**D) SOLUTION PROCEDURE**

**D.3) SUBSYSTEMS**

**D.3.2) Image Acquisition Subsystem**

Image acquisition subsystem consists of a camera and a Raspberry Pi. The Python will be utilized for the software.

***Capturing An Image:***

An image will be captured via Raspberry Pi camera module. This camera has 2592x1944 pixel resolution. The command to take the picture will be given by user via a pushbutton. After command of user, 2 seconds will be spent before taking the photo to give the camera’s sensor enough time to sense the light levels. [1]

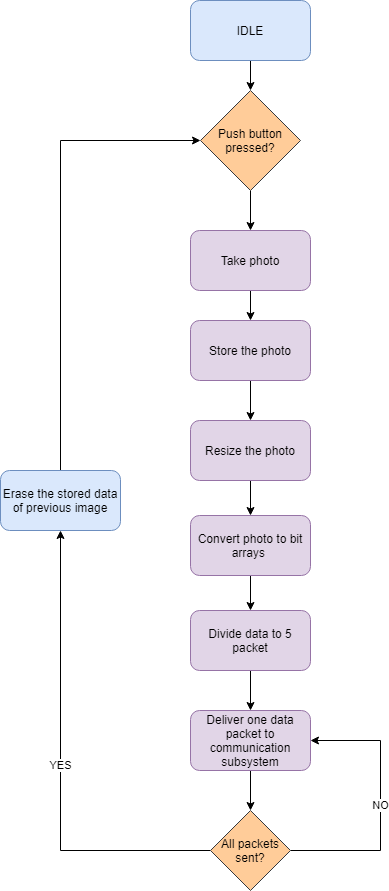
***Storing, Resizing and Division of the Captured Image:***

After image is captured, it will be stored and processed using the Raspberry Pi placed in the first terminal.

The maximum resolution of camera module is 2592x1944(5 MP) and the minimum resolution is 64x64. The image will be obtained in the maximum resolution and it will be resized using the tools of Python Image Library (PIL). By doing optimizations on the tradeoff between data amount (resolution) and photograph quality (compression), the image file will be saved in JPEG format.

To transfer an image, it is required to convert an image to a bit arrays in such a way that these bit arrays represents the image. The binary data will be split into 5 packets of data chunk by simply splitting the array of data into five blocks of arrays. These packets will be delivered to communication subsystem to be sent via visible light communication. Throughout these steps, tools such as “numpy”, “math”, “os” and “image-slicer” are utilized.

The algorithm followed by the image acquisition subsystem is given in Figure 1 as a flowchart diagram.



*Figure 1: Flowchart of the Image Acquisition Subsystem*

**D.3.3 Image Reconstruction Subsystem**

Image reconstruction subsystem consists of a display and a Raspberry Pi. The Python will be utilized for the software.

***Collecting Data:***

At the second terminal, communication subsystem will deliver the data that is received by the receiver to the image reconstruction subsystem and data is stored in the Raspberry Pi.

***Merging the Data:***

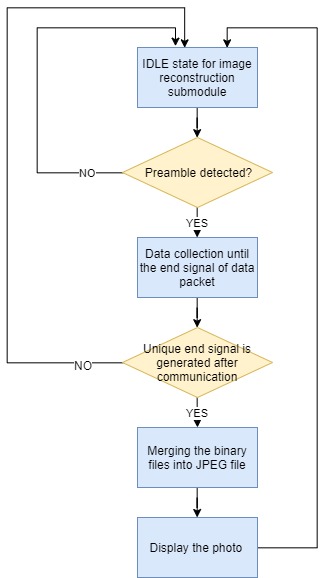
Upon receiving unique end signal, the process of collecting data is finished.

The bin files are merged on the general purpose computer as a JPEG file via PIL of Python. As image file is merged the proper signal is generated to initiate the display process.

***Displaying the Reconstructed Image:***

A Raspberry Pi compatible LCD screen, having a resolution of 128x128, is utilized by using SPI (Serial Peripheral Interface) via its drivers. The screen operates with 3.3 V. [2]

The algorithm followed by the image acquisition subsystem is given in Figure 2 as a flowchart diagram.



*Figure 2: Flowchart of the Image Reconstruction Subsystem*