



*VISUAL IMPAIRED **HELPER***

A QUICK RECAP

Problem

Many visually impaired people are using older methods for guidance and do not leverage the technology available today.

Solution

My proposed solution is to use a system that a user wears i.e a headband that guides the visually impaired person

PROJECT DESIGN & DEVELOPMENT

High Level Design Computer Vision
Low Level Design Computer Vision
Prototype (Computer Vision)



High Level Design Engineering
Low Level Design Engineering
Prototype (Engineering)





HIGH LEVEL DESIGN (VISION)

- 1** *INPUT*
- 2** *CAPTURE IMAGE DATA*
- 3** *BLOB DETECTION*
- 4** *PATH RECOGNITION*
- 5** *OUTPUT*

HIGH LEVEL DESIGN (VISION)

1

INPUT

KINECT VIDEO FEED

2

CAPTURE IMAGE DATA

CAPTURE DEPTH AND VIDEO
FRAMES

3

BLOB DETECTION

DETECT BLOBS USING OPENCV

4

PATH RECOGNITION

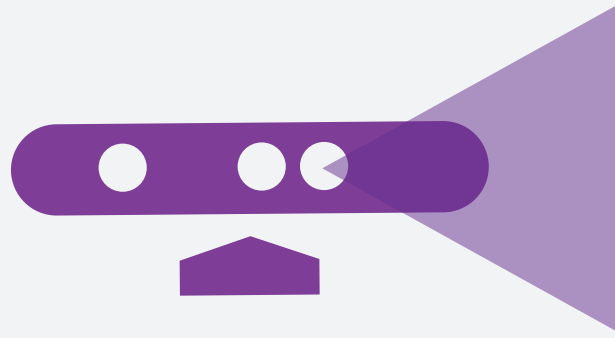
CLASSIFY WHICH SEGMENTS OF
IMAGE ARE OPEN

5

OUTPUT

SEND ENCODED STRING TO
ARDUINO MICROCONTROLLER

HIGH LEVEL DESIGN (VISION)



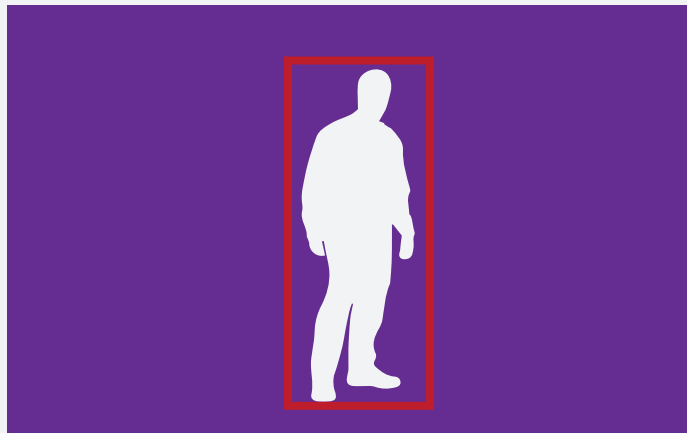
1

INPUT



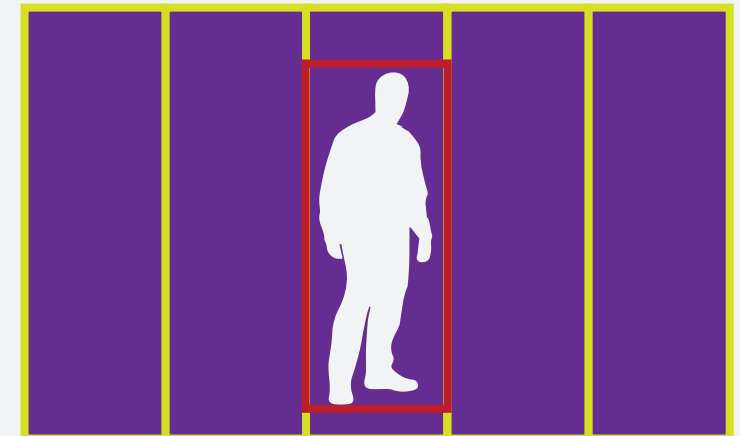
2

CAPTURE IMAGE DATA



3

BLOB DETECTION



4

PATH RECOGNITION

LOW LEVEL DESIGN (VISION)

1

INPUT

START KINECT SENSOR

ENABLE COLOR & DEPTH STREAMS

3

BLOB DETECTION

CONVERT KINECT DATA STREAM
INTO AN OPENCV IMAGE

CONVERT OPENCV IMAGE TO
GRAYSCALE

DETECT OUTERMOST CONTOURS

DRAW MINIMUM BOUNDING BOX
AROUND THE CONTOUR

DETECT CLOSEST PART OF BLOB
AND OBTAIN THE DISTANCE
USING KINECT API

2

CAPTURE IMAGE DATA

CAPTURE FRAMES USING AN EVENT
HANDLER (THIS MODEL KEEPS
GETTING FRAMES UNTIL PROGRAM
HALTS)

4

PATH RECOGNITION

CHECK WHICH BOUNDING BOXES
COLLIDE WITH SEGMENTS

BUILD ENCODED STRING BASED ON
COLLISIONS

5

OUTPUT

SENT BUILT STRING TO ARDUINO
MICROCONTROLLER USING SERIAL
CONNECTION

PROTOTYPE

1 DEMO

HIGH LEVEL DESIGN (ENGINEERING)

1 *INPUT*

2 *PATH RECOGNITION*

3 *OUTPUT*



HIGH LEVEL DESIGN (ENGINEERING)

1

INPUT

LISTEN ON SERIAL PORT

*ACCEPT STRING DATA WHEN
AVAILABLE*

2

PATH RECOGNITION

*DECIDE WHICH VIBRATION
MOTORS TO VIBRATE*

3

OUTPUT

VIBRATE MOTORS

LOW LEVEL DESIGN (ENGINEERING)

1

INPUT

LISTEN ON SERIAL PORT USING SERIAL-BEGIN(9600) AND CHECK FOR SERIAL INPUT IN MAINLOOP USING SERIAL-AVAILABLE()

2

PATH RECOGNITION

CUSTOM METHOD TO SPLIT INPUT STRING, SPLITANDSET(CHAR [] INPUT)

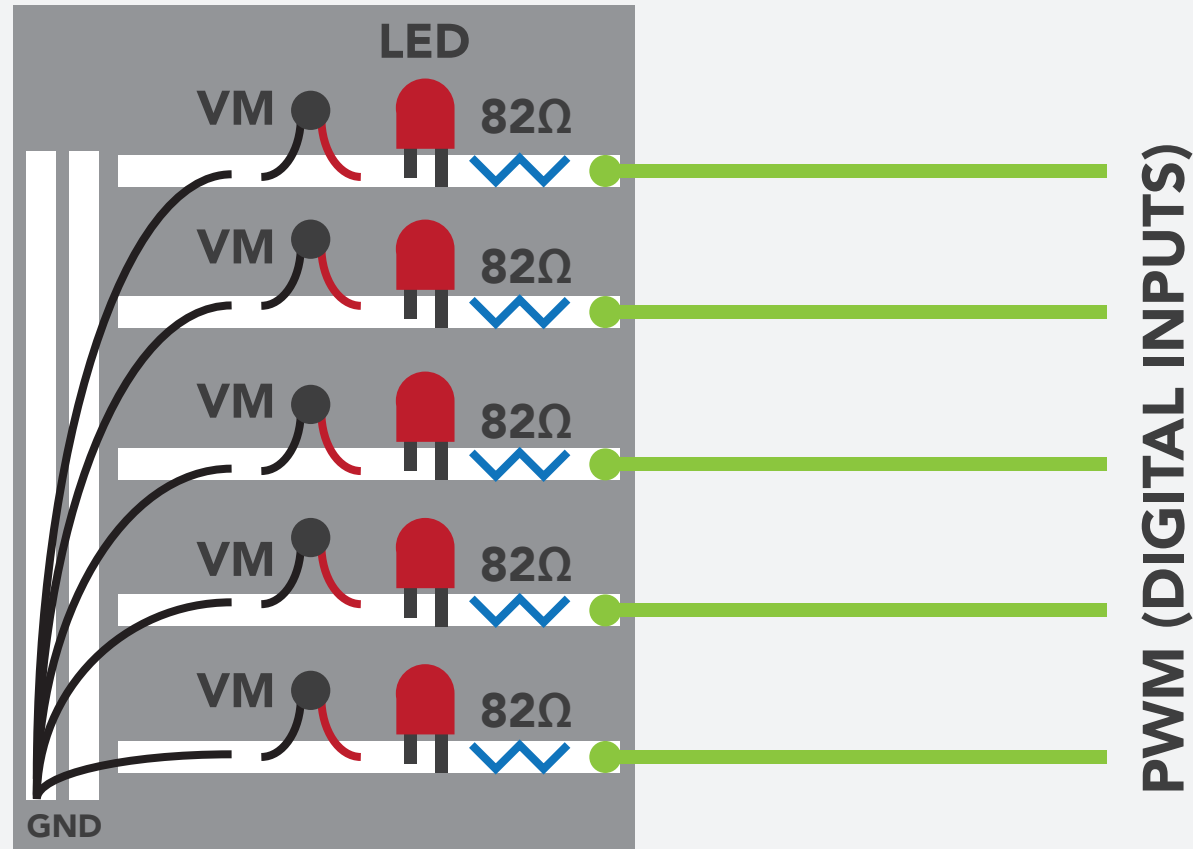
3

OUTPUT

VIBRATE MOTORS USING DIGITALWRITE(OBJECT, HIGH)

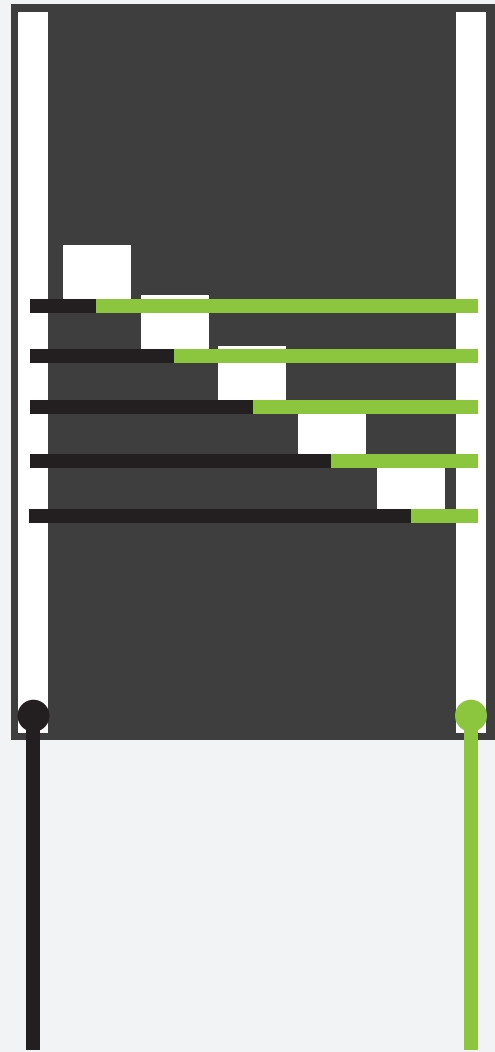
CIRCUIT DESIGN (ENGINEERING)

BREADBOARD PROTOTYPING



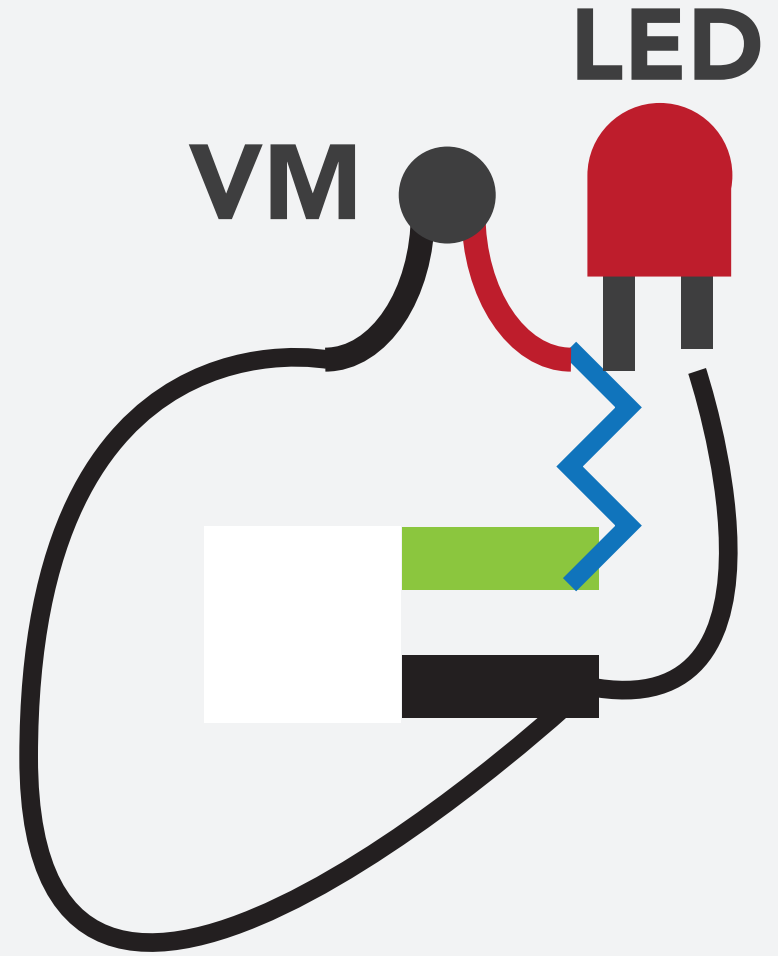
CIRCUIT DESIGN (ENGINEERING)

FINAL DESIGN (REAL CIRCUITS)



DISTRIBUTION BOARD

1X



VIBRATION CIRCUIT

5X

PROTOTYPE



DEMO



REFERENCES

**Gary Bradski & Adrian Kaehler.
(2008). Learning OpenCV. O'Reilly
Media Inc.**

Google+ Kinect Community

**Abhijit Jana. (2012) Kinect for
Windows SDK Programming Guide.
Packt Publishing.**