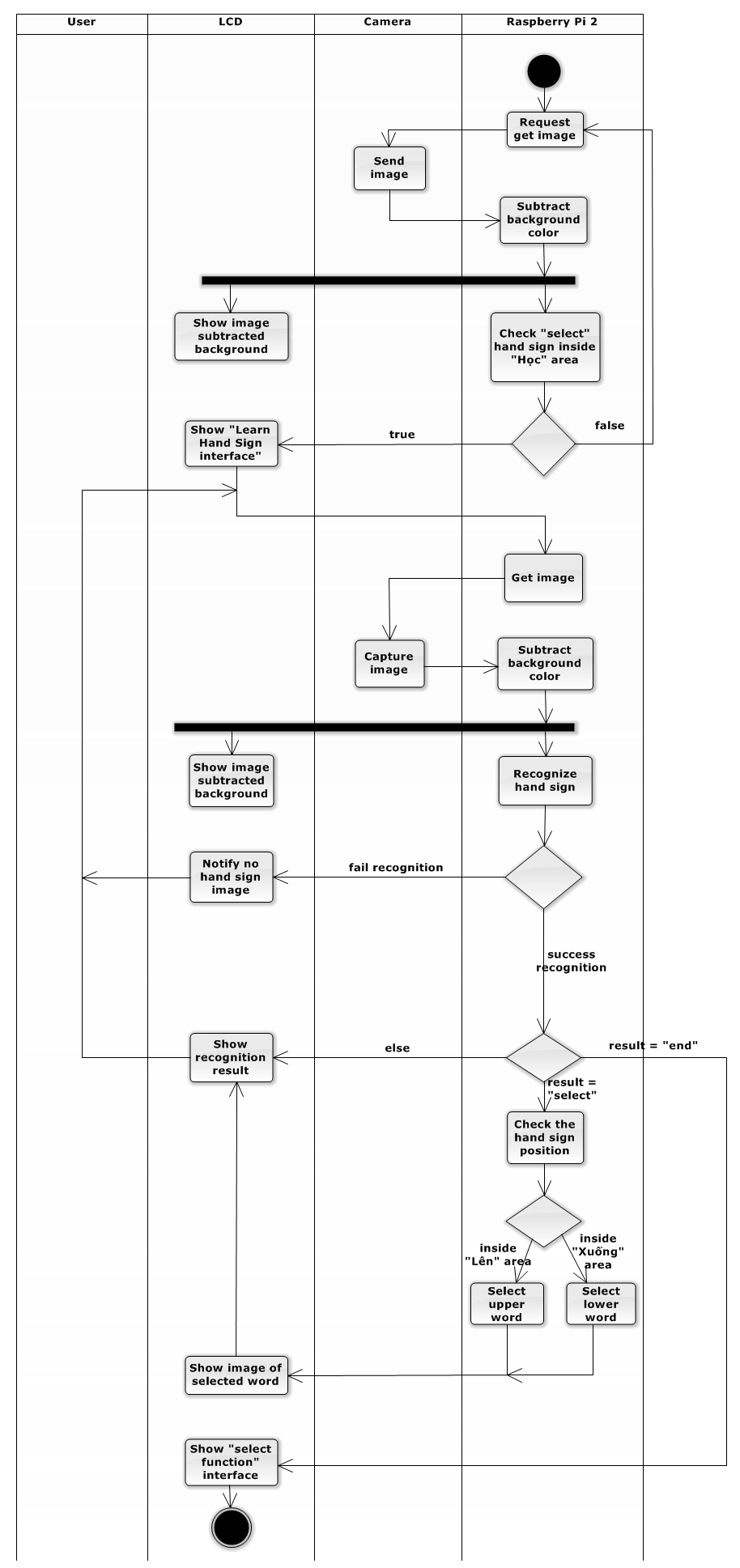
##### Background color subtraction



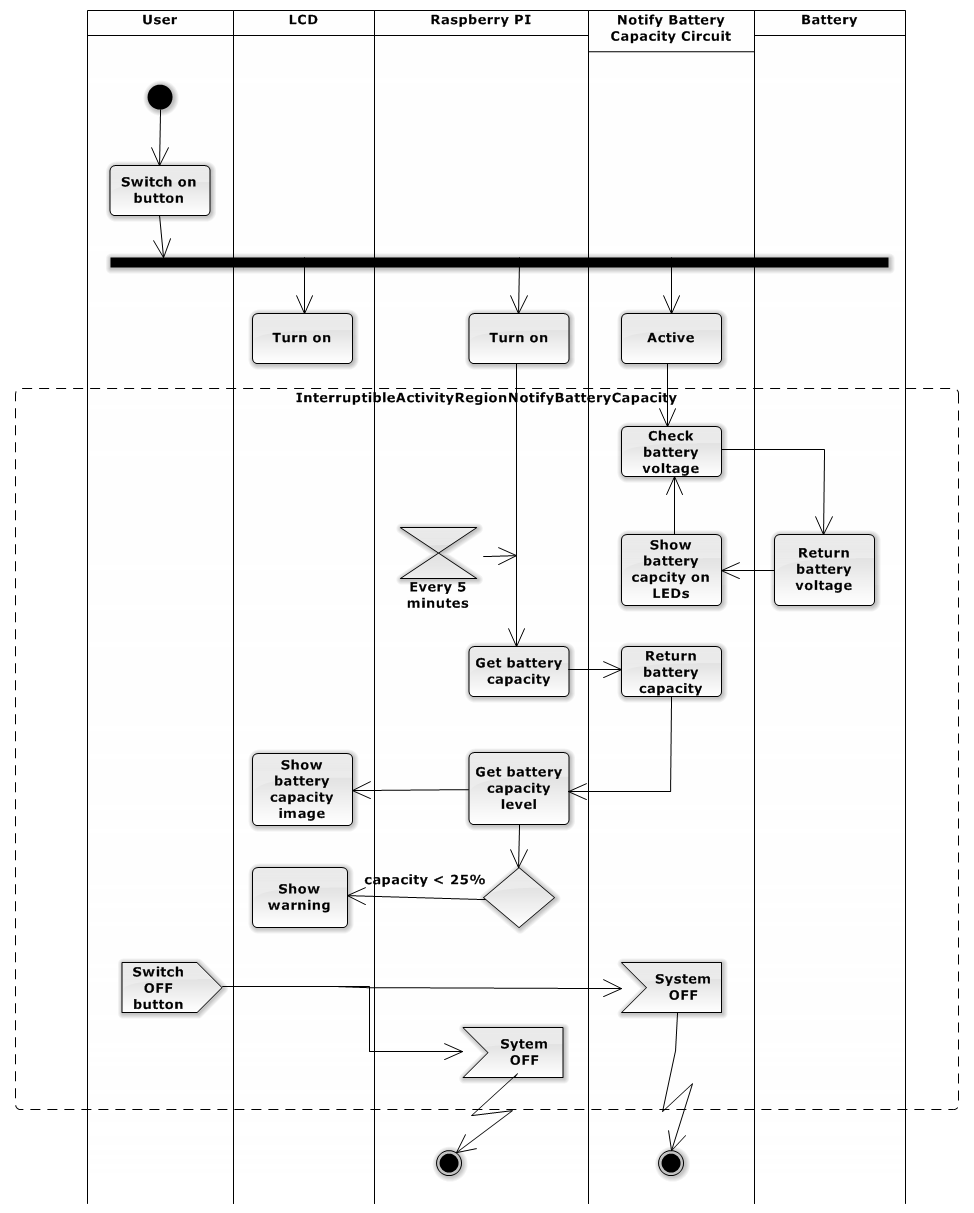
##### Recognize Hand Sign Language



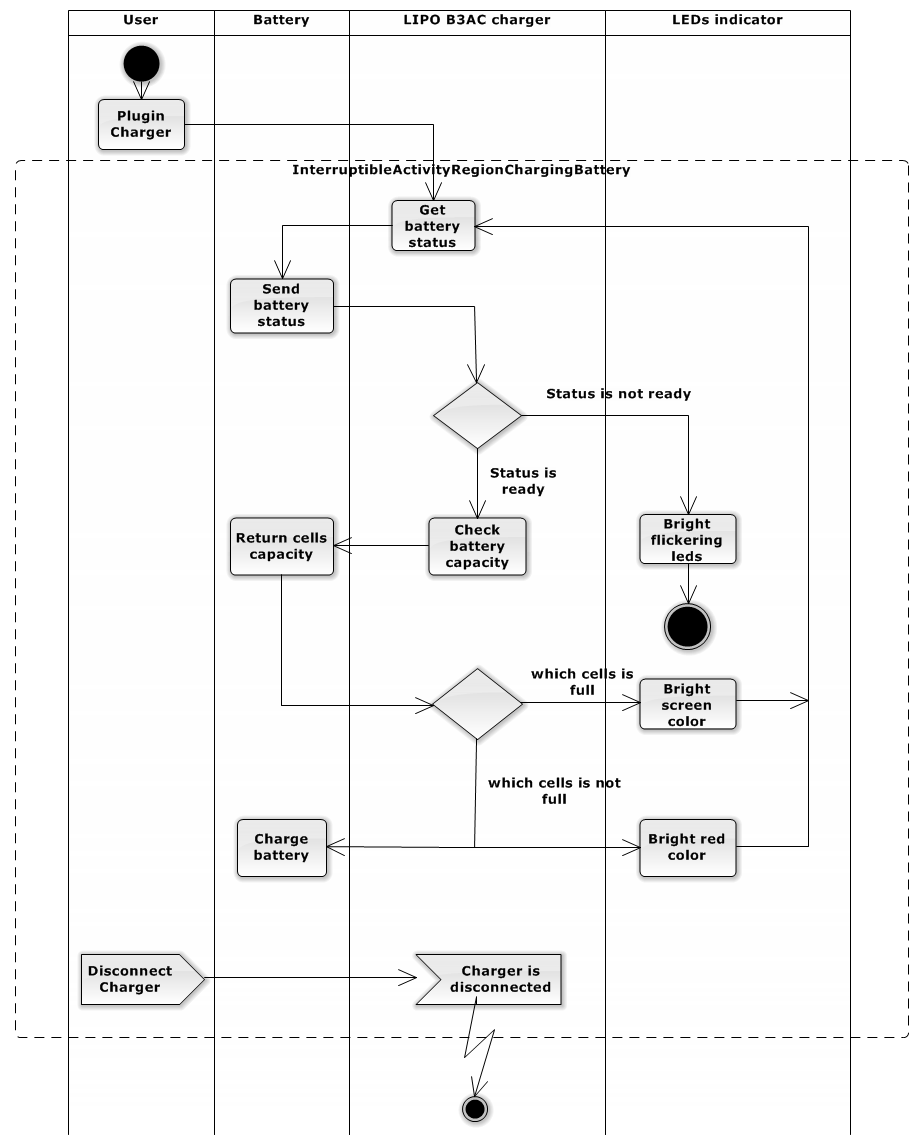
##### Learn Hand Sing Language



##### Notify Battery Capacity

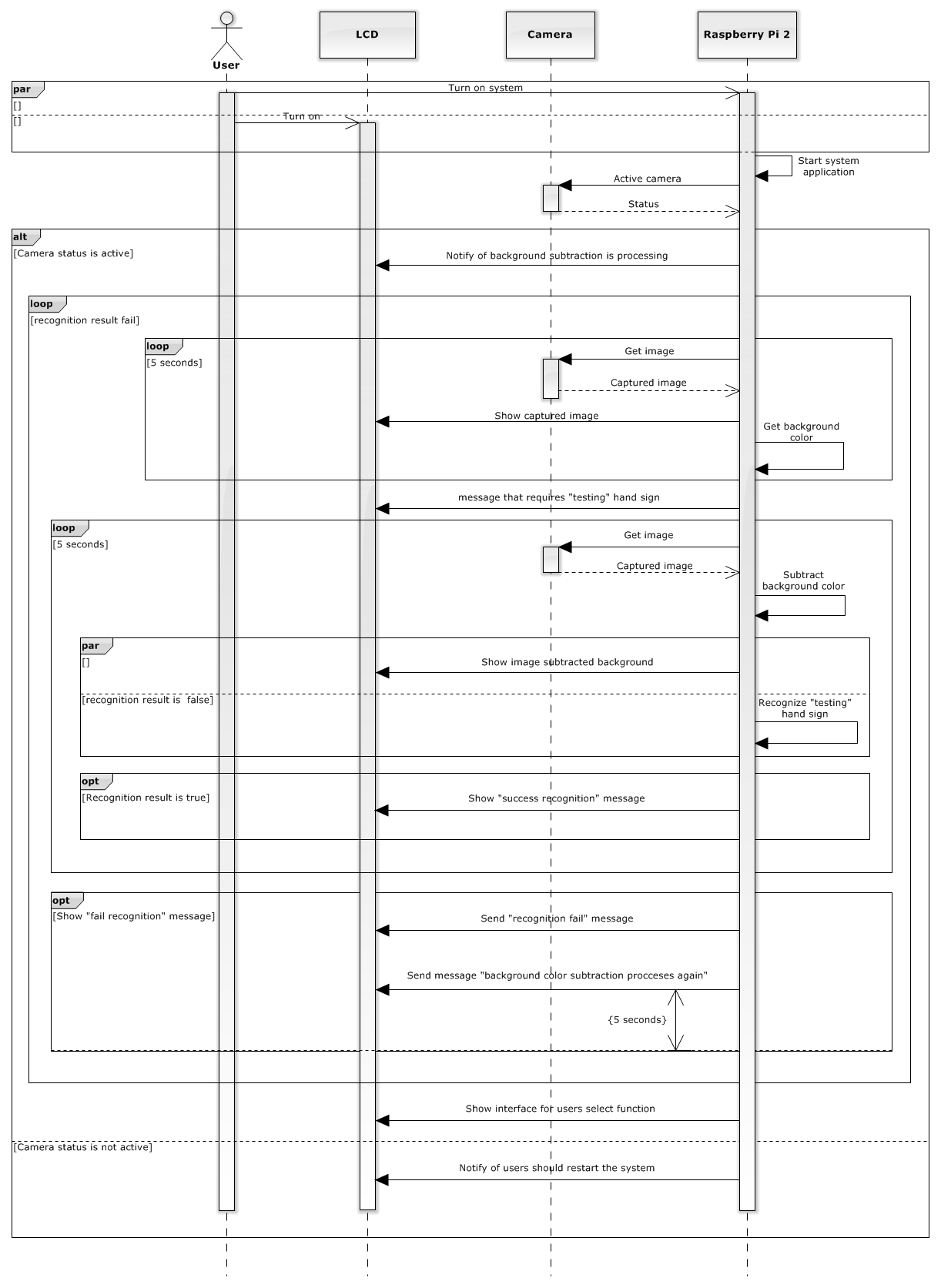


##### Charge Battery

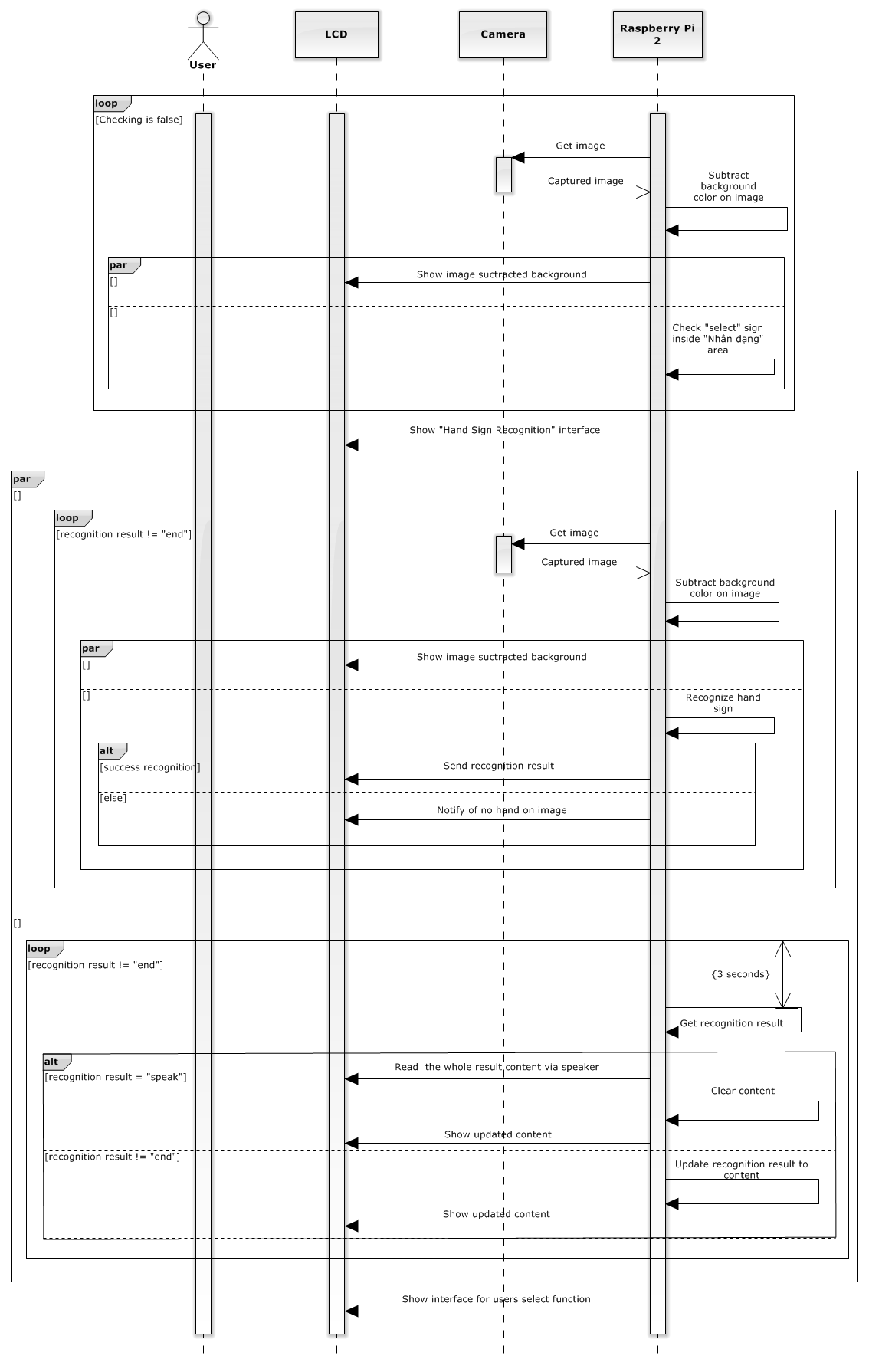


#### Sequence Diagram

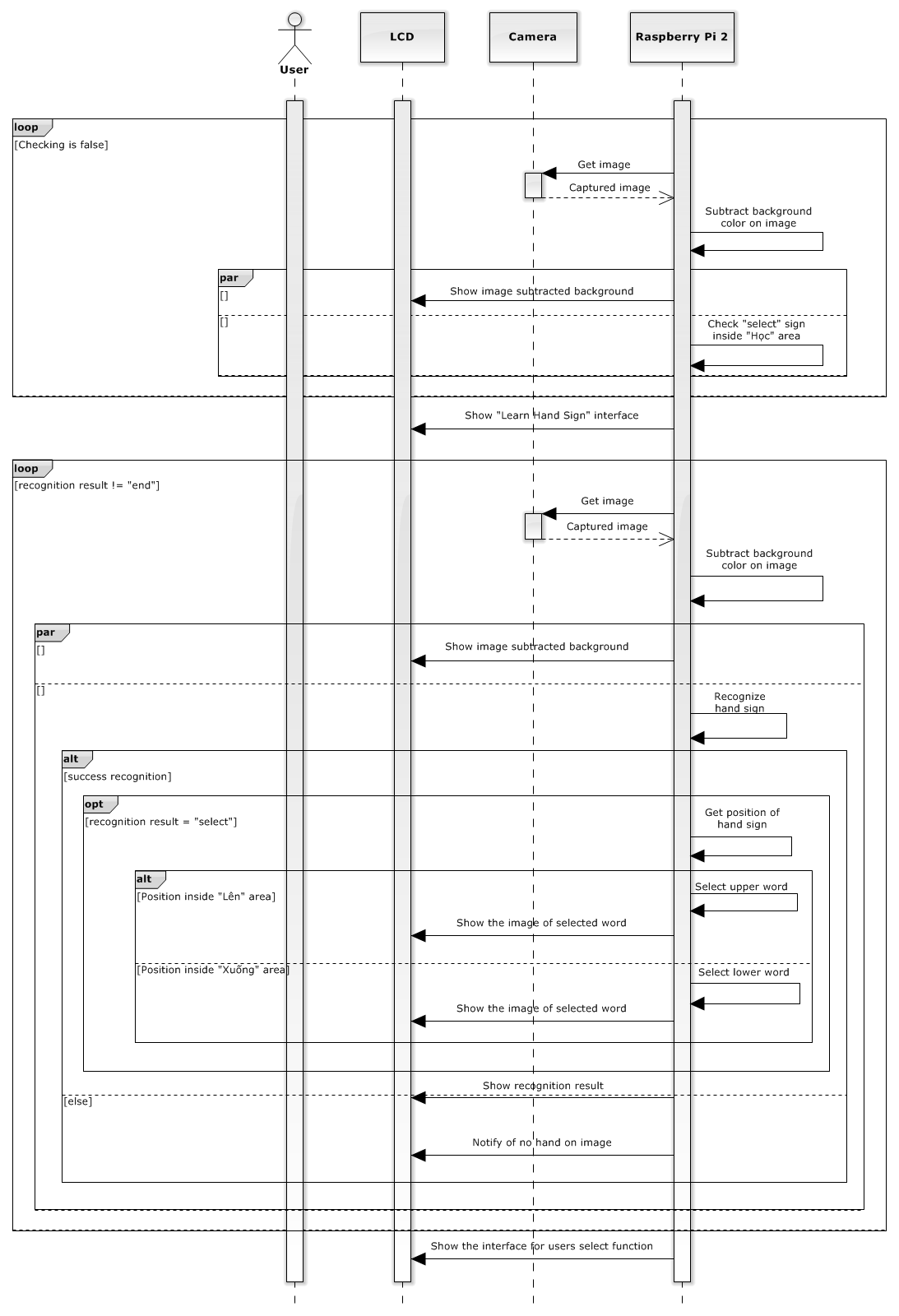
#### Subtract Background Color



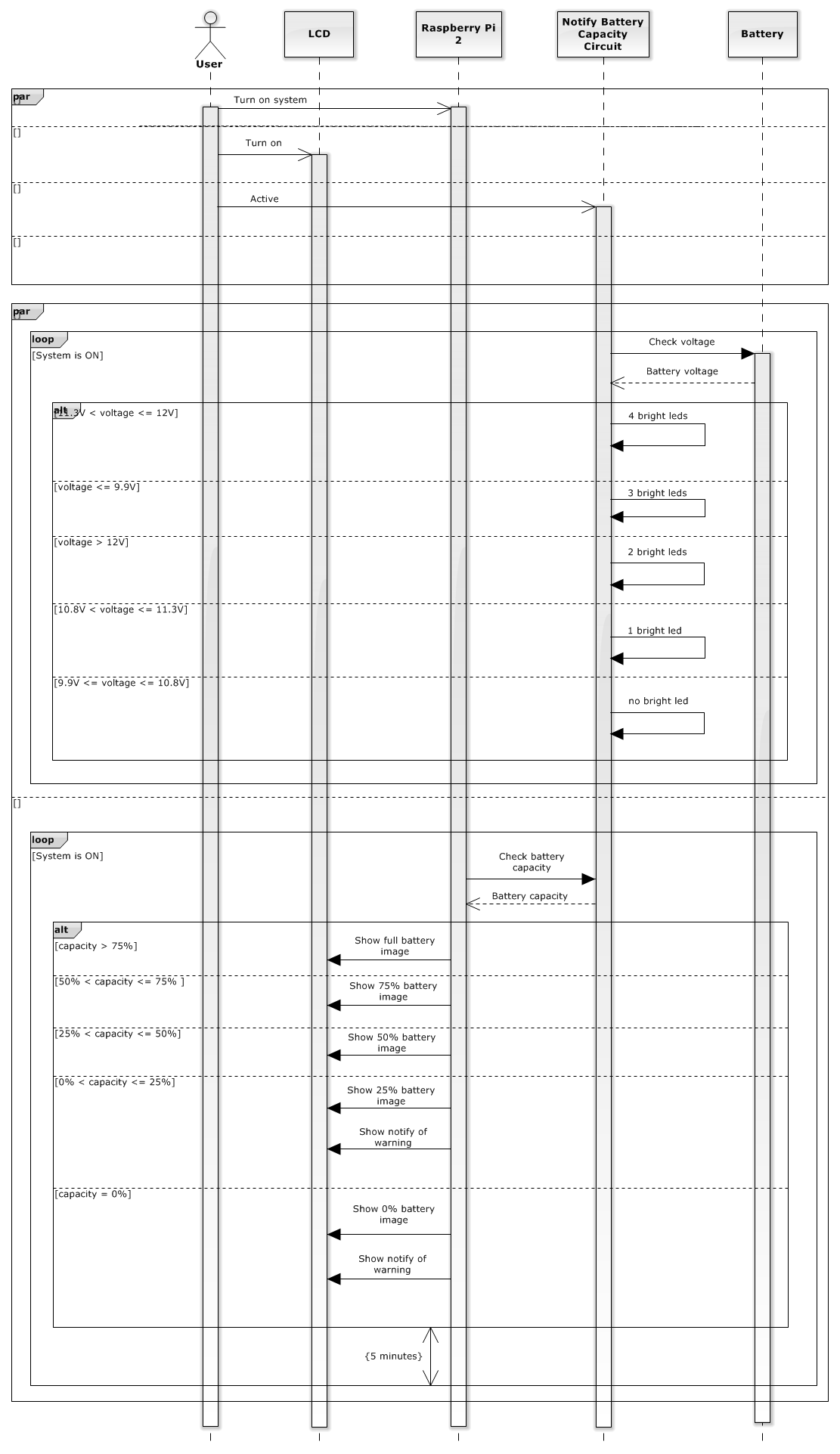
#### Recognize Hand Sign Language



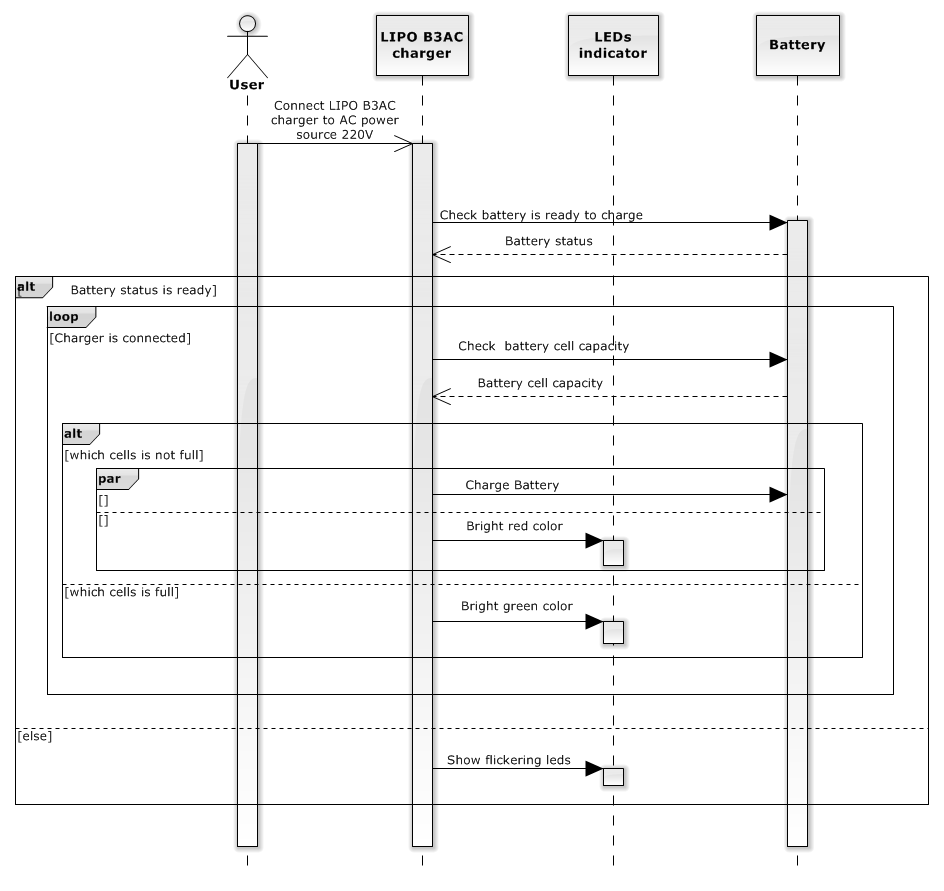
#### Learn Hand Sign Language



#### Notify Battery Capacity



#### Charge Battery



## User Interface Design

### Subtract Background Color



### Recognize Hand Sign Language



### Learn Hand Sign Language



### Select



### Hardware Detailed Description

#### Raspberry Pi B2



**Figure 33. Raspberry Pi B2 Kit**

The Raspberry Pi is a low cost, **credit-card sized computer** that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python. It’s capable of doing everything you’d expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and playing games.

***FOR***

* Quad-core CPU
* Backwards compatible
* More RAM
* Will fit existing cases

***AGAINST***

* Could prove intimidating for Linux newcomers
* No micro-USB adapter included

Raspberry Pi B2 specification:

SoC: Broadcom 2836  
CPU: Quad-core ARM7 800MHz  
GPU: Videocore IV 250MHz  
Memory: 1GB  
GPIO: 40 pin  
Ports: 4x USB 2.0, 100BaseT Ethernet, HDMI and MicroSD card  
Size: 85.60 × 56.5mm (about 3.2 x 2.1-inch)

#### Create a Portable System

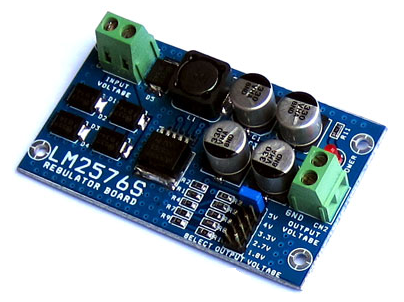
Raspberry use input is 5v and 1A so; we do not use battery AA or AAA for Raspberry. We choose lipo battery use for Raspberry Pi B2 kit because lipo battery supplies 11.1v and 1A.



But Raspberry use input is 5v and 1A so, we need a circuit transformer convert from 11.1v to 5v.

That reason why, we choose LM2576ADJ - 3A UNI REG Board.

**LM2576ADJ - 3A UNI REG Board**



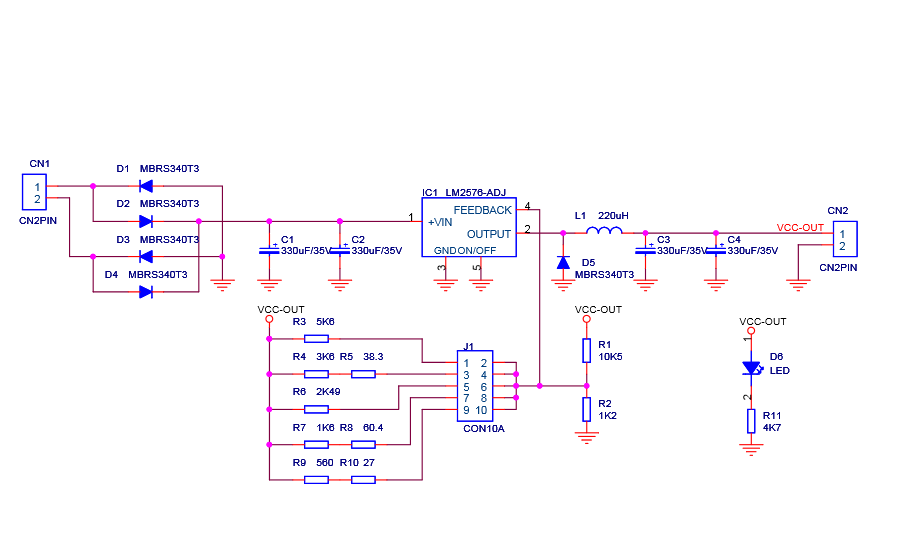
**Overview:**

* UNI-REG board allows changing voltages from 7-23V AC (or 9-32V DC) to 5V, 4V, 3.3V, 2.7V or 1.8V.
* Circuit Board using LM2576 - Step-Down Voltage Regulator.
* On-board screw-terminals are available for easy connection.

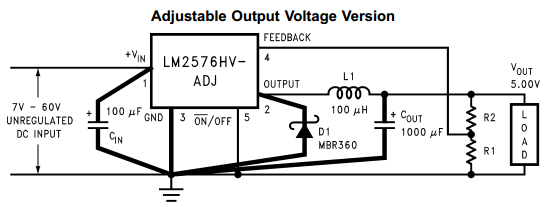
**Description:**

LM2576ADJ - UNI REG Board uses voltage regulator IC provides functional step-down (buck) switching regulator, capable of responding and changing load voltage lines are excellent. This is the ideal motherboard for projects requiring high voltage switching from lower to AC (DC).

Circuit Board accepts 7-23V AC input voltage (9-32V DC or), and stable output 5V, 4V, 3.3V, 2.7V or 1.8V DC, suitable for most electronics projects. The output voltage is selected via a jumper on the board. Compact and affordable, this board is perfect for use when switching power supplies are needed for your embedded project.



IC LM2576HV-ADJ



VOUT = VREF (1+ R2/R1)

R2 = R1(VOUT/VREF  - 1)

Where VREF = 1.23v, R1 between 1k and 5k

#### 4.2.3 The Battery Capacity Display Circuit

**Components of the circuit:**

- Battery lipo

- LT084

- 1 diode zener 5.1v (1N4733)

- 3 resistors 220 ohm

- 2 resistors 3.3k ohm

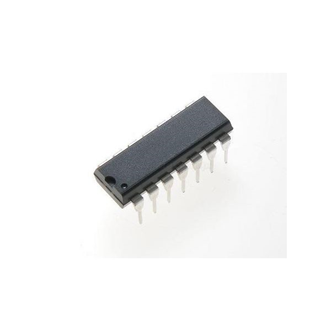
- 2 resistors 2.2k ohm

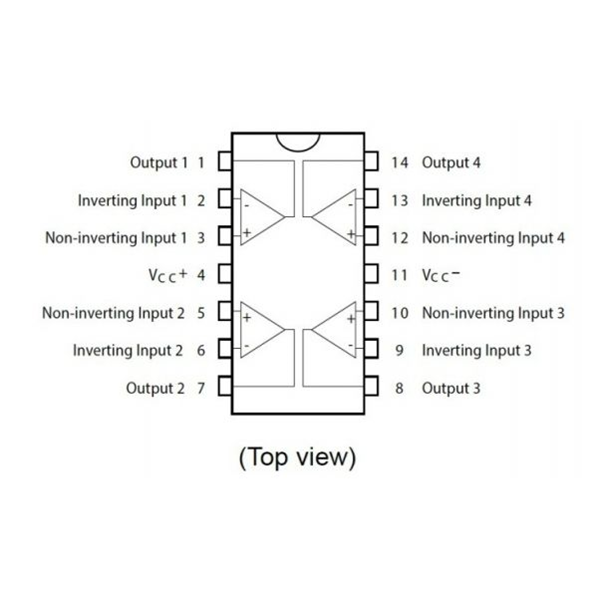
- 5 resistors 1k ohm

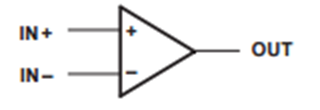
- 4 LED

**a. LT084**

The TL084 JFET-input operational amplifier family is designed to offer a wider selection than any previously developed operational amplifier family. Each of these JFET-input operational amplifiers incorporates well-matched, high-voltage JFET and bipolar transistors in a monolithic integrated circuit. The devices feature high slew rates, low input bias and offset currents, and low offset voltage temperature coefficient.



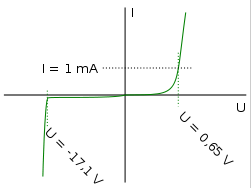




**b. Diode zener**

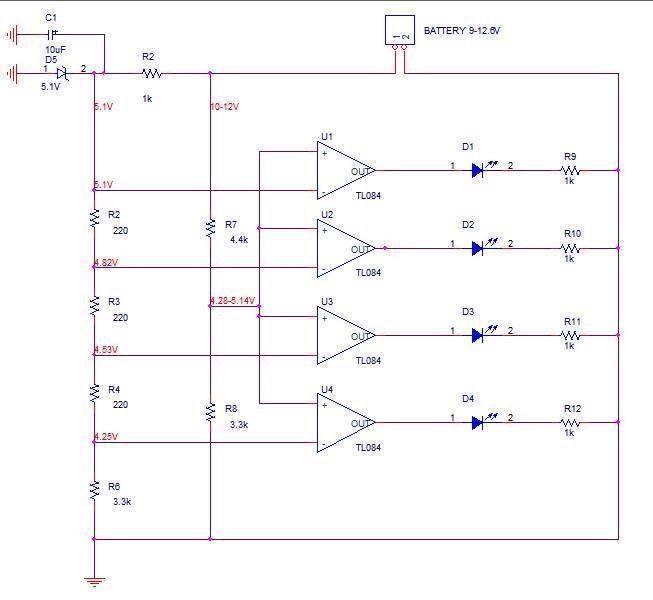
Zener diode, also known as voltage regulator diode, is a semiconductor diode work in reverse polarity mode on the breakdown voltage (breakdown). This voltage is also called Zener voltage avalanche or cascade (avalanche). At that voltage value little changed. It was built so that the reverse polarity, the Zener diodes will pin a fixed voltage level nearly equal to the value indicated on the diode, do Stabilizers of circuit.

When biased diodes Zener diodes operate like normal. When the polarity invert, at first only a small electric current through the diode truth. But if the voltage is increased to a value inversely adaptation: Vnguoc = Vz (Vz: Zener voltage), the current through diodes increase, but the voltage between the two ends of the diodes hardly change, so-called Zener effect.

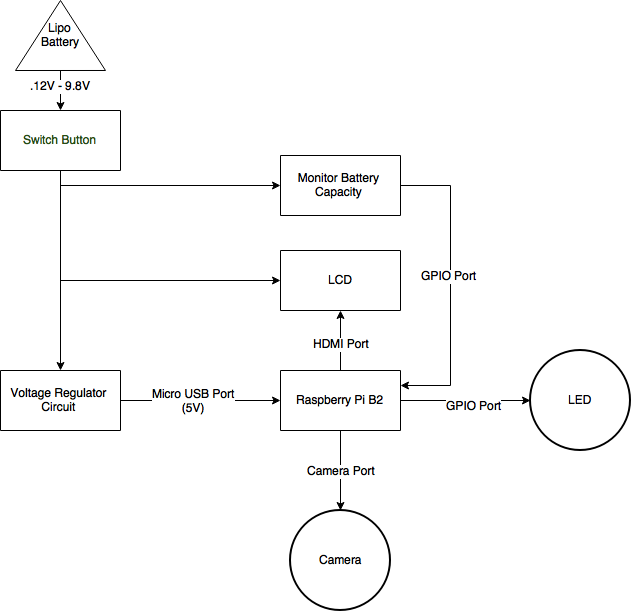


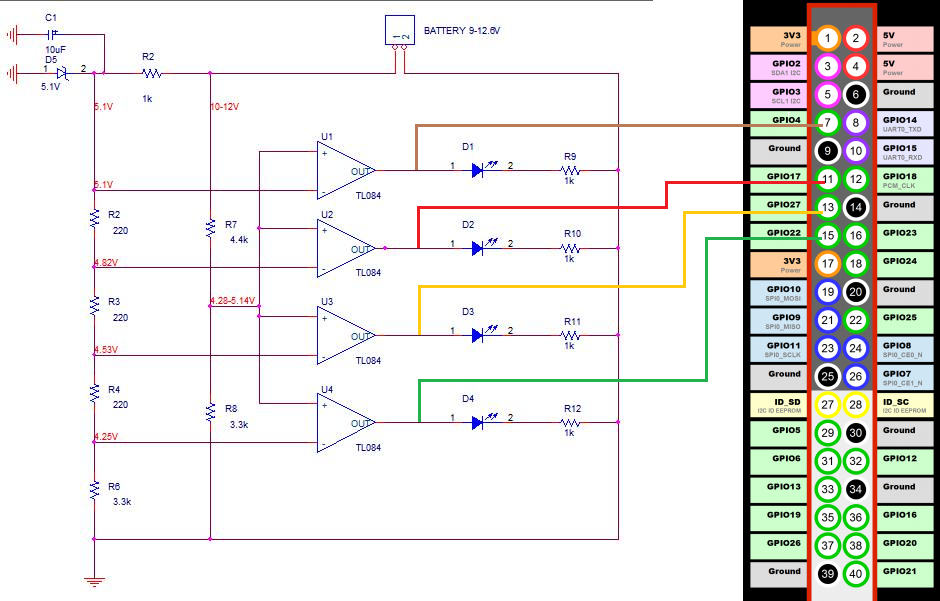


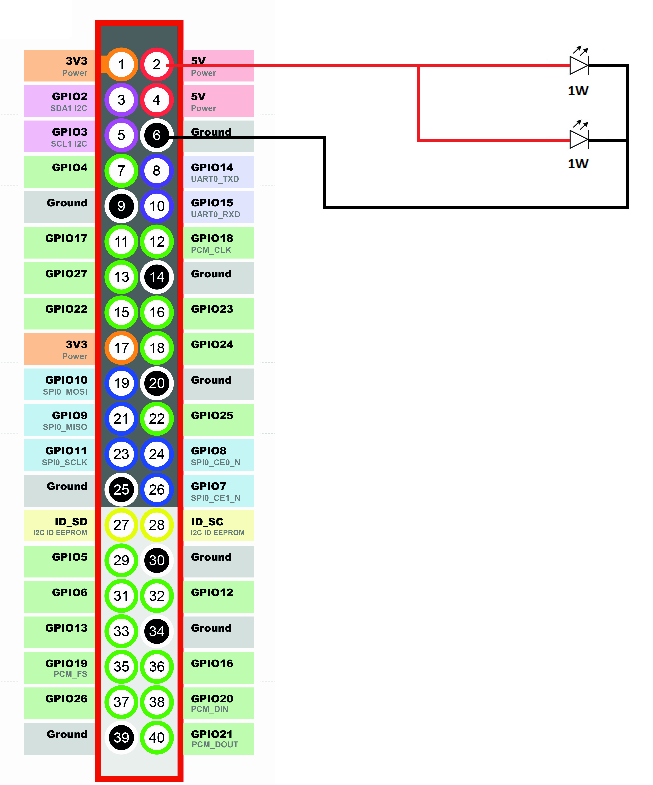
**2. Principles of circuit**



#### Connecting Components In System







## Algorithms