A black background with text and a person in a hat

Description automatically generated

Mechatronics System Integration

(MCTA3203)

Semester 2, 2023/2024

Report: experiment; week 9

Lecturers:

Dr/ WAHJU SEDIONO

Dr/ ZULKIFLI BIN ZAINAL ABIDIN

Dr/ ALI SOPHIAN

Section 1 Group 3

|  |  |
| --- | --- |
| **Names** | **Matric** |
| Abdallah Mahamat Abacar | 2117777 |
| Almasri Suhail Jihad | 2128771 |
| Ibrahim Bin Nasrum | 2116467 |
| Khan Toqi Tahmid | 2025209 |

**Color Detection and Analysis Using Arduino Uno and Pixy Camera**

**Introduction**

**The Pixy Camera, developed by Charmed Labs in collaboration with Carnegie Mellon University, is a robust vision sensor designed for easy integration with microcontrollers like the Arduino. It enables projects that require high-speed object detection and tracking based on color signatures, making it a popular choice for robotics, automation, and interactive applications.**

**Objective**

**The objective of this experiment is to build a system using an Arduino Uno and Pixy camera to detect three different colored objects. This involves setting up the hardware, calibrating the Pixy camera, installing the necessary software libraries, writing the Arduino code to interface with the Pixy camera, and testing the system for accurate color detection.**

**Hardware Setup**

*Connect Pixy Camera to Arduino Uno*:

Connect the Pixy camera to the Arduino Uno.

**Connect the Pixy camera to the Arduino Uno. Depending on the chosen communication interface, connect the pins as follows:**

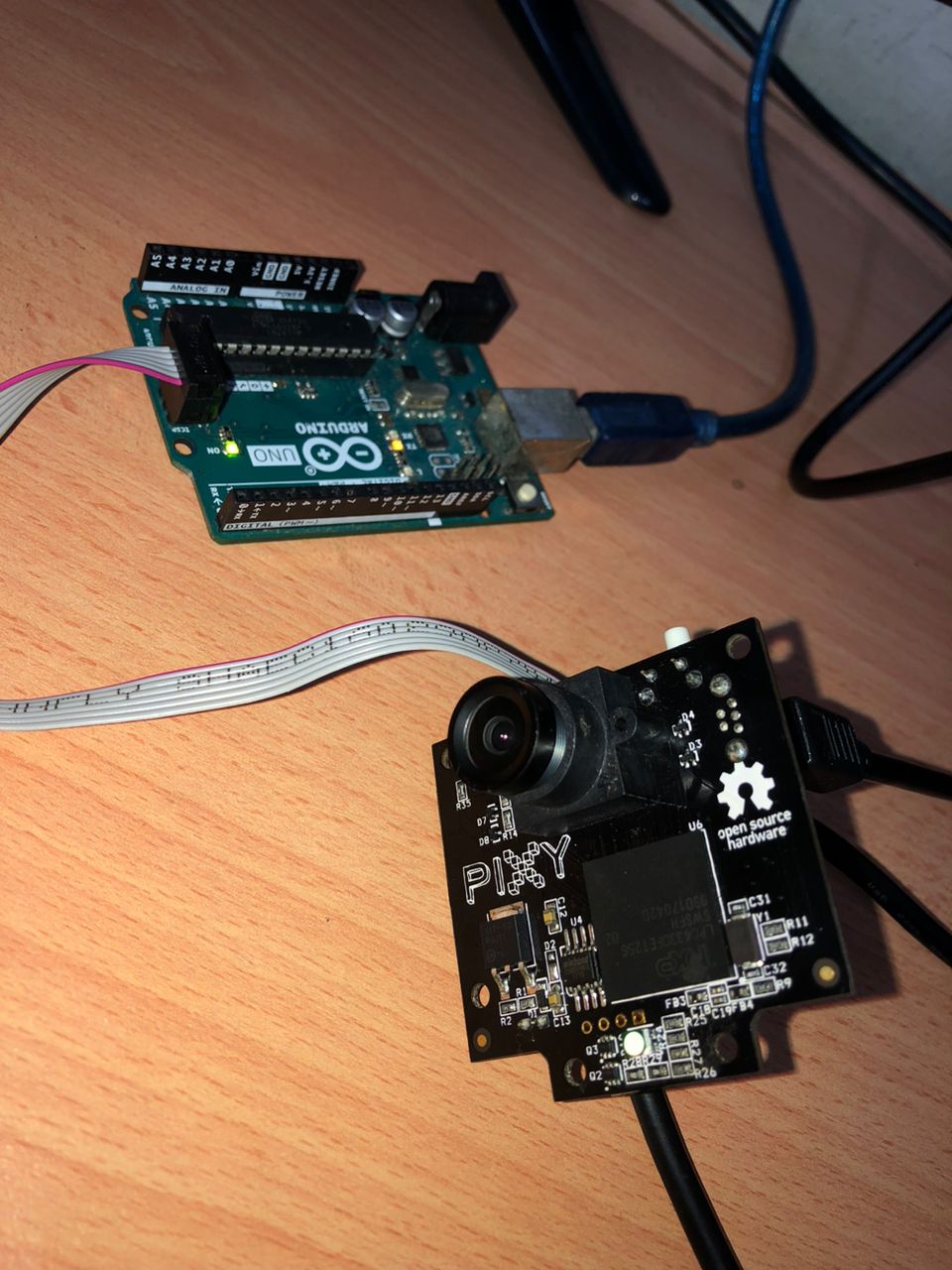
**I2C Interface:**

**GND (Ground)**

**+5V (Power)**

**SDA (Data Line)**

**SCL (Clock Line)**

****

**Arduino Code** *to Detect Objects*:

#include <SPI.h>

#include <Pixy.h>

// This is the main Pixy object

Pixy pixy;

void setup()

{

  Serial.begin(9600);

  Serial.print("Starting...\n");

  pixy.init();

}

void loop()

{

  static int i = 0;

  int j;

  uint16\_t blocks;

  char buf[32];

  // grab blocks!

  blocks = pixy.getBlocks();

  // If there are detect blocks, print them!

  if (blocks)

  {

    i++;

    // do this (print) every 50 frames because printing every

    // frame would bog down the Arduino

    if (i%50==0)

    {

      sprintf(buf, "Detected %d:\n", blocks);

      Serial.print(buf);

      for (j=0; j<blocks; j++)

      {

        sprintf(buf, "  block %d: ", j);

        Serial.print(buf);

        pixy.blocks[j].print();

      }

    }

  }

}

This code reads the blocks detected by the Pixy camera, prints their signatures, X, and Y coordinates, and provides a basic structure for identifying the three different colored objects.

**Test and Debug**

Serial Monitor in the Arduino IDE

A screenshot of a computer

Description automatically generated

Result and Discussion

After conducting the experiment using the Arduino Uno and Pixy camera for color detection, it was evident that the Pixy camera effectively recognized red, green, and yellow objects based on calibrated color signatures, operating at up to 50 frames per second for real-time processing. Despite occasional challenges with environmental factors affecting detection precision, such as lighting variations, the system demonstrated robust performance suitable for applications in robotics, automation (like object sorting), and interactive systems (such as educational tools). Moving forward, implementing dynamic calibration strategies and sensor fusion approaches would enhance adaptability and overall accuracy, ensuring reliable performance across diverse operational environments.

Here is the links for the video and image: https://github.com/Moayed13795/MCTA3203/tree/main/Week9/week9

Conclusion

With these steps, you should have a basic system that detects three different colored objects using the Arduino Uno and Pixy camera. Adjust the code and configurations as needed for your specific application. This experiment demonstrates the integration of image/video input interfacing with a microcontroller and computer-based system, highlighting the practical application of mechatronics principles in color detection and analysis.