

A COMPUTER VISION BASED FRUIT SORTING AND GRADING SYSTEM

Project Report Submitted
in partial fulfillment of the requirements for the degree
Of
Bachelor of Technology
In
Department of Electronic and Communication Engineering

Syed Hussain Razavi (IUST0119012664)

Dr. Liyaqat Nazir

Rizwan Yousuf Dar (IUST0119012591)

Assistant Professor

Syed Mohsin Bukhari (IUST0119012765)

Supervisor

Burhan Showkat (IUST0119012623)

Dr. Afshan Amin

Assistant Professor

Co-Supervisor



**DEPARTMENT OF ELECTRONIC AND COMMUNICATION
ISLAMIC UNIVERSITY OF SCIENCE & TECHNOLOGY
AWANTIPORA, J&K, INDIA – 192122
SEPTEMBER, 2023**

DECLARATION

This project report presents the work that has been carried out in **Department of Electronics and Communication Engineering, Islamic University of Science and Technology, Awantipora** under the guidance of **Dr. Liyaqat Nazir(Supervisor)** and **Dr. Afshan Amin(Co-Supervisor)**. This project report is a product of our own work and has been completed independently. Neither the whole nor any part of this project report has been submitted for any degree or diploma except that than for B.Tech degree at **Department of Electronics and Communication Engineering, Islamic University of Science and Technology, Awantipora, J&K.**

We further declare that the figures/data utilized for the completion of this project report from sources such as other works in the literature, the internet, and estimates provided by corporations in the form of white paper or on their websites, have been mentioned in the form of reference. By referencing, the data/figures have been borrowed.

1) Syed Hussain Razavi (ECE-19-38)

2) Rizwan Yousuf Dar (ECE-19-27)

3) Syed Mohsin Bukhari (ECE-19-47)

4) Burhan Showkat (ECE-19-33)

CERTIFICATE

This is to certify that the Project report entitled "**A COMPUTER VISION BASED FRUIT SORTING AND GRADING SYSTEM**" submitted by: **Syed Hussain Razavi, Rizwan Yousuf Dar, Syed Mohsin Bukhari, Burhan Showkat** for the award of the degree of **Bachelor of Technology** in the Department of Electronics and Communication Engineering, Islamic University of Science and Technology, Awantipora, Jammu and Kashmir, India, is a record of bonafide work carried out by them under our guidance and supervision. They have fulfilled the requirements for the submission of the report, which to our knowledge has reached the requisite standard. The results contained in this project have not been submitted in part or in full to any university or institute for the award of any degree or diploma.

SEPTEMBER, 2023

Signature

**Dr. Liyaqat Nazir
Assistant Professor**

Signature

**Dr. Afshan Amin Khan
Assistant Professor**

Signature

**Head of Department
Dr. Faroze Ahmad
Associate Professor**

ACKNOWLEDGEMENT

In the name of Allah, the Most Gracious, the Most Merciful.

All praise and gratitude are due to Allah, the Lord of the worlds, the Most Compassionate and Most Merciful. We begin by acknowledging His blessings throughout this journey. It is only by His grace that we have been able to complete our final year project successfully. Being in IUST we had the privilege of being able to work for a research project awarded to Dr. Liyaqat Nazir, funded by Jammu & Kashmir Science and Technology and Innovation Council (JKST&IC). We are extremely indebted to JKST&IC for their financial support without which the completion of this project would not have been possible. Although the experience has not been easy, We are incredibly grateful to the Principal Investigators(PI) Dr. Burhan Khursheed, and Co-PI Mr. Rouf-ul-Alam Bhat for providing us with an opportunity to work on the project with the continued support of the funding agency. We would like to express our deepest appreciation to our project supervisor, Dr. Liyaqat Nazir, whose unwavering support and dedication have been invaluable. Dr. Liyaqat, you have been more than just a supervisor; you have been a mentor, a guide, and a teammate. Your expertise, encouragement, and willingness to work with us like a team have been instrumental in shaping this project and our understanding of the subject matter.

We also extend our heartfelt gratitude to our co-supervisor, Dr. Afshan Amin Khan, for his invaluable suggestions and tireless efforts in helping us troubleshoot various bugs and challenges encountered during the development phase of our project. Your insights and expertise have greatly enriched our work.

We would like to thank the Head of the Department, Dr. Faroze Ahmad, for providing us with the necessary resources and facilities to undertake this project. Your support and encouragement have been vital in our academic journey.

Last but not least, we express our gratitude to our families for their unwavering support, patience, and understanding during the challenging times of this project. Your belief in us has been our driving force.

We are deeply grateful to everyone who has played a role, big or small, in the successful completion of our final-year project. This project would not have been possible without the guidance, support, and blessings from Allah, and the collaboration and assistance of our mentors, colleagues, and loved ones.

Thank you all for being a part of this significant journey.

ABSTRACT

This report presents the development and deployment of an innovative fruit sorting and grading system, leveraging the power of computer vision and Convolutional Neural Networks. The project encompasses a comprehensive journey, from data acquisition and model training to hardware design and deployment on single-board computers and FPGA devices. The primary objective of this project is to create an efficient and accurate solution for automating the fruit sorting and grading process, thereby enhancing productivity and reducing human labor. The project's foundation is built upon the TensorFlow framework, which enabled the training of a robust Convolutional neural network model capable of accurately detecting and classifying different types of fruits. Several datasets were meticulously curated and employed for model training, leading to a highly reliable and versatile fruit recognition system. In the hardware domain, we designed a custom structure using T-slot aluminum extrusion and 18-gauge mild steel sheets to house and protect the essential electrical components. The electrical circuitry was meticulously planned and implemented using KiCad, facilitating seamless connectivity between microcontrollers, sensors, and the Jetson Nano. The design ensures stability, durability, and ease of maintenance for long-term operation. Extensive testing and optimization efforts were carried out on various single-board computers and an FPGA device, culminating in the deployment of the system on the Jetson Nano. Real-world trials and evaluations demonstrated the system's efficiency and accuracy in sorting and grading a wide variety of fruits. The results of this project signify a significant advancement in automation within the agricultural industry. The automated fruit sorting and grading system not only reduces labor costs but also enhances the precision and consistency of the grading process, resulting in improved product quality and increased overall productivity. Additionally, the project showcases the seamless integration of state-of-the-art technologies and hardware design, setting a precedent for future endeavors in the field of automation and computer vision. This report provides a comprehensive overview of the project's inception, development, and successful deployment, shedding light on the methodology, challenges faced, and future prospects.

TABLE OF CONTENTS

Title	Page No.
DECLARATION	i
CERTIFICATE	ii
ACKNOWLEDGEMENT	iii
ABSTRACT	iv
TABLE OF CONTENTS	v
LIST OF TABLES	viii
LIST OF FIGURES	ix
1 Introduction	1
2 LITERATURE SURVEY	2
3 BACKGROUND	3
4 METHODODOLOGY	5
4.1 Framework Selection	6
4.1.1 Community Support	6
4.1.2 Versatility and Flexibility	6
4.1.3 TensorBoard Visualization	6
4.1.4 Deployment Options	6
4.1.5 Transfer Learning and Pre-trained Models	7
4.1.6 Ecosystem and Integration	7
4.2 Neural Network Selection	7
4.2.1 Hierarchical Feature Extraction	8
4.2.2 Spatial Invariance	8
4.2.3 Parameter Sharing and Efficiency	8

4.2.4	Handling Varied Image Sizes	9
4.2.5	Transfer Learning	9
4.2.6	Availability of Pre-trained Models	9
4.2.7	State-of-the-Art Performance	9
4.3	Dataset	9
4.4	Training Process	10
4.4.1	Dataset Preprocessing	10
4.4.2	Model Selection	11
4.4.3	Transfer Learning	12
4.4.4	Configuration and Training	13
4.4.5	Model Evaluation	13
4.4.6	Fine-tuning and Iterative Improvement	14
4.5	Optimization Techniques	15
4.5.1	Model Export and Deployment	17
4.6	Hardware Implementation	18
4.6.1	Hardware Interconnections	18
4.7	Raspberry pi 4	26
4.7.1	Nvidia Jetson Nano	27
4.8	PYNQ Z2 FPGA Board	28
4.9	Embedded Hardware	29
4.9.1	Arduino Nano	29
4.10	Stepper Motor	29
5	ARCHITECTURE	31
5.1	SYSTEM DESIGN	31
5.1.1	Cleansing Unit	32
5.1.2	Image Capturing Section	32
5.1.3	Conveyor level 1	33
5.1.4	Conveyor level 2	33
5.1.5	Human Machine Interface	33

5.2	Convolutional Neural Network Architecture	33
5.2.1	Layers of Convolutional Neural Network	34
6	PHYSICAL STRUCTURE	35
6.1	Feeder	36
6.2	Cleaner Section	36
6.3	Conveyor System	37
6.3.1	Conveyor level 1	38
6.3.2	Sorting Flap	39
6.3.3	Conveyor level 2	40
6.3.4	Curved Ramp	41
6.4	Grading Gates	41
6.4.1	Gate-1	42
6.4.2	Gate-2	43
6.5	Collector Tray	43
6.6	Gas Spring Monitor Arm	44
6.7	Aluminum Support	45
7	RESULTS	46
8	CONCLUSION AND FUTURE WORK	51
APPENDIX	53
1	JETSON NANO DATA-SHEET	53
2	ARDUINO MEGA DATA-SHEET	95
3	TB6600 DATA-SHEET	114
4	LM7805 IC DATA-SHEET	123
REFERENCES	152

LIST OF TABLES

1	Comparison of datasets based on specified criteria.	10
2	Accuracy percentage of different datasets.	47
3	Avg delay obtained on each device.	47
4	Model Optimization Results	49

LIST OF FIGURES

1	Block Diagram of McCulloch/Pitts neuron	4
2	Block Diagram of Perceptron	4
3	Block Diagram of Multilayer Architecture of Artificial Neural Network	5
4	Main window of TensorBoard	7
5	Basic Architecture of Convolutional Neural Network	8
6	Labeling being used to annotate custom dataset.	11
7	Detection Box precision on TensorBoard while training.	15
8	Model accuracy with no. of epochs on TensorBoard.	16
9	Block Diagram of Hardware interconnected with different peripherals	18
10	Figure showing Different interfaces of Nextion	19
11	Schematic of Arduino MEGA interconnected parallelly with different sections	20
12	PCB Routing diagram of Arduino MEGA interconnected parallelly with different sections	21
13	Drawing of Robotic Arm Designed in CAD	22
14	KiCAD diagram visualizing the Sorting and Cleaner Section	22
15	PCB Routing diagram visualizing the Sorting and Cleaner Section .	23
16	KiCAD diagram visualizing the Conveyor level 1 and level 2	24
17	PCB Routing diagram visualizing the Conveyor level 1 and level 2	24
18	PCB routing of Grading Section	25
19	Schematic visualizing the Grading Section	25
20	Pinout diagram of Raspberry Pi v4 with 8GB RAM.	27

21	Pinout diagram of Nvidia Jetson Nano.	28
22	Pinout diagram of Pynq FPGA Z2.	29
23	Pinout diagram Arduino Nano.	30
24	NEMA23 Stepper Motor.	30
25	Picture Depicting the Internal Construction of a Stepper Motor.	31
26	Block diagram of computer-vision-based fruit grading sorting system	32
27	Layers of Convolutional Neural Network	34
28	Block diagram of Convolutional Neural Network	34
29	Figure showing top view CAD drawing of Conveyor system	35
30	Figure showing the CAD drawing of Conveyor system	36
31	Figure showing Top view of the Feeder	36
32	Figure showing Top view of the Cleaner section with pulleys	37
33	Figure showing the shaft connected with stepper motor using Timing belt	37
34	Figure showing Top view of the Conveyor system	38
35	Figure showing Top view of the Conveyor level 1	38
36	Figure showing Front view of the Gear-Chain mechanism	39
37	Figure showing Side view of the Gear-Chain mechanism	39
38	Figure showing top view of sorting flap using servo motor	40
39	Figure showing top view of Conveyor 2 with the peripherals attached to it	40
40	Figure showing a lateral view of the gear-chain mechanism of conveyor 2	41
41	Figure showing a side view of curved ramp	41
42	Figure showing a the front view of grading gates	42
43	Figure showing a front view of Gate-1	42
44	Figure showing a front view of Gate-2	43
45	Figure showing a top view of collector tray at its extreme position .	43
46	Figure showing how actuator is coupled with collector tray	44

47	Figure showing how linear actuator is connected with the Aluminum structure.	44
48	Figure showing a side view of the Gas Spring Monitor Arm	45
49	Figure showing the front view of the Gas Spring Monitor Arm	45
50	Figure showing the structure supporting the overall system	46
51	Figure showing the lateral view of the wheel attached with aluminum structure	46
52	Bar graph of accuracy achieved.	47
53	Bar graph of the speed of devices.	48
54	Top view of grading & sorting structure.	48
55	Right Side view of grading & sorting structure.	49
56	Front view of grading & sorting structure.	49
57	Back view of grading & sorting structure.	50
58	Left side view of grading & sorting structure.	50
59	Output window with detection for Raspberry Pi.	50
60	Output window with detection for JETSON Nano.	51
61	Output window with detection for PYNQ Z2.	51

1 Introduction

Agriculture, as the foundation of human civilization, faces the need for technological enhancement. Our project introduces a cutting-edge computer vision-based fruit grading and sorting system, contributing to this transformation. In recent years, computer vision and machine learning have become transformative forces in agriculture. Previous studies focused on disease detection, food quality improvement, and nutrient deficiency assessment. However, our project expands the application of these technologies. Automation in agriculture through computer vision plays a crucial role in disease detection and fruit classification based on texture, color, and shape. Currently, agricultural product waste amounts to 40 percent[1], partially due to inadequate maintenance and fruit identification challenges. Manual visual inspections are time-consuming, labor-intensive, and inconsistent. This approach disrupts fruit grading and impacts productivity. The use of novel techniques to estimate agricultural goods' quality has been sped up by the investigation and development of some fundamental ideas and methodologies of machine vision for pear quality detection and sorting operations [2]. For agricultural grading purposes, numerous color vision systems have been developed. The direct color mapping technique used to assess the quality of tomatoes and dates [3] and the automated examination of golden delicious apples using color computer vision [4] are examples of these uses. Automating inspection procedures like fruit quality assessment is now possible because of advancements in imaging technology and pattern recognition algorithms. An electromechanical fruit handler that can place a fruit on a conveyor belt and transport it through a computer vision system to the sorting bins is a typical component of a computer vision system that can visually inspect the fruit, evaluate its quality, and sort it. An image processor receives the image of the fruit underneath from the computer vision system. The image is presented to a pattern recognizer by the processor once it has been processed. The recognizer evaluates the underlying fruit's quality, categorizes it into predetermined quality groups, and instructs the sorter to place the fruit in the proper bin [5]. Numerous academics and scholars have committed their efforts to finding effective solutions for fruit grading and sorting utilizing computer vision and machine learning techniques. Image segmentation, feature extraction, pattern recognition, and deep learning are among the approaches used in these solutions. Various systems may accurately classify fruits based on their quality and assign them to suitable grade categories by combining various strategies. Our project addresses these challenges. Automated grading and sorting are imperative to meet the demand for high-quality, affordable food. Transitioning from manual sorting to machine vision offers accuracy, precision, and speed. These systems analyze criteria like size, shape, color, and imperfections,

distinguishing between fruit varieties and grades. Computer vision and machine learning have automated fruit quality assessment. Advancements in imaging and pattern recognition enable precise grading and sorting. Our project aims to develop an economical, portable fruit grading and sorting system with a user-friendly interface, benefiting farmers. In the following chapters, we delve into methodologies, technical aspects, and outcomes, advancing agriculture through modern technology.

2 LITERATURE SURVEY

Computer vision has emerged as a transformative technology with far-reaching implications across various industries. Its importance lies in its ability to extract valuable information from visual data, enabling automation, efficiency improvements, and enhanced decision-making processes. In recent years, computer vision has made substantial contributions in fields such as healthcare, manufacturing, robotics, and agriculture. Several researchers have contributed significantly to the development of computer vision-based fruit sorting and grading systems.

In [6] Authors photographed fruits, and their RGB color model was converted to the HSI color model. A constructed Backpropagation network's input was a simple histogram of hue H that was computed. The description of the fruits' appraised quality was the Backpropagation network's output. After training, the Backpropagation network used the simplified histogram of H of the colored image of the fruit to determine the quality of the fruit.

The Author in [5] introduces a computer vision-based system for grading date fruits. It employed image processing techniques to extract features such as size, color, and shape for grading. The system is designed to automate the date fruit grading process, reducing the need for manual labor. It aimed to classify dates into different grades based on visual attributes. The paper does not provide detailed results but serves as an early exploration into computer vision-based fruit grading systems, highlighting the potential for automation.

The Author in [7] explores fruit quality evaluation using the HSI color model, a significant departure from traditional RGB-based approaches. It provides insights into how color models can be leveraged for fruit assessment. The paper lays the groundwork for using alternative color models in fruit quality evaluation.

The Author in [8] provides a review that consolidates existing research on color measurements using computer vision in food quality control. It does not present a new methodology but provides an overview of methods used in the field. The paper summarizes various approaches and techniques used for color-based food quality control, involving image analysis and color feature extraction.

The Author in [9] introduces a novel method for sorting and grading mangoes us-

ing a computer vision system. It involves image processing and feature extraction techniques. The system is designed to automate mango sorting and grading, using visual attributes such as size, color, and shape.

The Author in [10] employs computer vision techniques, including image correction and classification using a weighted Relevance Vector Machine (RVM) classifier. The system detects defective apples through image analysis, correcting for lighting variations and employing a weighted RVM classifier for classification.

The Author in [11] provides the details for the development of an automated machine vision-based system for fruit sorting and grading, employing image processing techniques and machine learning algorithms. The system aims to automate the fruit sorting and grading process, using visual features for classification. The paper present results showcasing the system's performance in fruit sorting and grading, highlighting its potential for agricultural automation.

Authors in [12] review the fundamentals and applications of computer vision for food color measurement. An introduction of color space and traditional color measurements is also given. The advantages and disadvantages of computer vision for color measurement are analyzed and its future trends are proposed. This review covers the fundamentals and typical applications of computer vision in food color measurement.

The paper [13] presents a novel automatic defective apple detection method by using a computer vision system combined with automatic lightness correction, number of the defect candidate (including true defect, stem, and calyx) region counting, and weighted Relevance Vector Machine (RVM) classifier.

3 BACKGROUND

Artificial Neural Networks (ANNs) represent a prominent subset of machine learning and artificial intelligence (AI) algorithms that draw inspiration from the structure and functioning of the human brain. The first mathematical model of neuron was given by Warren McCulloch and Walter Pitts in 1943. Fig 1 shows the block diagram of the McCulloch/Pitts neuron. This laid the foundation for understanding how neurons process information. This model, known as the McCulloch-Pitts neuron, inspired the early development of artificial neural networks.

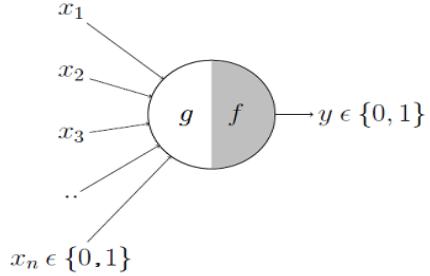


Figure 1: Block Diagram of McCulloch/Pitts neuron

There are ' n ' number of inputs to the McCulloch/Pitts neuron, the value of each input is either '1' or '0' representing the synaptic connection of the neuron.

$$g(x) = \sum_{n=1}^N X_n \quad (1)$$

$$y = f(g(x)) = \begin{cases} 1 & \text{if } y \geq \theta \\ 0 & \text{if } y < \theta \end{cases} \quad (2)$$

All these input signals are summed up and then applied to a transfer function which gives an output of '0' or '1' based on whether the summed signal is above or below a certain threshold. The problem with this model is that it was not capable of learning because of its simplicity and didn't take into account the weights associated with each connection which represents different synaptic strengths of neuron and which is the prime factor in training/learning of neuron. The limitation of McCulloch/Pitts was overcome by introducing the Perceptron. Frank Rosenblatt introduced the perceptron, an early neural network model capable of learning binary classifiers. The perceptron marked a significant milestone in neural network research, as it demonstrated the potential for machines to learn and make decisions based on data. Fig 3 shows the block diagram of Perceptron.

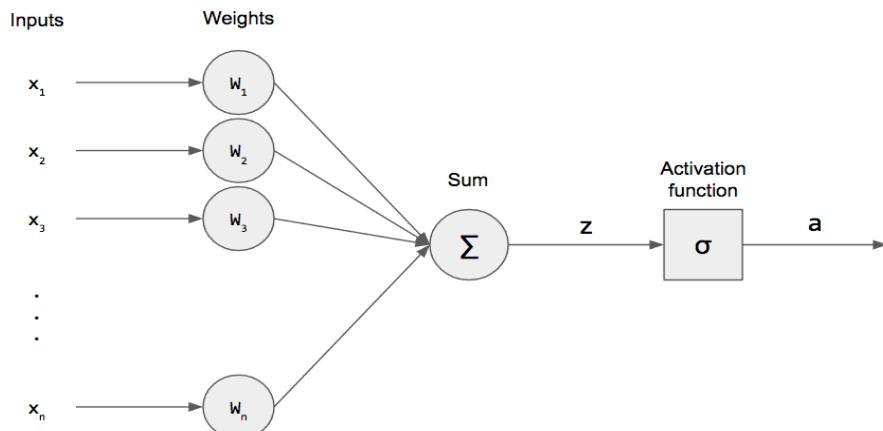


Figure 2: Block Diagram of Perceptron

In this model each input is multiplied by a specific weight that is associated with it, this weight represents excitatory or inhibitory synaptic strength, these signals are summed and applied to a transfer function based on which neuron activates and gives an output of '1' if the sum is above the certain threshold.

Based on a set of input and their corresponding weights if the expected output of the neuron is '1' and we get '0' then we adjust the individual weights of the input until the desired result is achieved this process of adjusting the weight represents the learning of a neuron.

The problem with the single perceptron is that it is incapable of solving linearly inseparable problems, to solve linearly separable problems, a multi-neuron structure is introduced.

A multilayer neural network, also known as a feedforward neural network or a multilayer perceptron (MLP), is a fundamental type of artificial neural network architecture used for various machine learning tasks, including classification, regression, and function approximation. It consists of multiple layers of interconnected neurons (also called nodes or units) organized into three main types of layers: input layer, hidden layers, and output layer.

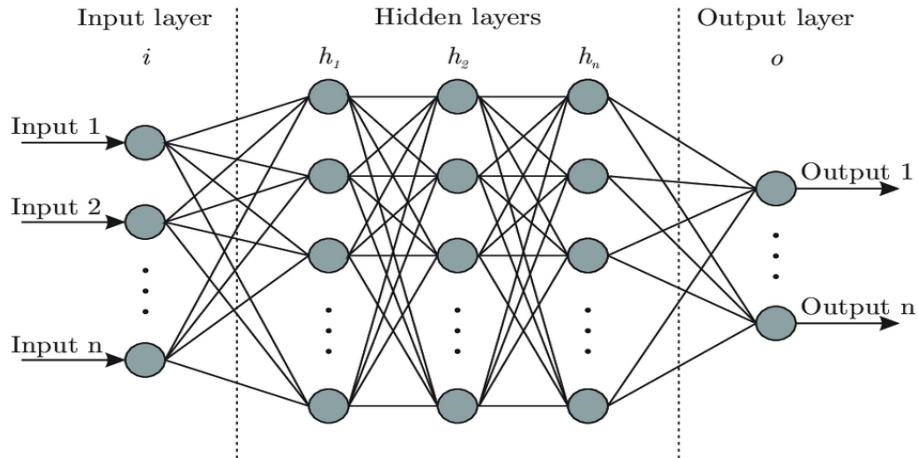


Figure 3: Block Diagram of Multilayer Architecture of Artificial Neural Network

4 METHODOLOGY

This section provides the outline of the methodology implemented by us in the project to create a computer vision-based fruit grading and sorting system. It explains the process of selection for neural network architecture, a framework for training that architecture, the process of collection and organization of data, and how different hardware components were chosen to finally create a complete system.

4.1 Framework Selection

For the selection of a deep learning framework for our project, there were several options available, including TensorFlow, PyTorch, and YOLO (You Only Look Once). While each framework has its own merits and demerits, we choose TensorFlow as the preferred framework for the following reasons:

4.1.1 Community Support

TensorFlow has gained significant popularity and has a large and active community of developers, researchers, and enthusiasts. This widespread adoption ensures that there is extensive documentation, tutorials, and a wealth of resources available. The active community also means that TensorFlow is continuously evolving with regular updates and improvements.

4.1.2 Versatility and Flexibility

TensorFlow is a highly versatile framework that offers a wide range of functionalities beyond deep learning for computer vision tasks. It provides a unified platform for developing, training and deploying machine learning models across different domains and platforms. TensorFlow's flexibility allows for easy integration with other libraries and tools, making it suitable for complex projects that may require additional functionalities beyond computer vision.

4.1.3 TensorBoard Visualization

TensorFlow provides a powerful visualization tool called TensorBoard. It allows for interactive visualization of the training process, model architecture, and various metrics. This visualization capability is valuable when monitoring the performance and progress while Training the data set. TensorBoard provides insights into the model's behavior, making it easier to debug, analyze, and optimize the system's performance. Fig 4 depicts the main window of TensorBoard.

4.1.4 Deployment Options

TensorFlow offers multiple deployment options, including integration with various platforms and frameworks. It supports deployment on CPUs, GPUs, and specialized hardware like Tensor Processing Units (TPUs). TensorFlow also provides options for deploying models on mobile and embedded devices, which is beneficial for our project as we are using hardware-constrained single-board computers.

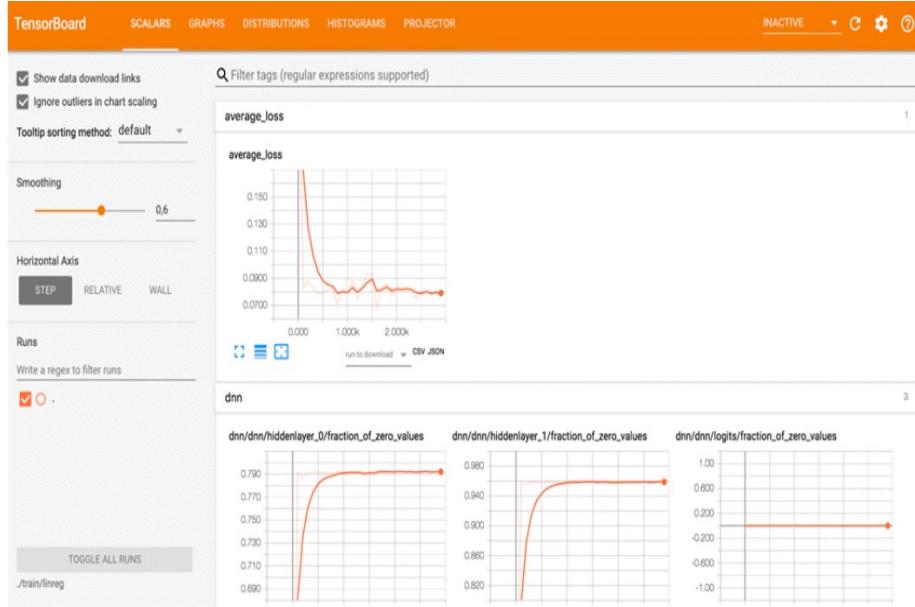


Figure 4: Main window of TensorBoard

4.1.5 Transfer Learning and Pre-trained Models

TensorFlow has a wide range of pre-trained models available through its TensorFlow Hub and TensorFlow Model Zoo. These pre-trained models, especially those trained on large-scale datasets like ImageNet, can be leveraged for transfer learning. Transfer learning allows for efficient training of models on smaller datasets, which is often the case in fruit grading and sorting applications. By utilizing pre-trained models, it becomes easier to achieve good performance with limited labeled data.

4.1.6 Ecosystem and Integration

TensorFlow has a rich ecosystem with numerous libraries, tools, and extensions that enhance its functionality. Libraries like TensorFlow Object Detection API and TensorFlow Lite provide specific functionalities for object detection and deployment on edge devices, respectively. TensorFlow's integration with other libraries and frameworks, such as OpenCV and sci-kit-learn, allows for seamless collaboration and utilization of existing tools.

So, the choice of TensorFlow as the framework for the computer vision-based fruit sorting and grading system project is based on its ease of use, a large community of developers and forums available, and its ability to be used on devices with low hardware specifications.

4.2 Neural Network Selection

For this project, Convolutional Neural Network is the obvious choice for the following reasons:

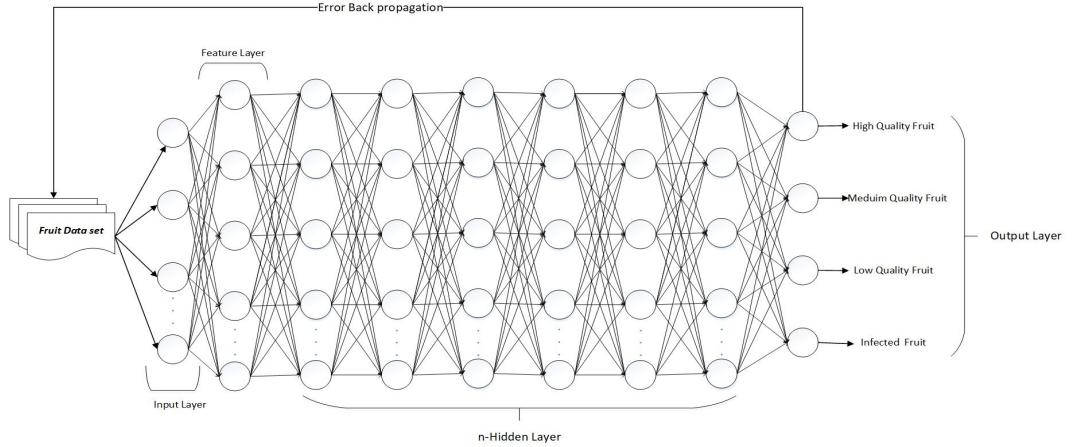


Figure 5: Basic Architecture of Convolutional Neural Network

4.2.1 Hierarchical Feature Extraction

Convolutional Neural Networks are specifically designed for analyzing visual data such as images. They excel at hierarchical feature extraction, capturing both low-level features (e.g., edges, textures) and high-level features (e.g., shapes, patterns) from images. This ability is crucial in fruit grading and sorting, as it allows the network to learn discriminative features that distinguish between different fruit qualities, sizes, colors, and defects.[10]

4.2.2 Spatial Invariance

Convolutional Neural Networks incorporate convolutional layers, which exploit the spatial relationships between pixels in an image Fig 5. By using local receptive fields and weight sharing, Convolutional Neural Networks can capture spatial patterns regardless of their location within the image. This spatial invariance property is beneficial in fruit sorting and grading, as it enables the network to recognize important features regardless of their position, rotation, or scale in the image.

4.2.3 Parameter Sharing and Efficiency

Convolutional Neural Networks employ parameter sharing, where a set of learnable filters is applied across the entire image or feature maps. This sharing of parameters drastically reduces the number of parameters compared to fully connected networks, making Convolutional Neural Networks more memory-efficient and computationally efficient. This efficiency is essential when working with large-scale fruit datasets, enabling faster training and inference times.

4.2.4 Handling Varied Image Sizes

Fruits come in different shapes and sizes, and their images may vary in resolution and aspect ratio. Convolutional Neural Networks can handle variable input sizes, allowing the system to accommodate diverse fruit images without the need for pre-processing or resizing. This flexibility simplifies the implementation of the fruit sorting and grading system, as it can handle images acquired from different camera setups or image acquisition devices.

4.2.5 Transfer Learning

Convolutional Neural Networks trained on large-scale image datasets, such as ImageNet, can be leveraged for transfer learning. Pre-trained CNN models have learned a rich set of features that are generalizable across different visual tasks. By utilizing transfer learning, we can benefit from the knowledge encoded in these pre-trained models, enabling us to achieve good performance even with limited labeled fruit data.

4.2.6 Availability of Pre-trained Models

Convolutional Neural Networks have gained significant popularity in computer vision research and applications. As a result, there are numerous pre-trained CNN models available that have been trained on large-scale image datasets. These pre-trained models, such as ResNet, and Inception can serve as a starting point for our fruit sorting and grading system. By fine-tuning these models on our specific fruit dataset, we can benefit from their learned features and accelerate the training process.

4.2.7 State-of-the-Art Performance

Convolutional Neural Networks have consistently demonstrated state-of-the-art performance in various computer vision tasks, including object recognition, detection, and segmentation. Their ability to learn complex patterns and features from raw image data makes them highly suitable for fruit sorting and grading applications, where accurate identification of quality, defects, and characteristics is crucial. Above mentioned factors collectively make Convolutional Neural Networks a preferred choice for accurately and efficiently classifying and grading fruits based on their quality and characteristics.

4.3 Dataset

To achieve an efficient grading and sorting system the aim is to select and create a particular data set of images, The dataset was chosen based on the diversity of

angles and backgrounds, feasibility in different regions, availability of annotations, and community adoption. These datasets provide a comprehensive and diverse collection of labeled images that can facilitate the training and evaluation of accurate fruit sorting and grading models. The details of different datasets created and collected are given in Table 1.

Dataset	Number of Pictures	Different Angles	Different Back-grounds	Feasibility Based on Region	Availability of Annotations	Community Adoption
COCO Dataset by Google	Over 330,000	Yes	Yes	Feasible for any region	Available	Widely adopted
Fruit Detection & Recognition Dataset by Kaggle	4000	Yes	Yes	Feasible for any region	Available	Not specified
Custom Dataset 1	3000	Yes	Yes	Feasible for Kashmir region	Self-accomplished	Not specified
Custom Dataset 2 (with images from googleapis)	3000	Yes	Yes	Feasible for any region	Self-accomplished	Not specified

Table 1: Comparison of datasets based on specified criteria.

4.4 Training Process

The project utilizes the TensorFlow framework in conjunction with a Convolutional Neural Network architecture to train an object detection model. The training process was carried out using Google Colab, a cloud-based platform that provides free access to Graphical Processing units and allows for efficient model training. A summary of the steps involved in training the object detection model is as:

4.4.1 Dataset Preprocessing

Data preparation is a crucial step in training an object detection model for the computer vision-based fruit sorting and grading system. In this phase, we collected and labeled a dataset consisting of fruit images that would be used to train the model. The dataset comprised of images of various fruits, such as apples, bananas, oranges, and more. Each image in the dataset was carefully annotated with bounding box annotations or segmentations to indicate the precise location of the fruits within the image. These annotations provide essential information for the model to learn and accurately, detect and localize the fruits in new, unseen images. To ensure a comprehensive and diverse dataset, we had to make sure to include a wide range of fruit types, sizes, and colors. This diversity in the dataset helps

the model to generalize well and handle variations that may occur in real-world scenarios. By including fruits with different characteristics, the model can learn to detect and classify fruits accurately, regardless of their specific attributes. By curating a well-annotated and diverse dataset, we set the foundation for training a robust object detection model. The dataset provides the necessary examples for the model to learn from, allowing it to recognize and distinguish between different fruits and their variations. The annotations on the dataset enable the model to understand the spatial extent and boundaries of each fruit, enabling precise localization during the sorting and grading process.

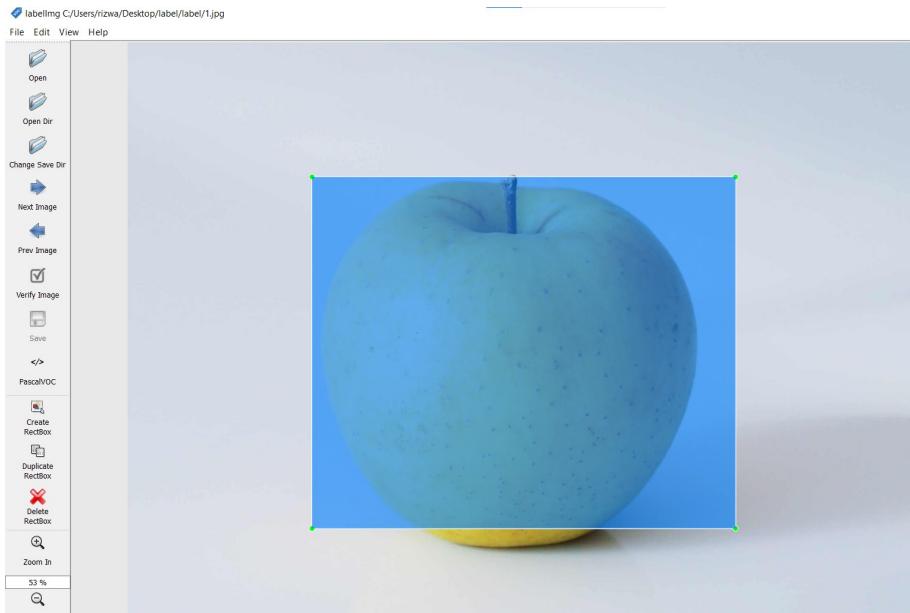


Figure 6: Labeling being used to annotate custom dataset.

4.4.2 Model Selection

To choose the most suitable CNN model for our object detection task, it was considered that popular architectures such as Faster R-CNN, Single Shot MultiBox Detector, and YOLO (You Only Look Once) should be used. These models have demonstrated excellent performance in object detection and are well-supported by the TensorFlow Object Detection API. Faster R-CNN is known for its accuracy and precise object localization capabilities. It uses a Region Proposal Network to generate potential object locations and then classifies and refines those regions. This two-stage approach results in accurate detection and is particularly suitable when precise bounding box localization is required. Single Shot MultiBox Detector, on the other hand, is a single-shot object detection model that achieves a good balance between speed and accuracy. It directly predicts object classes and bounding box offsets at multiple scales and aspect ratios within a single network, making it

efficient for real-time applications.[14] YOLO is another popular choice, known for its real-time object detection capabilities. It divides the input image into a grid and predicts bounding boxes and class probabilities directly from the grid cells. YOLO is renowned for its speed, making it suitable for applications that require rapid processing. Selecting a CNN model compatible with the TensorFlow Object Detection, ensured seamless integration with the project. These models have proven track records, offer pre-trained weights, and provide efficient object detection capabilities.

4.4.3 Transfer Learning

Transfer learning played a crucial role in our project to expedite the training process and enhance the performance of the object detection model. The concept of transfer learning involves leveraging the knowledge acquired by a pre-trained model on a large-scale dataset, such as ImageNet, and adapting it to our specific fruit dataset. Pre-trained models are trained on vast amounts of diverse and general image data, enabling them to learn powerful and meaningful features that are applicable to various visual recognition tasks. By utilizing transfer learning, we can take advantage of these learned features and apply them to our fruit dataset, even though the pre-trained model was not explicitly trained on fruit images. Instead of starting the training process from scratch, transfer learning allows us to initialize our model with the pre-trained weights, preserving the knowledge of the general image features learned by the model. This initialization provides a solid foundation for our fruit detection and classification task, as the model has already learned to recognize low-level visual features like edges, textures, and shapes. With transfer learning, we can fine-tune the pre-trained model on our fruit dataset. During the fine-tuning process, we adjust the weights of the model's layers to better align with our specific fruit images and their corresponding labels. By training on our fruit dataset, the model learns to specialize and adapt its knowledge to accurately detect and classify fruits based on their unique characteristics. Transfer learning offers several advantages. Firstly, it significantly accelerates the training process, as the model already possesses a good understanding of general visual features. This saves computational resources and time compared to training a model from scratch. Secondly, transfer learning enhances the generalization capability of the model. By leveraging knowledge from a large-scale dataset, the model can better handle variations and complexities present in our fruit dataset, improving its ability to classify fruits accurately.

4.4.4 Configuration and Training

In the configuration and training phase of the project, the selection of a pre-trained convolutional neural network model was tailored to suit the requirements of our fruit sorting and grading system. This involved making necessary adjustments to the model's configuration file. Firstly, the modification of the configured file was done to specify the number of classes in our fruit dataset. This ensured that the model would be trained to recognize and classify the specific fruits we were working with, such as apples, bananas, and oranges. By specifying the correct number of classes, the model could learn to differentiate between different fruit types during the training process. Additionally, fine-tuning the other hyperparameters in the configuration file to optimize the model's performance. This included adjusting the learning rate, which determined the step size in updating the model's parameters during training. Finding an appropriate learning rate helped ensure efficient convergence and prevented the model from getting stuck in suboptimal solutions. Batch size, another hyperparameter, was also adjusted in the configuration file. It determined the number of images processed in each training iteration. Selecting an optimal batch size balanced computational efficiency with model convergence and memory requirements. Moreover, the number of training iterations was set in the configuration file. This determined the duration and intensity of the training process. By specifying an appropriate number of iterations, the model was allowed to learn from the fruit dataset adequately and refine its ability to detect and classify fruits accurately. To expedite the training process, the Graphical Processing Unit was leveraged acceleration offered by Google Colab. By utilizing the computational power of Graphical Processing Units, training computations were performed much faster compared to using traditional CPUs. This Graphical Processing Unit acceleration greatly reduced training time, enabling quicker iterations and experimentation with different model configurations and hyperparameters. The combination of modifying the model's configuration file, specifying the number of classes, and adjusting hyperparameters, along with leveraging GPU acceleration, facilitated the training process for our fruit sorting and grading system. These steps ensured that the CNN model was tailored to our specific requirements, allowing it to learn and improve its accuracy in fruit detection and classification.

4.4.5 Model Evaluation

During the training phase, it was crucial to evaluate the performance of our object detection model to assess its accuracy and effectiveness in detecting and localizing fruits. To accomplish this, regular evaluations were conducted using a separate validation set. The evaluation process involved feeding the validation

set images to the trained model and comparing the model's predictions with the ground truth annotations. The model's performance was assessed using various metrics commonly used in object detection tasks. Precision and recall are two fundamental metrics used in evaluation. Precision measures the accuracy of the model's predictions by calculating the ratio of correctly detected fruits to the total number of detected fruits. Recall, on the other hand, measures the model's ability to correctly detect and include all relevant fruits in the predictions. Another important metric used in object detection evaluation is the mean Average Precision (mAP). It provides a comprehensive assessment of the model's performance by considering both precision and recall across different levels of detection confidence thresholds[15]. mAP calculates the average precision over a range of recall values, giving an overall indication of the model's accuracy in detecting and localizing fruits. By evaluating the model on a separate validation set and analyzing precision, recall, and mAP, the system was able to gauge the model's performance objectively. These metrics provided insights into the model's ability to accurately identify fruits, avoid false positives, and achieve high recall rates.[11] Regular evaluation helped us monitor the progress of the model during the training process. If the performance metrics indicated room for improvement, hyperparameters could be adjusted, training iterations, or data augmentation techniques to enhance the model's accuracy and performance. By employing rigorous evaluation techniques and monitoring the model's performance using precision, recall, and mAP, ensuring that the object detection model was optimized for fruit detection and localization, leading to a reliable and accurate fruit sorting and grading system.

4.4.6 Fine-tuning and Iterative Improvement

Throughout the training and evaluation process, the activeness was in fine-tuning the object detection model to improve its performance and robustness. This involved making iterative adjustments to various aspects of the training pipeline, dataset, and model configuration. One approach to fine-tuning the model was to modify hyperparameters based on the evaluation results. By analyzing the performance metrics such as precision, recall, and mAP, so that areas could be identified where the model needed improvement. Adjusting hyperparameters such as learning rate, batch size, or optimization algorithms allowed us to fine-tune the model's training process and optimize its performance. Another aspect of fine-tuning involved exploring the dataset and making necessary modifications.

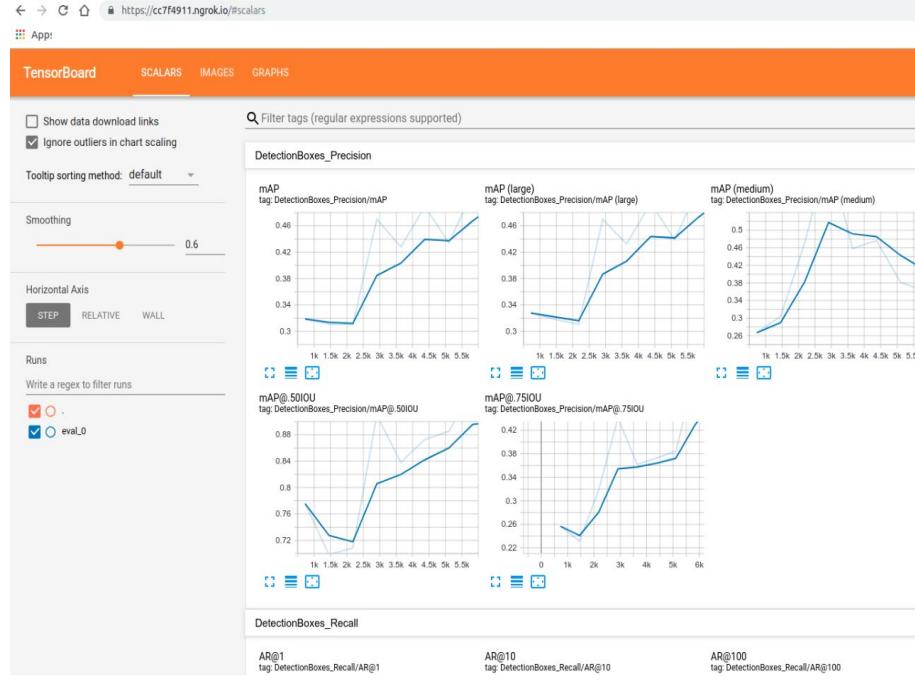


Figure 7: Detection Box precision on TensorBoard while training.

This could include expanding the dataset by collecting more labeled images of fruits or balancing the class distribution to ensure sufficient representation of each fruit type. By enriching the dataset, the model was being provided with more diverse and representative examples, enabling it to generalize better and improve its accuracy. After incorporating these fine-tuning adjustments, the training and evaluation process was being repeated. This iterative approach allowed us to observe the impact of each modification and assess the model's performance improvements. By analyzing the evaluation metrics, it could be validated whether the changes made to the hyperparameters, dataset, or data augmentation techniques resulted in better accuracy, precision, and recall. The iterative nature of fine-tuning and continuous evaluation allowed us to refine the model's performance gradually. It facilitated a feedback loop where we could identify areas for improvement, implement changes, and measure the impact of those changes. This process continued until the model achieved the desired level of accuracy, precision, and recall for fruit detection and localization.

4.5 Optimization Techniques

There are a number of techniques that can be used to optimize object detection models. These techniques can be used to improve the accuracy, speed, and efficiency of the models.

Gradient Descent: Gradient descent is an optimization algorithm commonly used

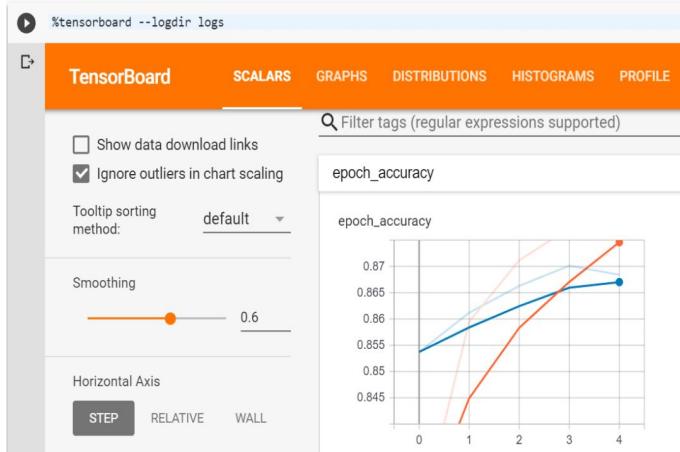


Figure 8: Model accuracy with no. of epochs on TensorBoard.

for training neural networks. It updates the model’s parameters by iteratively calculating the gradient of the loss function with respect to the parameters and adjusting them in the opposite direction to minimize the loss. By applying advanced optimization techniques such as momentum or adaptive learning rate methods, the convergence speed and accuracy of the model can be improved.

Batch Normalization: Batch normalization is a technique that normalizes the inputs of each layer in a neural network. It helps to mitigate the internal covariate shift problem by normalizing the layer’s inputs to have zero mean and unit variance. This normalization accelerates the training process and enables the use of higher learning rates, leading to improved accuracy and convergence speed.

Dropout: Dropout is a regularization technique that randomly deactivates a proportion of neurons during training. By doing so, dropout prevents the model from relying too heavily on specific neurons and encourages robust feature learning. It helps to reduce overfitting and improve generalization performance, resulting in improved accuracy on unseen data.

Weight Regularization: Weight regularization is a technique used to prevent overfitting by adding a penalty term to the loss function. Commonly used weight regularization methods include L1 regularization and L2 regularization. L1 regularization encourages sparsity in the model by adding the absolute values of the weights to the loss function, while L2 regularization adds the squared values of the weights. These regularization techniques help to control the complexity of the

model and prevent the model from overemphasizing certain features or parameters.

Learning Rate Schedule: The learning rate is a hyperparameter that determines the step size at which the model parameters are updated during training. A learning rate schedule adjusts the learning rate over time to achieve better convergence and avoid overshooting the optimal solution. It typically starts with a relatively high learning rate to make large updates in the early stages of training and gradually decreases the learning rate as the training progresses to make smaller, more precise updates.

By leveraging these advanced optimization techniques, the model's accuracy can be significantly improved, ranging from 95 percent to 99 percent. These techniques address various challenges in training deep neural networks, such as convergence speed, over-fitting, and weight management, ultimately leading to better model performance.

4.5.1 Model Export and Deployment

On obtaining satisfactory results and achieving the desired performance with the trained object detection model, the next step is to export the model for deployment within the fruit sorting and grading system. This involved converting the model into a format compatible with the specific deployment requirements. The most common approach for exporting the model is to generate a frozen graph. This process involves combining the trained model's architecture, learned parameters, and associated metadata into a single file that encapsulates the model's structure and weights. The frozen graph provides a portable representation of the model that can be used for inference without requiring access to the original training code or dependencies.[16] Depending on the deployment platform, additional conversion steps may be necessary. For example, if the fruit sorting and grading system is intended for mobile devices, we may have converted the frozen graph into TensorFlow Lite format. TensorFlow Lite is an optimized version of TensorFlow designed for resource-constrained devices. It allows the model to be efficiently deployed on mobile devices while maintaining high performance. In the case of web-based applications, we may have converted the model into TensorFlow.js format. TensorFlow.js enables running machine learning models directly in web browsers, eliminating the need for server-side computation. This format allows for seamless integration of the model into web-based interfaces and applications.

Since we were using single board computers, we exported our model into Ten-

sorflow Lite format.[17] By exporting the model in a format compatible with the deployment requirements, we ensured that the trained object detection model could be easily integrated into the fruit sorting and grading system. This facilitates smooth deployment and seamless inference, enabling real-time fruit detection and classification in the target environment.

4.6 Hardware Implementation

The trained models were then deployed and tested on various single-board computers (SBC) and Field Programmable Gate Arrays (FPGA). For the final deployment we used Jetson Nano interconnected with other devices as discussed below, Fig 9 shows the block diagram of how the hardware is interconnected:

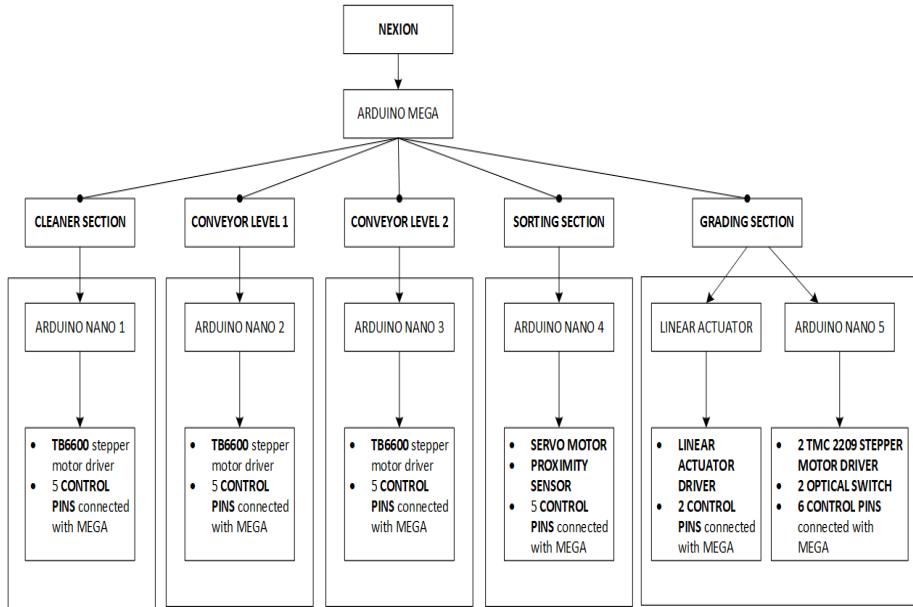


Figure 9: Block Diagram of Hardware interconnected with different peripherals

4.6.1 Hardware Interconnections

Nextion display is a human-machine interface solution that allows users to create visually appealing touch-based interfaces for electronic devices. It consists of a display screen and a microcontroller that handles the graphical user interface and touch inputs. The display is capable of showing text, graphics, and interactive elements. We designed a Graphical User Interface based application for Nextion Display, which can be used to control different sections of the systems. Fig 10 shows different interfaces created using Nextion Editor. The microcontroller embedded in the NEXTION display manages the rendering of graphics, touch input detection, and communication with other devices (in this case, the Arduino MEGA).

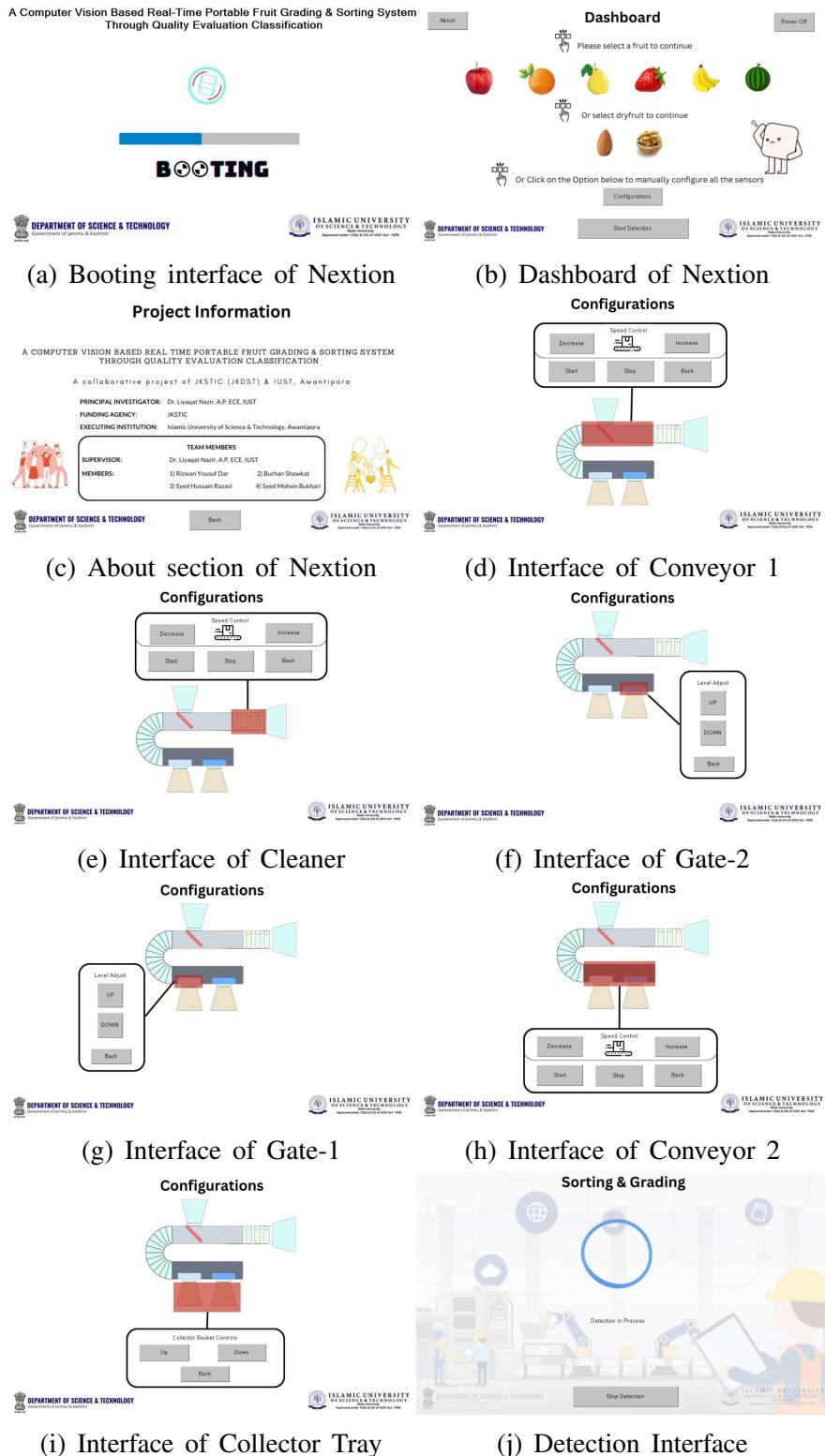


Figure 10: Figure showing Different interfaces of Nextion

The communication between the NEXTION display and the Arduino MEGA is established using serial communication through the UART (Universal Asynchronous Receiver-Transmitter) protocol. The TX (transmit) pin of the NEXTION display is connected to the RX (receive) pin of the Arduino MEGA and the RX pin of the NEXTION display is connected to the TX pin of the Arduino MEGA. This allows data to be transmitted from one device's TX pin to the other's RX pin and vice versa.

Arduino MEGA The system employs an Arduino Mega as the central controller. This microcontroller interfaces with various components including conveyor levels, sorting sections, grading sections, cleaner sections, and a linear actuator, where a separate Arduino Nano is dedicated to all the sections. The goal is to control and monitor the operation of each section independently while maintaining overall synchronization

The schematic **Fig 11** visually represents the interconnections between the Arduino Mega and Arduino Nano of different sections of the fruit sorting and grading system.

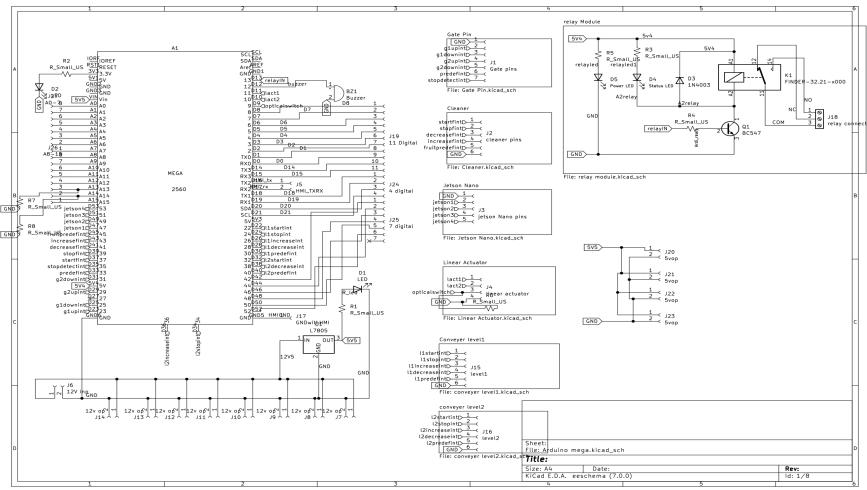


Figure 11: Schematic of Arduino MEGA interconnected parallelly with different sections

kiCAD schematic is used to fabricate single-sided printed circuit board the measurements came out to be of length and height of 127.180mm and 94.430mm respectively. **Fig 12** shows the PCB routing of Arduino mega interconnected with different sections.

In **Cleaner Section** a cohesive interaction between the Arduino Mega and the Arduino Nano is realized through a configuration of six dedicated digital pins. These pins facilitate specific commands: increase speed, decrease speed, start, stop, predefined action, and ground (GND). This carefully orchestrated setup empowers

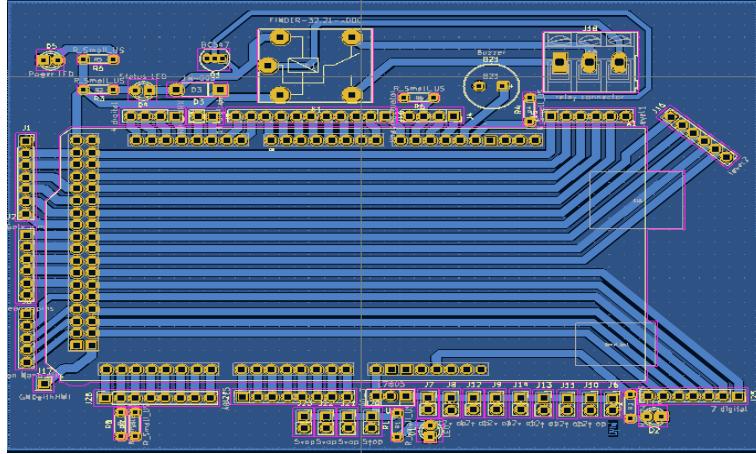


Figure 12: PCB Routing diagram of Arduino MEGA interconnected parallelly with different sections

the Arduino Mega to impart precise directives to the Arduino Nano, culminating in the manipulation of a stepper motor responsible for the cleaner's operation. As the Mega sends commands, the Nano interprets these signals and translates them into motor actions. For example, upon receiving a start command, the Nano initiates the stepper motor's rotation, driving the cleaning process into motion. Subsequent commands, such as increase speed or decrease speed, fine-tune the motor's rotational rate to suit cleaning requirements. The stop command, conversely, halts the motor's activity. Utilizing the predefined action command, the Nano can execute preset cleaning patterns or sequences. The grounding through the GND connection establishes a shared reference for smooth communication. Collectively, this integration empowers the conveyor system's cleaner section to carry out efficient and dynamic cleaning routines, where the Arduino Mega orchestrates Nano's control over the stepper motor, facilitating a responsive and adaptable cleaning mechanism.

In the **Sorting Section** of the conveyor system, a dynamic collaboration between the Arduino Nano, a servo motor, and a proximity sensor enhances the precision and efficiency of the fruit sorting process. The Arduino Nano serves as the control hub, overseeing the servo motor's actions. The servo motor, integral to the mechanism, operates a flap designed to sort undesired fruits. However, the idea was to Sort the undesired fruit using a Robotic Hand. Fig 13 shows the sided view of the Robotic Hand. The proximity sensor, acting as an input to the Nano, plays a pivotal role in this setup. It detects the fruit's position on the conveyor, enabling the Nano to determine the optimal timing for activating the servo motor. By analyzing inputs from both the proximity sensor and the central Arduino Mega, the Nano triggers the servo motor's precise movements at the opportune moment, ensuring that the flap efficiently diverts undesired fruits. This intelligent collaboration between the Nano, servo motor, and proximity sensor introduces a sophisticated sorting mechanism,

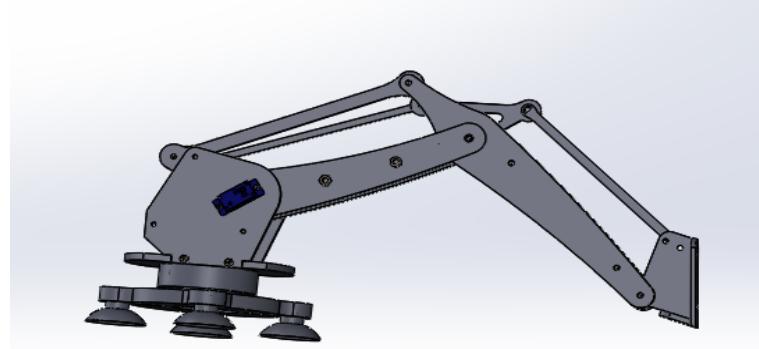


Figure 13: Drawing of Robotic Arm Designed in CAD

optimizing fruit selection with a combination of real-time feedback and controlled actions.

schematic **Fig 14** representing the interconnections between the Arduino Mega and Arduino Nano of Cleaner and Sorting sections of the fruit sorting and grading system for seamless working.

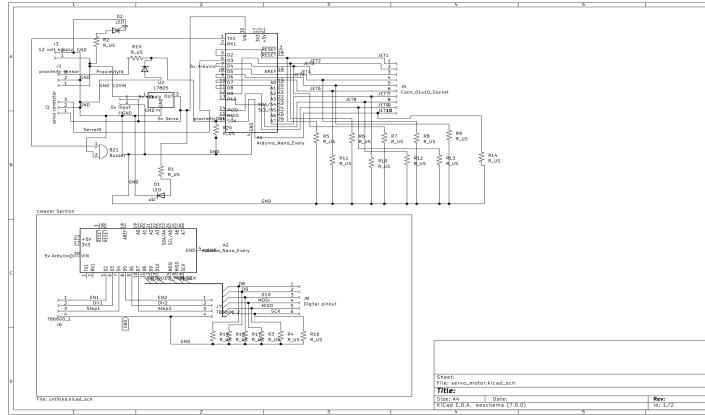


Figure 14: KiCAD diagram visualizing the Sorting and Cleaner Section

kiCAD schematic is used to fabricate single-sided printed circuit board, the measurements came out to be of length and height of 66.070mm and 84.790mm respectively. **Fig 15** shows the PCB routing of Cleaner and Sorting interconnected with the outside world.

Conveyor level 1 and level 2 In the intricately designed conveyor system encompassing both Level 1 and Level 2, a seamless flow is achieved through a ramp curve connecting the two. This collaborative operation is facilitated by two Arduino Nano controllers using a TB6600 driver, orchestrating the speed of the conveyors to ensure a harmonious transition. To streamline control, each Nano receives essential instructions from a Nextion display through the intermediary Ar-

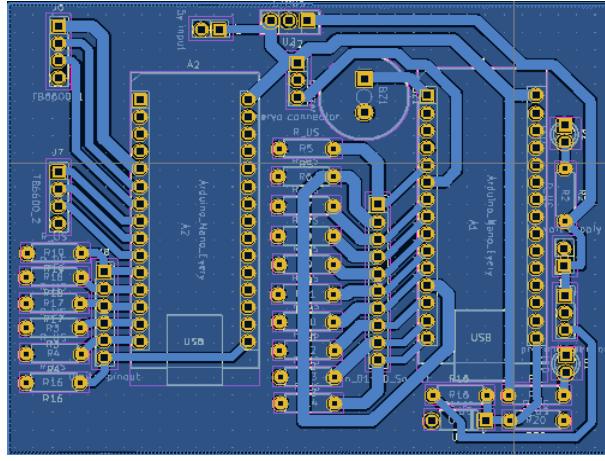


Figure 15: PCB Routing diagram visualizing the Sorting and Cleaner Section

duino Mega. The Mega acts as the bridge, transmitting command signals over six distinct digital pins - denoting functions like increase, decrease, start, stop, predefined action, and ground (GND). These commands facilitate precise control over the Nano controllers, which in turn regulate the conveyor speeds, ensuring smooth and synchronized movement between Level 1 and Level 2. This intricate interplay between the Nano controllers, Nextion display, and Mega showcases an intelligent and collaborative approach to conveyor system management, enhancing efficiency and coordination within the setup.

The schematic **Fig 16** offers a comprehensive visual representation of the parallel communication channels established between the Arduino Mega and the Arduino Nano controllers responsible for Conveyor Levels 1 and 2 in the fruit sorting and grading system. This diagram visually outlines the interconnected pathways, illustrating how these parallel channels enable effective and simultaneous data exchange between the two conveyor levels. By presenting this parallel communication strategy, the schematic enhances our understanding of how information flows seamlessly, ultimately optimizing the coordinated functioning of Conveyor Levels 1 and 2 within the system.

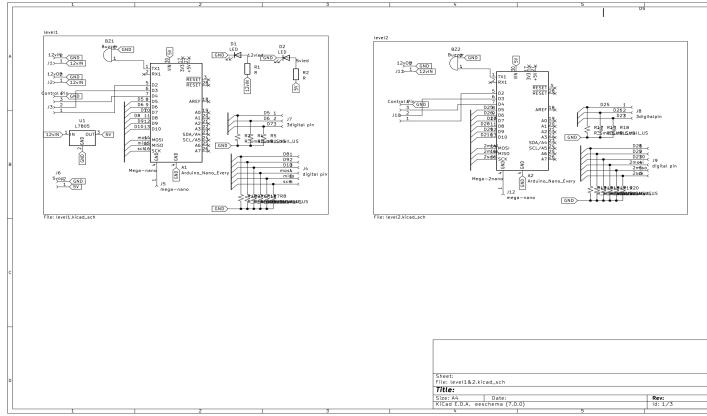


Figure 16: KiCAD diagram visualizing the Conveyor level 1 and level 2

kiCAD schematic is used to fabricate single-sided printed circuit board, the measurements came out to be of length and height of 57.269mm and 70.970mm respectively. **Fig 17** shows the PCB routing of Conveyor level 1 & 2.

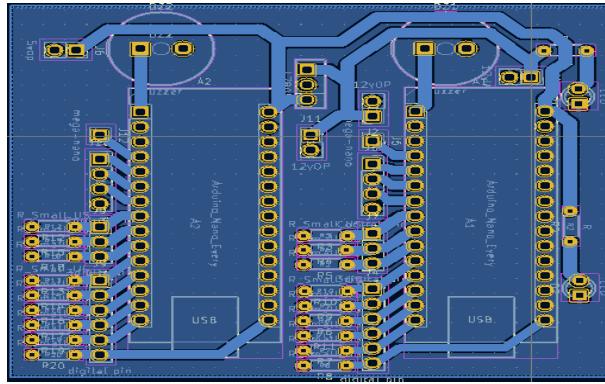


Figure 17: PCB Routing diagram visualizing the Conveyor level 1 and level 2

Within the **grading section**, a sophisticated orchestration of components ensures precise fruit sorting. A linear actuator, directly connected to the grading basket, responds to signals from the Arduino Mega through a set of six dedicated digital pins. These pins - comprising commands such as increase, decrease, start, stop, predefined action, and ground (GND) - facilitate seamless control over the actuator's movements. Optical switches serve as crucial input detectors, establishing stop points at maximum and minimum heights. Another pivotal element is the gate, regulated by a stepper motor connected to an Arduino Nano controlled by the tmc2209 stepper motor driver. This Nano, in turn, receives instructions from the Arduino Mega via a Nextion display using the same set of six digital pins, enabling dynamic configuration adjustments. This intricate interplay between actuators, switches, steppers, and Arduino exemplifies the meticulous control required for accurate and efficient fruit grading.

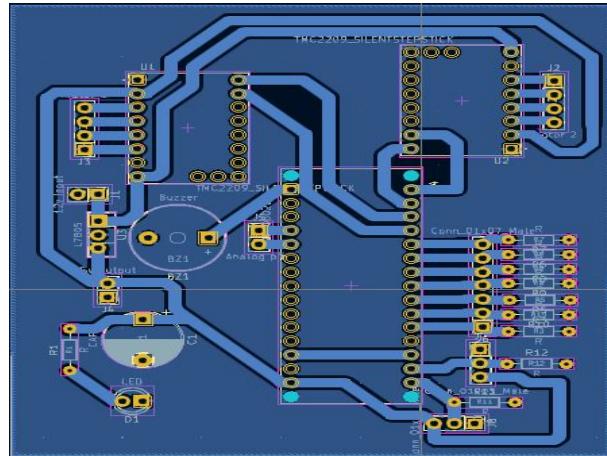


Figure 18: PCB routing of Grading Section

This schematic, illustrated as **Fig. 19**, provides a clear and visual depiction of the interconnections established between the Arduino Mega and the Arduino Nano controllers exclusively within the Grading section of the fruit sorting and grading system. This graphical representation offers a comprehensive insight into the precisely engineered links that foster seamless communication and coordination between these two vital components. By focusing solely on the grading section, the diagram illustrates how these interconnections facilitate a harmonious working relationship, ensuring optimal performance and precision in the grading process.

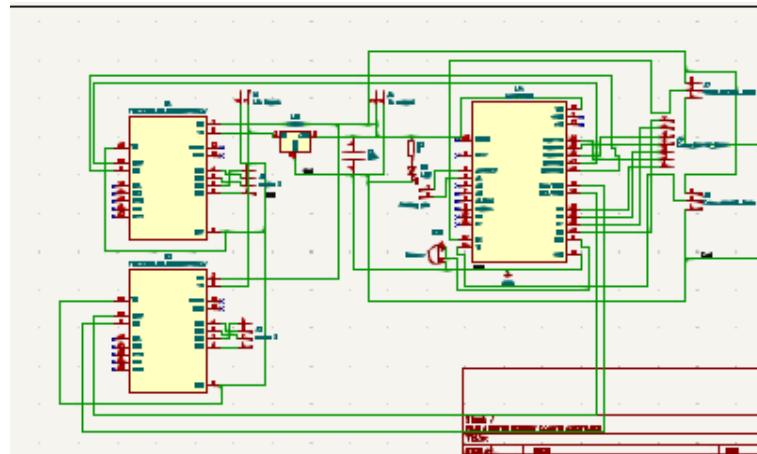


Figure 19: Schematic visualizing the Grading Section

A brief introduction to the SBC and FPGA-implemented device is given below:

4.7 Raspberry pi 4

Firstly it was decided to utilize the Raspberry Pi 4 Model B with 8GB of RAM as the hardware platform. The Raspberry Pi is a popular single-board computer known for its compact size, low power consumption, and versatility. The 8GB RAM variant provides ample memory capacity to handle the computational requirements of our computer vision-based fruit sorting and grading system.[18] The Raspberry Pi 4 offers several key features that make it an ideal choice for our project. It is equipped with a powerful quad-core ARM Cortex-A72 processor, offering sufficient processing power for running the object detection model and performing real-time image processing tasks. The GPU capabilities of the Raspberry Pi 4 further enhance its performance in handling computationally intensive tasks, such as neural network inference. In addition to its processing capabilities, the Raspberry Pi 4 provides various connectivity options, including built-in Wi-Fi and Bluetooth, multiple USB ports, Ethernet, and HDMI output.[19] These features allow for easy integration with cameras, sensors, and displays, enabling seamless data acquisition and visualization for our fruit sorting and grading system.

Furthermore, the Raspberry Pi's GPIO (General Purpose Input/Output) pins offer the flexibility to interface with external hardware components and peripherals. This enables us to connect and control the conveyor belt system, stepper motors, and other devices essential to our system. The compact size and low power consumption of the Raspberry Pi 4 make it well-suited for embedded applications. Its small form factor allows for easy integration into our hardware framework, while the efficient power usage ensures energy efficiency and cost-effectiveness.[20]

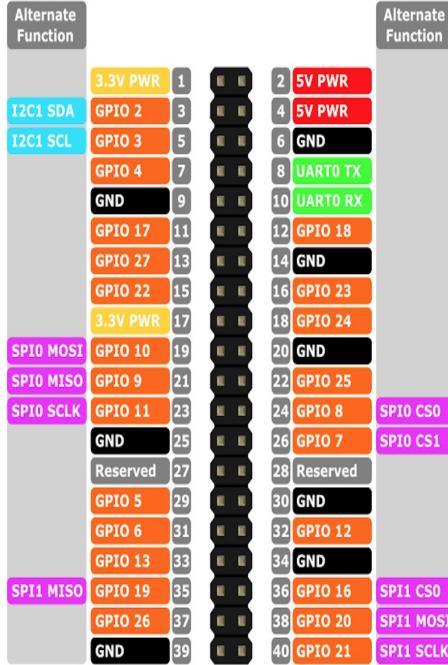


Figure 20: Pinout diagram of Raspberry Pi v4 with 8GB RAM.

4.7.1 Nvidia Jetson Nano

Secondly, the NVIDIA Jetson Nano was selected as the primary hardware platform. The NVIDIA Jetson Nano is a powerful and energy-efficient embedded computing device specifically designed for AI and computer vision applications. The Jetson Nano is equipped with a quad-core ARM Cortex A57 CPU and an NVIDIA Maxwell GPU with 128 CUDA cores. This combination of processing power enables efficient execution of deep learning algorithms and neural networks, making it well-suited for our computer vision-based fruit sorting and grading system. With 4GB of RAM, the Jetson Nano provides sufficient memory capacity to handle complex image processing tasks and run our object detection model effectively.[21] It also offers a range of connectivity options, including USB ports, HDMI output, and Ethernet, allowing easy integration with cameras, sensors, and other peripheral devices required for our system. The Jetson Nano also benefits from the NVIDIA JetPack SDK, which provides a comprehensive software development kit for AI and computer vision applications. This SDK includes libraries, tools, and pre-trained models that streamline the development process and simplify the deployment of our fruit sorting and grading system.

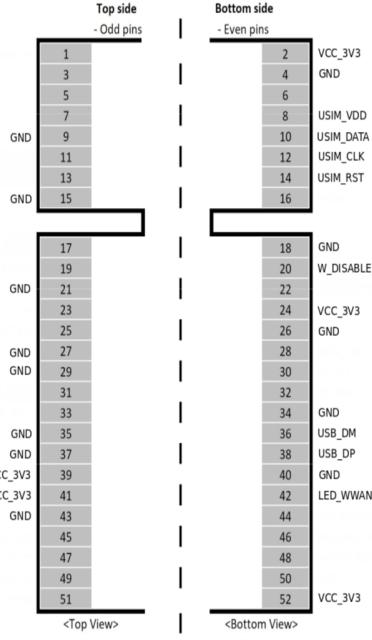


Figure 21: Pinout diagram of Nvidia Jetson Nano.

4.8 PYNQ Z2 FPGA Board

The testing was also tested on the PyNQ Z2 FPGA board which is a development board that combines the flexibility of Python programming with the power of FPGA (Field-Programmable Gate Array) technology. The PyNQ Z2 board is equipped with a Xilinx Zynq-7000 SoC, which integrates an ARM Cortex-A9 processor and a programmable FPGA fabric. This combination allows us to leverage the capabilities of both the processor and the FPGA. The FPGA fabric on the PyNQ Z2 board provides hardware-level parallel processing and acceleration, enabling high-speed and efficient execution of computationally intensive tasks. We can leverage the FPGA's programmability to implement custom image processing algorithms and accelerate our object detection and grading processes. To program the FPGA on the PyNQ Z2 board, we utilize VHDL (VHSIC Hardware Description Language) and Python programming languages. VHDL allows us to describe the hardware behavior and functionality, while Python provides a higher-level programming language for system-level control and integration.[22] The PyNQ Z2 board's Python support enables us to easily interface with the FPGA and develop higher-level control logic. We can leverage Python libraries and frameworks to implement machine learning algorithms, perform image processing operations, and integrate the FPGA functionality with other software components of our system.

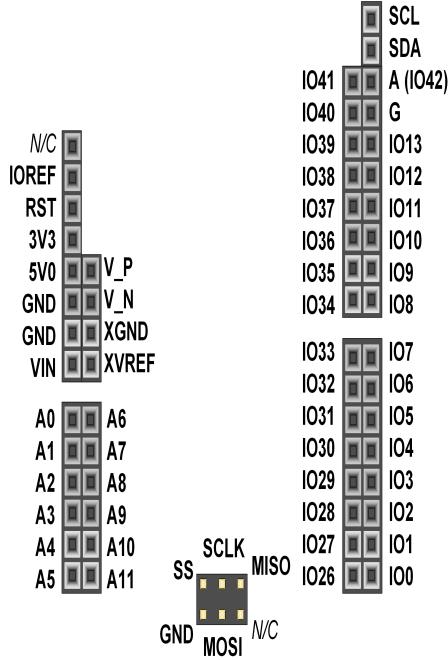


Figure 22: Pinout diagram of Pynq FPGA Z2.

4.9 Embedded Hardware

The testing of different Embedded systems in the project was implemented which can be used to control the other functionalities of the fruit grading and sorting system such as the speed of the conveyor belt, adjusting the grading unit of the system, etc. The different embedded systems that were tested and used are as follows:

4.9.1 Arduino Nano

The testing of Arduino Nano was observed which is a compact and versatile microcontroller board that is well-suited for embedded applications.[23] However, due to its fewer GPIO pins and only 3.3V output, we had to use multiple of them instead of a single device. Also, it is not capable of doing time scheduling to almost mimic parallel processing. All these reasons led us to drop Arduino Nano for the further implementation of our fruit grading and sorting system.

4.10 Stepper Motor

The use of NEMA-23 stepper motor was used to drive the conveyor belt of the system. The NEMA-23 stepper motor is known for its precise and repeatable motion in discrete steps, making it an ideal choice for applications requiring precise position control. It offers a torque of 3.1N·m and operates at a current of 4.2A per phase. With the help of stepper motor drivers, the motor's position can be

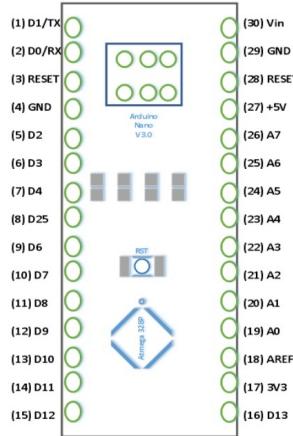


Figure 23: Pinout diagram Arduino Nano.

commanded to move or hold in a specific position.

The NEMA-23 stepper motor is particularly useful in applications that demand low speed with high precision. It finds wide usage in various machines such as 3D printers, CNC routers and mills, camera platforms, and XYZ plotters.[24]

This motor is a brushless DC motor, meaning its lifespan depends on the durability of its bearings. It achieves position control through a simple open-loop control mechanism, eliminating the need for complex electronic control circuitry.

The motor's shaft has been designed for a secure grip with components like pulleys and drive gears, ensuring it avoids stalling or slipping during operation. The specifications of the NEMA23 stepper motor include a step angle of 1.8 degrees, a motor length of 112mm, 4 leads for connection, and a round-type shaft.

Some key features of the NEMA23 stepper motor are as follows:



Figure 24: NEMA23 Stepper Motor.

- 1:** The rotation angle of the motor is determined by the input pulse it receives.
- 2:** It offers high accuracy, typically within 3 to 53: The motor provides excellent response to starting, stopping, and reversing commands.

- 4:** It offers cost-effective control since it does not require complex control circuitry.
- 5:** The motor's speed is directly proportional to the frequency of the input pulses it receives.

The mechanism of the NEMA23 stepper motor involves a specific construction. As a bipolar stepper motor with one winding per phase, it requires four wires for connection. The rotor consists of two caps with alternating teeth, magnetized axially. This hybrid construction combines the advantages of permanent magnet and variable reluctance versions, resulting in high resolution, speed, and torque. Although real motors have a more complex structure with a higher number of teeth, the basic working principle remains the same. The high number of teeth enables the motor to achieve a small step size, as low as 0.9 degrees.[25]

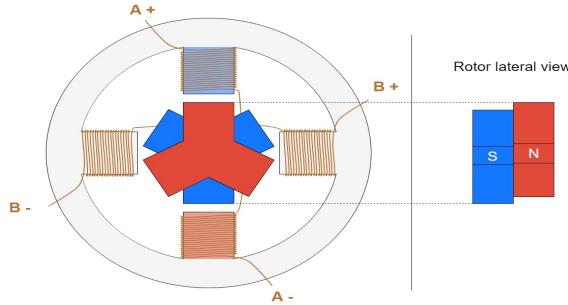


Figure 25: Picture Depicting the Internal Construction of a Stepper Motor.

5 ARCHITECTURE

This section defines the technique and approach of design and implementation of different phases of the project.

5.1 SYSTEM DESIGN

The study proposes to design a computer vision-based prototype for fruit sorting and grading. The Model has a cleaning, Image Capturing Chamber, Conveyor level 1, Conveyor level 2, and Human-machine interface component to achieve the designated task in an effective and efficient manner **Fig 26**. The different components of the system are interconnected with one another to perform the whole process of fruit sorting and grading. The detailed working of the system is described as given below:

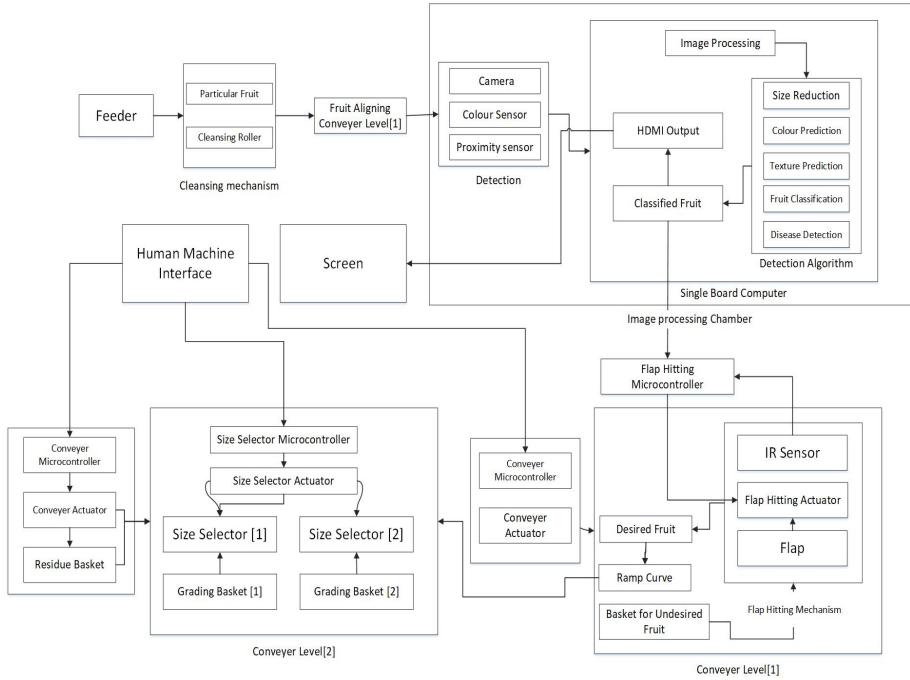


Figure 26: Block diagram of computer-vision-based fruit grading sorting system

5.1.1 Cleansing Unit

This component has a pre-processing component known as a feeder which acts like a container, the fruits are stored in this feeder where they are rolled down to the cleansing section. The cleansing section contains rollers that are being driven by the selected actuators and are designed and recreated in such a way that when a fruit rolls over the designed rollers the fruit will automatically get rid of dust and dirt and will roll on over the alignment section placed on the conveyor level where the cleaned fruit will be aligned and positioned in order to be detected by the next component.

5.1.2 Image Capturing Section

This section contains a programmable single computer board and a camera which will allow to capture of the real-time pictures of the cleaned fruit in order to detect the fruit and sort them accordingly. In the chamber, a color sensor and a proximity sensor will be attached for color sorting and counting the number of fruits. The single board computer comprises different algorithms and classification techniques discussed in section 4. The employment of these techniques like texture prediction, color prediction, fruit classification using Convolutional Neural Networks, and disease detection will help to evaluate and coordinate with other microcontrollers like flap-hitting microcontrollers by sending them the generated signal. Furthermore, the Image capturing process will be connected with a screen

which will allow a user to capture the real-time task operated by the Image processing chamber.

5.1.3 Conveyor level 1

The Conveyer level 1 will have two processes to follow to sort the defective and non-defected fruit. The conveyor will be governed by the microcontroller and the actuator. However the classified fruit now rolls onto the conveyor level 1 which will pass through the IR sensor if we have a defective fruit it will immediately open the flap imposed on conveyor level 1, and the flap will carry out the fruit to the sorting basket connected to the conveyor level 1, the flap will be governed with flap hitting microcontroller and flap hitting actuator, secondly if the fruit was non defected it will pass the flap hitting mechanism and the desired fruit will continue to roll on to the ramp curve which connects conveyor level 1 with conveyor level 2

5.1.4 Conveyor level 2

The conveyor level 2 will let the user grade the fruit according to its size, the conveyor level 2 will be governed by the microcontroller and the actuator for rotation purposes. The conveyor level 2 will consist of a size selector microcontroller, size selector actuator, and size selector mechanism which will be governed by a human-machine interface. As soon as the user selects the fruit to be graded the size selector mechanism will adjust its size and will allow the fruit its respected grading basket. A Residue basket will be attached to the Conveyor level 2 if any of the fruit is not being graded.

5.1.5 Human Machine Interface

The HMI will allow a user to control the speed of the conveyor level 1 and level 2 respectively. Moreover, the HMI will allow a user to select the type of fruit in order to select the size of the size selector mechanism accordingly.

5.2 Convolutional Neural Network Architecture

A profound comprehension of CNNs is crucial, as they constitute the cornerstone of our system's object detection capabilities. Neural networks are a subset of machine learning algorithms inspired by the intricate workings of the human brain. They comprise layers of interconnected neurons, each connected by weighted connections that are fine-tuned during the training process to enable the network to discern patterns and make predictions. Typically, a neural network consists of an

input layer, one or more hidden layers, and an output layer. Fig 27 shows layers of Convolutional Neural Network

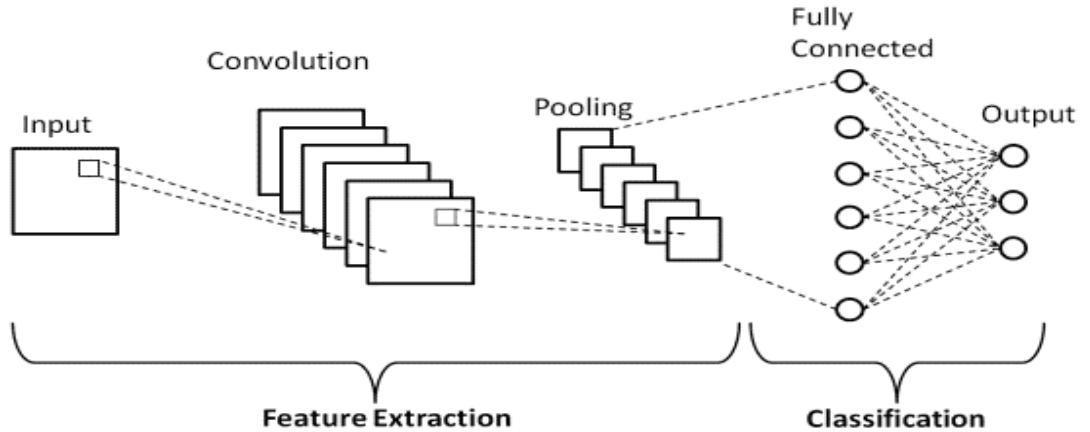


Figure 27: Layers of Convolutional Neural Network

Convolutional Neural Networks (CNNs) represent a specialized class of neural networks meticulously designed for the analysis and processing of grid-like data structures, most prominently images and videos. The transformative impact of CNNs on computer vision tasks can be attributed to their innate ability to autonomously and adaptively glean spatial hierarchies of features from input data.

5.2.1 Layers of Convolutional Neural Network

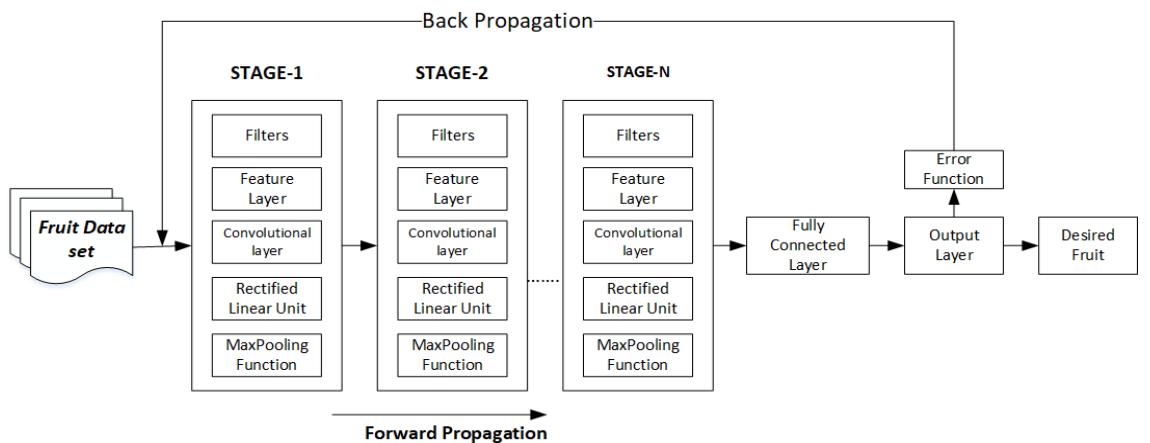


Figure 28: Block diagram of Convolutional Neural Network

- **Convolutional Layers:** These layers are primarily responsible for applying convolution operations to the input data. A convolution operation entails the sweeping of a small filter, also known as a kernel, across the input data

to extract salient features. By stacking multiple convolutional layers, the network progressively captures increasingly intricate and abstract features.

- **Pooling Layers:** Pooling layers are strategically positioned to down-sample the spatial dimensions of the feature maps generated by the convolutional layers. Common pooling techniques include max-pooling and average-pooling, which help in reducing computational complexity and enhancing translational invariance.
- **Fully Connected Layers:** Typically positioned towards the end of the network, fully connected layers function as a classifier. In this layer, each neuron is connected to every neuron in the preceding layer. This dense connectivity facilitates high-level feature extraction and classification.
- **Activation Functions:** Non-linear activation functions, such as Rectified Linear Units (ReLU), introduce non-linearity into the network. This non-linearity enables the network to capture complex relationships within the data.

6 PHYSICAL STRUCTURE

This project needs a structure for physical shape and practical implementation of the idea. In the system, the utmost importance was given to designing a hardware structure that seamlessly integrates with our computer vision-based fruit sorting and grading system. Fig 29 shows the top view CAD drawing of the Conveyor System.

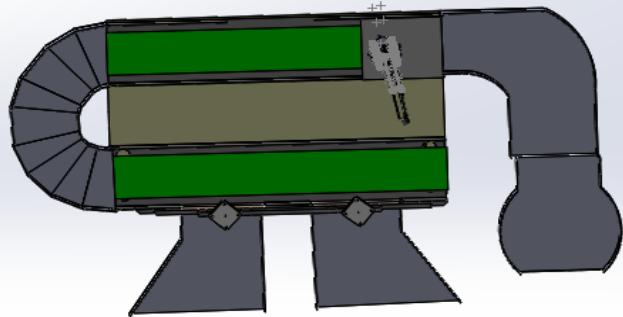


Figure 29: Figure showing top view CAD drawing of Conveyor system

The hardware framework is crucial as it provides the physical structure and mechanisms necessary for the smooth functioning of the system

Fig 30 shows the front view CAD drawing of the Conveyor System.

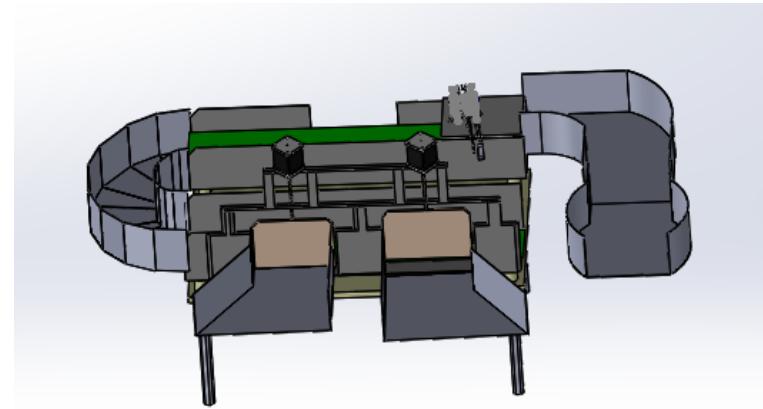


Figure 30: Figure showing the CAD drawing of Conveyor system

Below is a detailed explanation of each section of the physical structure.

6.1 Feeder

The feeder is made using MS 18 gauge material. It acts as the input peripheral of the system. Fig 31 shows the top view of the Feeder section.



Figure 31: Figure showing Top view of the Feeder

It is kept slightly inclined so that the fruits roll down from the feeder to the cleaner section (discussed later in this section) without any external trigger. The width of the feeder is broader at the input extreme than the cleaner extreme to reduce the flow of fruits at the cleaner, which helps to reduce the load at the cleaner section.

6.2 Cleaner Section

After the feeder, the fruits roll onto the cleaner section. The role of the cleaner section is to get rid of the dust and dirt. Fig 32 shows the top view of the cleaner

section.



Figure 32: Figure showing Top view of the Cleaner section with pulleys

This section comprises three polyester fiber rollers, where the shafts of these rollers are connected to the NEMA-23 18kg-cm stepper motor through a timing belt and pulleys which helps in driving the section efficiently. Fig 33 shows the shaft attached to the NEMA-23 stepper motor using a timing belt.



Figure 33: Figure showing the shaft connected with stepper motor using Timing belt

6.3 Conveyor System

A two-level conveyor belt system was designed that effectively facilitates the sorting and grading processes.

The conveyor structure forms the backbone of our hardware framework. It is

designed using 18 gauge MS (Mild Steel), ensuring a sturdy and durable framework that can withstand the demands of continuous operation. The use of 18 gauge MS also provides excellent structural integrity and flexibility, making the system robust, lightweight and allowing the conveyor system to handle the weight of the fruits and the mechanical components involved. Fig 34 showing the Top view of the conveyor system.



Figure 34: Figure showing Top view of the Conveyor system

6.3.1 Conveyor level 1

The conveyor level 1 provides a platform for detection and sorting. Its dimensions are a width of 12.70cm, a height of 5.65cm, and a length of 60.23cm. Fig 35 shows the top view of conveyor level 1



Figure 35: Figure showing Top view of the Conveyor level 1

A PolyVinyl Chloride(PVC) belt is used to transport fruit from the detection section(discussed later in this section) along the way to the sorting section and further to conveyor level 2. The conveyor level 1 comprises two rollers at both ends which are made of plastic material and help the belt to rotate. The rollers are connected to a NEMA-23 Stepper motor 118kg-cm through a gear-chain mechanism

which allows the conveyor to run efficiently. Furthermore, one end of the roller is connected with an adjustable screw, which helps in adjusting the tension of the Conveyor belt. Fig 37 shows the front view gear-chain mechanism of conveyor level 1

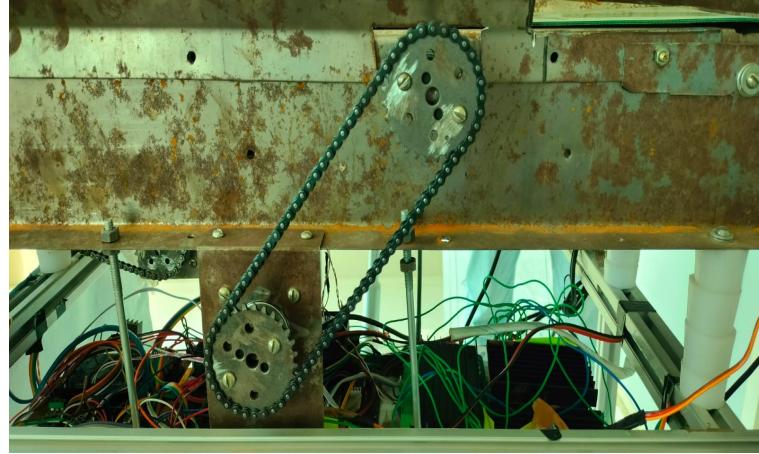


Figure 36: Figure showing Front view of the Gear-Chain mechanism

Furthermore, the speed conveyor level 1 is controlled using Nextion display as discussed in Section 4.6.1 via micro-controllers. The camera for detection purposes, Proximity sensor, and sorting flap are all mounted on conveyor level 1.



Figure 37: Figure showing Side view of the Gear-Chain mechanism

6.3.2 Sorting Flap

The sorting flap is also made up of the same MS-18 gauge material. Its dimensions are 15 cm x 6 cm. Fig 38 shows the top view of the Sorting flap.



Figure 38: Figure showing top view of sorting flap using servo motor

It is driven by a servo motor. Its job is to separate out the odd fruit from the set of available fruits. When the odd fruit comes within range of the proximity sensor, which is placed near the sorting flap, the servo gear is attached to the flap using a servo link which helps in opening and closing of flap.

6.3.3 Conveyor level 2

Conveyor level 2 is used for grading purposes. It is made of aluminum extrusions and a polyvinyl chloride belt. It also has two rollers which enable the belt to move. Fig 39 shows the top view of conveyor level 2

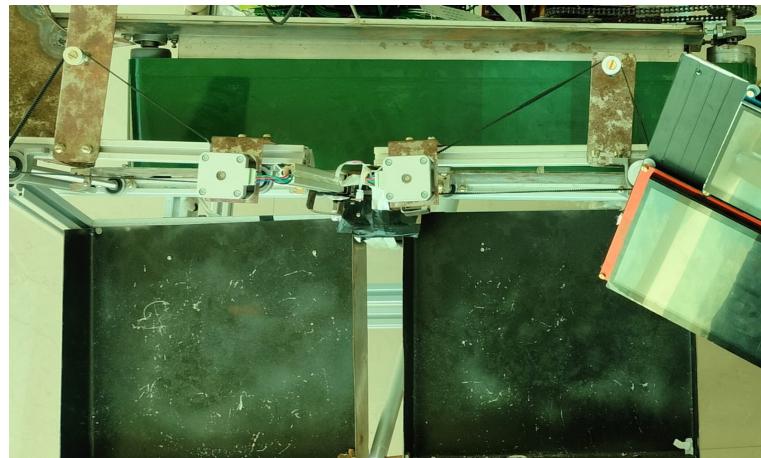


Figure 39: Figure showing top view of Conveyor 2 with the peripherals attached to it

The roller is connected mechanically to the NEMA-23 stepper motor using a gear-chain mechanism. Conveyor level 2 provides support to two grading gates that grade the fruit based on their size. Fig 40 shows the lateral view of the gear-chain mechanism of conveyor 2.

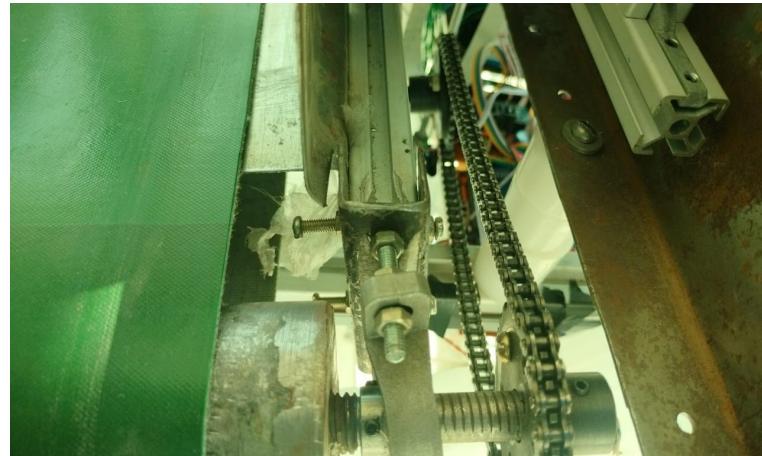


Figure 40: Figure showing a lateral view of the gear-chain mechanism of conveyor 2

Two collector trays are also attached to the conveyor level 2. Furthermore, one end of the roller is connected with an adjustable screw, which helps in adjusting the tension of the Conveyor belt

6.3.4 Curved Ramp

It is made up of 18 gauge MS. Its role is to provide a path between Conveyor level 1 and Conveyor level 2, The Semi-circular design of the curved ramp helps the fruit to roll over it without providing any damage to the fruit. Fig 41 shows the side view of the curved ramp



Figure 41: Figure showing a side view of curved ramp

6.4 Grading Gates

Grading gates are for the purpose of grading the fruit according to their size. Fig 42 shows the front view of grading gates.



Figure 42: Figure showing a the front view of grading gates

Moreover, the grading gates are adjusted automatically whenever a particular type of fruit is selected which helps in grading fruit in an efficient way.

6.4.1 Gate-1

This gate is made up of MS-18 gauge material. The purpose of Gate-1 is to grade a particular fruit of relatively small size. It has an extrusion at one end which helps the gate to attain its position whenever the extrusion comes in contact with the optical switch. Fig 43 shows the front view of Gate-1.



Figure 43: Figure showing a front view of Gate-1

A timing belt helps the gates to move up and down using the screw rod connected with the gate which is further connected to the NEMA-23 10kg-cm via a pulley. Furthermore, a steel plate is placed on the structure which allows for adjustment of the timing belt as needed.

6.4.2 Gate-2

This gate is also made up of MS-18 gauge material. The purpose of grading gate-2 is to allow the fruits through that could not pass through gate 1. This gate also has an optical switch as a limit switch that stops the gate's motion once the gate reaches the maximum vertical displacement. Fig 44 shows the front view of Gate-1.

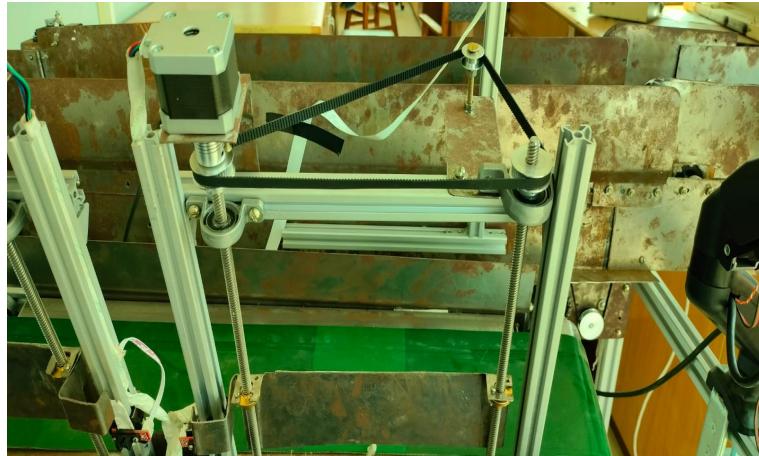


Figure 44: Figure showing a front view of Gate-2

A timing belt helps the gates to move up and down using the screw rod connected with the gate which is further connected to the NEMA-23 10kg-cm via pulley. Furthermore, a steel plate is placed on the structure which helps in the adjustment of the timing belt as needed.

6.5 Collector Tray

The tray collects the fruit that are graded by the grading gates, the first tray is in the series of Gate-1, and the second tray is in the series of Gate-2. Fig 45 shows the top view of the collector tray.

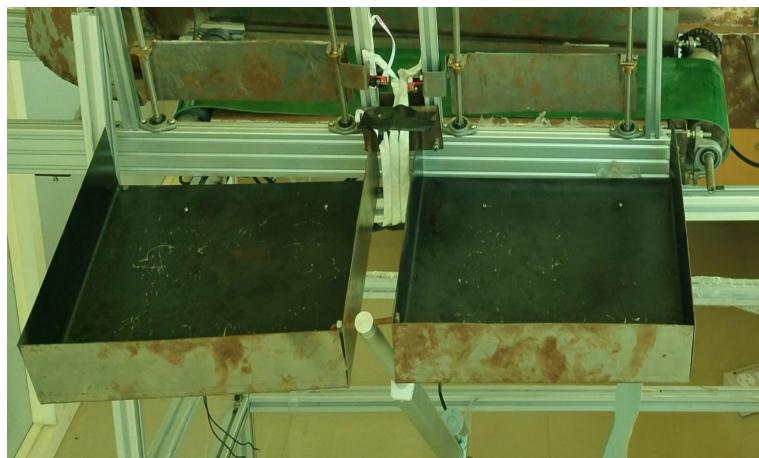


Figure 45: Figure showing a top view of collector tray at its extreme position

A linear actuator is attached at the bottom of the structure, and the other end is attached to the two collector trays that are joined together using a screw mechanism. Fig 46 how actuator is coupled with collector tray.



Figure 46: Figure showing how actuator is coupled with collector tray

Initially, the gates are at vertical position as soon as the VCC is applied the linear actuator gets turned on and the trays start moving upwards till they reach their extreme position, where the corner of the tray comes in contact with the optical switch. The height of the tray can be controlled manually using the Nextion display as mentioned earlier. Fig 47 how linear actuator is connected with the Aluminum structure.



Figure 47: Figure showing how linear actuator is connected with the Aluminum structure.

6.6 Gas Spring Monitor Arm

The Gas Spring Monitor Arm is used for the HMI display. Fig 48 shows the side view of the Gas Spring Monitor Arm



Figure 48: Figure showing a side view of the Gas Spring Monitor Arm

It allows us to move the display in 3 planes and adjust the height and position of the display as per our needs. Fig 49 shows the front view of the Gas Spring Monitor Arm



Figure 49: Figure showing the front view of the Gas Spring Monitor Arm

6.7 Aluminum Support

The whole conveyor system is placed on a supportive structure made of aluminum extrusions. The gas spring monitor arm and linear actuator, both are placed directly on this aluminum support. Fig 50 shows the front view of the aluminum structure.



Figure 50: Figure showing the structure supporting the overall system

All electrical circuitry, circuit boards, and control boards are placed on this support. This support also has 4 wheels that make the movement of the system easy and hence make the system portable. Fig 51 shows the lateral view of the wheel attached.



Figure 51: Figure showing the lateral view of the wheel attached with aluminum structure

7 RESULTS

In this project, an object detection model was developed using TensorFlow and a CNN architecture. The model exhibited excellent performance in accurately detecting and localizing fruits, achieving high levels of accuracy, precision, recall, and mean Average Precision (mAP) during evaluation. This demonstrated its effectiveness in identifying and classifying fruits based on their visual characteristics. All the models that were trained on the CNN and their accuracy is given in Table 2 below.

Dataset	Accuracy
Coco Dataset	96%
Fruit Detection and Recognition dataset by Kaggle	95%
Custom Dataset 1	87.23%
Custom Dataset 2	94.76%

Table 2: Accuracy percentage of different datasets.

Figure 52 shows the bar graph visualization of the different models. and their accuracy.

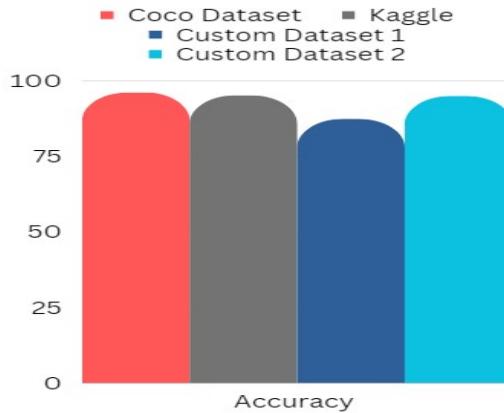


Figure 52: Bar graph of accuracy achieved.

These models after being deployed and tested on various single-board computers and FPGA devices showed promising results, they were able to recognize different classes of fruits in real-time. Raspberry Pi 4 8GB, NVIDIA Jetson Nano, and Pynq Z2 FPGA board were used to deploy these models. These platforms provided the necessary computational power to run the object detection model and the results that were obtained for each device are given in Table 3 below.

Device	Avg. Delay (sec)
Raspberry Pi 4	0.05
Jetson Nano	0.01
Pynq Z2	15

Table 3: Avg delay obtained on each device.

Figure 53 shows the bar graph visualization of avg. delay for each of the devices.

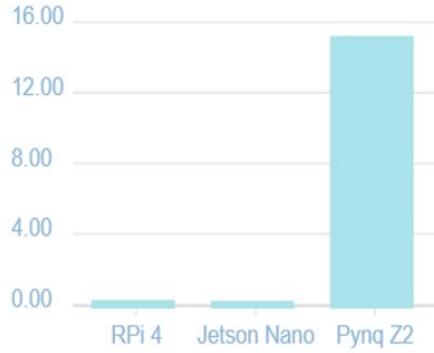


Figure 53: Bar graph of the speed of devices.

A lot of effort was put into creating a robust, low cost and modular structure for grading and sorting purposes. MS 18 gauge sheets were used to create the two-level conveyor belt structure and then a T-slot aluminum extrusion was used to create a stand for the structure whose height can be easily adjusted. This structure can be easily assembled and disassembled in a matter of seconds. Figure 54, 55, 56, 58 and 57 show different views of the conveyor structure .



Figure 54: Top view of grading & sorting structure.

The detailed comparison of model optimization techniques before and after optimization, including their respective accuracy values. Table 4 summarizes the results:

The detection output obtained when the datasets were deployed and executed on Raspberry Pi, Jetson Nano, pynq z2 showed the detection process in real-time, the bounding box was made for each detection made with the prediction percentage as shown in figures 59,60 and 61 respectively below:



Figure 55: Right Side view of grading & sorting structure.



Figure 56: Front view of grading & sorting structure.

Technique	COCO Dataset	Custom Dataset V1	Custom Dataset V2
Gradient Descent	Before: 85% After: 92%	Before: 92% After: 95%	Before: 93% After: 96%
Batch Normalization	Before: 88% After: 93%	Before: 93% After: 96%	Before: 92% After: 94%
Dropout	Before: 82% After: 91%	Before: 91% After: 94%	Before: 90% After: 93%
Weight Regularization	Before: 87% After: 93%	Before: 93% After: 95%	Before: 94% After: 97%
Learning Rate Schedule	Before: 84% After: 90%	Before: 90% After: 92%	Before: 91% After: 94%

Table 4: Model Optimization Results



Figure 57: Back view of grading & sorting structure.



Figure 58: Left side view of grading & sorting structure.

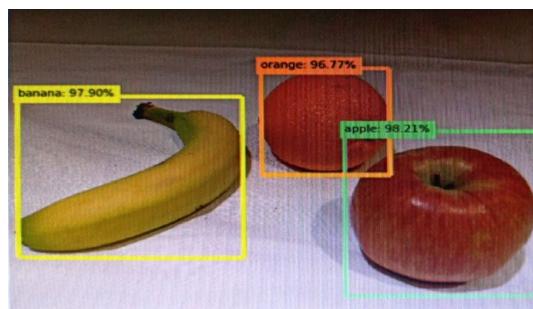


Figure 59: Output window with detection for Raspberry Pi.

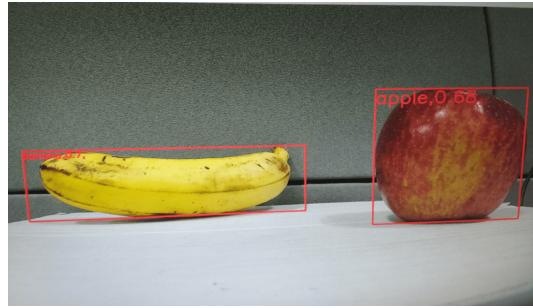


Figure 60: Output window with detection for JETSON Nano.

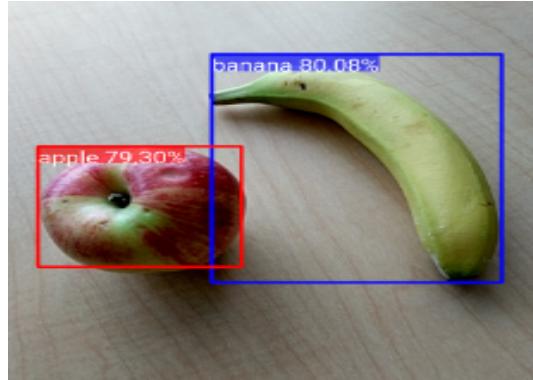


Figure 61: Output window with detection for PYNQ Z2.

8 CONCLUSION AND FUTURE WORK

Machine vision-based fruit grading and sorting systems are capable of replacing labor work for inspection of fruit grading. The system being a two-level conveyor system can help in achieving the modularity of the system which further promises to a wider applicability of the system. Furthermore, different devices were utilized like Raspberry Pi, PYNQ Z2, and NVIDIA Jetson Nano. However, by implementing these devices the result was observed that the NVIDIA Jetson nano gave the highest accuracy as by comparing the average delay of the different devices Raspberry Pi has 0.05 sec of delay, PYNQ Z2 has 15 sec of delay and Jetson nano has 0.01 sec of delay which is more preferable. The system is compact and robust it becomes a cost-effective system.

In conclusion, the proposed model is a modular, cost-effective, robust, and user-friendly fruit sorting and grading system

In view of the rapid technological advancements and wider applicability of the project, several areas for future work are identified to enhance the functionality

and performance of the whole system. for instance, the accuracy of the computer vision model of the proposed system can be improved by incorporating more advanced machine learning techniques and expanding the data set used for training. Also, we are considering a significant hardware optimization initiative. We aim to streamline the currently bulky hardware components, such as the Jetson NANO for image processing and detection, individual Printed Circuit Boards responsible for controlling various sensors and actuators, and the large stepper motor drivers. Our goal is to replace this assortment of hardware with a single FPGA (Field-Programmable Gate Array) device tailored to the specific requirements of our project.

Initially, we had planned to use the PYNQ-Z2 FPGA. However, we encountered some limitations, particularly in terms of the available I/O ports. As a result, we made the decision to transition to a more suitable FPGA board explicitly designed to handle computer vision processing. For this purpose, we have chosen the Kria AI Vision board. In the future, we intend to deploy versions of our project that are optimized for this hardware platform, ensuring enhanced speed and efficiency for our specific application. This optimization will not only reduce the overall hardware footprint but also improve the performance and capabilities of our system.

APPENDIX

1 JETSON NANO DATA-SHEET



DATA SHEET

NVIDIA Jetson Nano System-on-Module

Maxwell GPU + ARM Cortex-A57 + 4GB LPDDR4 + 16GB eMMC

Maxwell GPU [◊]

128-core GPU | End-to-end lossless compression | Tile Caching | OpenGL® 4.6 | OpenGL ES 3.2 | Vulkan™ 1.1 | CUDA® | OpenGL ES Shader Performance (up to): 512 GFLOPS (FP16)
Maximum Operating Frequency: 921MHz

CPU

ARM® Cortex® -A57 MPCore (Quad-Core) Processor with NEON Technology | L1 Cache: 48KB L1 instruction cache (I-cache) per core; 32KB L1 data cache (D-cache) per core | L2 Unified Cache: 2MB | Maximum Operating Frequency: 1.43GHz

Audio

Industry standard High Definition Audio (HDA) controller provides a multichannel audio path to the HDMI interface.

Memory

Dual Channel | System MMU | Memory Type: 4ch x 16-bit LPDDR4 | Maximum Memory Bus Frequency: 1600MHz | Peak Bandwidth: 25.6 GB/s | Memory Capacity: 4GB

Storage

eMMC 5.1 Flash Storage | Bus Width: 8-bit | Maximum Bus Frequency: 200MHz (HS400) | Storage Capacity: 16GB

Boot Sources

eMMC and USB (recovery mode)

Networking

10/100/1000 BASE-T Ethernet | Media Access Controller (MAC)

Imaging

Dedicated RAW to YUV processing engines process up to 1400Mpix/s (up to 24MP sensor) | MIPI CSI 2.0 up to 1.5Gbps (per lane) | Support for x4 and x2 configurations (up to four active streams).

Operating Requirements

Temperature Range (T_j): -25 – 97C* | Module Power: 5 – 10W | Power Input: 5.0V

Display Controller

Two independent display controllers support DSI, HDMI, DP, eDP:
MIPI-DSI (1.5Gbps/lane): Single x2 lane | Maximum Resolution: 1920x960 at 60Hz (up to 24bpp)
HDMI 2.0a/b (up to 6Gbps) | DP 1.2a (HBR2 5.4 Gbps) | eDP 1.4 (HBR2 5.4Gbps) | Maximum Resolution (DP/eDP/HDMI): 3840 x 2160 at 60Hz (up to 24bpp)

Clocks

System clock: 38.4MHz | Sleep clock: 32.768kHz | Dynamic clock scaling and clock source selection

Multi-Stream HD Video and JPEG

Video Decode

H.265 (Main, Main 10): 2160p 60fps | 1080p 240fps
H.264 (BP/MP/HP/Stereo SEI half-res): 2160p 60fps | 1080p 240fps
H.264 (MVC Stereo per view): 2160p 30fps | 1080p 120fps
VP9 (Profile 0, 8-bit): 2160p 60fps | 1080p 240fps
VP8: 2160p 60fps | 1080p 240fps
VC-1 (Simple, Main, Advanced): 1080p 120fps | 1080i 240fps
MPEG-2 (Main): 2160p 60fps | 1080p 240fps | 1080i 240fps

Video Encode

H.265: 2160p 30fps | 1080p 120fps
H.264 (BP/MP/HP): 2160p 30fps | 1080p 120fps
H.264 (MVC Stereo per view): 1440p 30fps | 1080p 60fps
VP8: 2160p 30fps | 1080p 120fps
JPEG (Decode and Encode): 600 MP/s

Peripheral Interfaces

xHCI host controller with integrated PHY: 1 x USB 3.0, 3 x USB 2.0 | USB 3.0 device controller with integrated PHY | EHCI controller with embedded hub for USB 2.0 | 4-lane PCIe: one x1/2/4 controller | single SD/MMC controller (supporting SDIO 4.0, SD HOST 4.0) | 3 x UART | 2 x SPI | 4 x I2C | 2 x I2S: support I2S, RJM, LJM, PCM, TDM (multi-slot mode) | GPIOs

Mechanical

Module Size: 69.6 mm x 45 mm | PCB: 8L HDI | Connector: 260 pin SO-DIMM

Note: Refer to the software release feature list for current software support; all features may not be available for a particular OS.

[◊] Product is based on a published Khronos Specification and is expected to pass the Khronos Conformance Process. Current conformance status can be found at www.khronos.org/conformance.

* See the *Jetson Nano Thermal Design Guide* for details. Listed temperature range is based on module T_j characterization.



Revision History

Version	Date	Description
v0.1	JAN 2019	Initial Release
v0.7	MAY 2019	<p>Description</p> <ul style="list-style-type: none">Memory: corrected peak bandwidthPeripheral Interfaces: corrected number of available I2C interfaces <p>Functional Overview</p> <ul style="list-style-type: none">Removed block diagram; see the <i>Jetson Nano Product Design Guide</i> for these details <p>Power and System Management</p> <ul style="list-style-type: none">Removed On-Module Internal Power Rails tableUpdated Power Domains tableUpdated Programmable Interface Wake Event tableUpdated Power Up/Down sequence diagrams <p>Pin Descriptions</p> <ul style="list-style-type: none">Updated throughout to reflect updated pinmuxGPIO Pins: updated table to reflect dedicated GPIO pins only (see pinmux for ALL GPIO capable pins) <p>Interface Descriptions</p> <ul style="list-style-type: none">Updated throughout to reflect updated pinmuxEmbedded DisplayPort (eDP) Interface: clarified DP use/limitations on DP0MIPi Camera Serial Interface (CSI) - Updated CSI description to remove erroneous reference to virtual channels <p>Physical/Electrical Characteristics</p> <ul style="list-style-type: none">Absolute Maximum Ratings - Added reference to Jetson Nano Thermal Design Guide for Operating Temperature; extended IDD_{MAX} to 5APinout: Updated to reflect updated pinmuxPackage Drawing and Dimensions – Updated drawing
v0.8	OCT 2019	<p>Description</p> <ul style="list-style-type: none">Operating Requirements: corrected Module Power to reflect power for module only (previous stated range included module + IO); updated Temperature Range for clarity, included maximum operating temperature and updated note to reflect module temperature is based on T_j.
v1.0	FEB 2020	<p>Pin Descriptions</p> <ul style="list-style-type: none">GPIO Pins: corrected pin number listing for GPIO01 <p>Interface Descriptions</p> <ul style="list-style-type: none">High-Definition Multimedia Interface (HDMI) and DisplayPort (DP) Interfaces reference to YUV output supportGigabit Ethernet – Corrected Realtek Gigabit Ethernet Controller part number <p>Physical/Electrical Characteristics</p> <ul style="list-style-type: none">Operating and Absolute Maximum Ratings – Added Mounting Force to Absolute Maximum Ratings table.Package Drawing and Dimensions – Updated drawingEnvironmental & Mechanical Screening – Added section
v1.1	MAY 2022	<p>Power and System Management</p> <ul style="list-style-type: none">Moved PMIC_BBAT information to new sub-sectionUpdated PMIC_BBAT Pin Description <p>Pin Descriptions</p> <ul style="list-style-type: none">GPIO Pins: updated GPIO8 (pin 208) description; Fan tachometer only <p>Interface Descriptions</p> <ul style="list-style-type: none">SD/SDIO – Updated pin descriptions to include 3.3V support; deprecated GPIO8 SD Card Detect supportSPI – updated master timing diagram and parametersUART – UART1_CTS (pin 209) updated PoR <p>Physical/Electrical Characteristics</p> <ul style="list-style-type: none">Package Drawing and Dimensions – added module dimensions table



NVIDIA.

Jetson Nano System-on-Module
Maxwell GPU + ARM Cortex-A57 + 4GB LPDDR4 + 16GB eMMC

Table of Contents

1.0 Functional Overview	5
1.1 Maxwell GPU	5
1.2 CPU Complex	6
1.2.1 Snoop Control Unit and L2 Cache	6
1.2.2 Performance Monitoring.....	7
1.3 High-Definition Audio-Video Subsystem	7
1.3.1 Multi-Standard Video Decoder.....	7
1.3.2 Multi-Standard Video Encoder.....	7
1.3.3 JPEG Processing Block	8
1.3.4 Video Image Compositor (VIC).....	8
1.4 Image Signal Processor (ISP)	9
1.5 Display Controller Complex	9
1.6 Memory.....	10
2.0 Power and System Management	11
2.1 Power Rails.....	12
2.2 PMIC_BBAT	12
2.3 Power Domains/Islands.....	12
2.4 Power Management Controller (PMC)	13
2.4.1 Resets	13
2.4.2 System Power States and Transitions	13
2.4.2.1 ON State	14
2.4.2.2 OFF State	14
2.4.2.3 SLEEP State.....	14
2.5 Thermal and Power Monitoring	15
2.6 Power Sequencing	15
2.6.1 Power Up	15
2.6.2 Power Down.....	15
3.0 Pin Descriptions	17
3.1 MPIO Power-on Reset Behavior	17
3.2 MPIO Deep Sleep Behavior	18
3.3 GPIO Pins.....	19
4.0 Interface Descriptions	20
4.1 USB	20
4.2 PCI Express (PCIe)	21
4.3 Display Interfaces	22
4.3.1 MIPI Display Serial Interface (DSI)	22
4.3.2 High-Definition Multimedia Interface (HDMI) and DisplayPort (DP) Interfaces	23
4.3.3 Embedded DisplayPort (eDP) Interface.....	24
4.4 MIPI Camera Serial Interface (CSI) / VI (Video Input).....	25
4.5 SD / SDIO	27
4.6 Inter-IC Sound (I ² S)	28
4.7 Miscellaneous Interfaces	29
4.7.1 Inter-Chip Communication (I ² C)	29
4.7.2 Serial Peripheral Interface (SPI)	29
4.7.3 UART	31
4.7.4 Gigabit Ethernet	32
4.7.5 Fan	33
4.7.6 Debug.....	33



NVIDIA.

Jetson Nano System-on-Module
Maxwell GPU + ARM Cortex-A57 + 4GB LPDDR4 + 16GB eMMC

5.0 Physical / Electrical Characteristics

34

5.1 Operating and Absolute Maximum Ratings	34
5.2 Digital Logic	35
5.3 Environmental & Mechanical Screening	35
5.4 Pinout	37
5.5 Package Drawing and Dimensions	38



NVIDIA.

Jetson Nano System-on-Module
Maxwell GPU + ARM Cortex-A57 + 4GB LPDDR4 + 16GB eMMC

1.0 Functional Overview

Designed for use in power-limited environments, the Jetson Nano squeezes industry-leading compute capabilities, 64-bit operating capability, and integrated advanced multi-function audio, video and image processing pipelines into a 260-pin SO-DIMM. The Maxwell GPU architecture implemented several architectural enhancements designed to extract maximum performance per watt consumed. Core components of the Jetson Nano series module include:

- NVIDIA® Tegra® X1 series SoC
 - NVIDIA Maxwell GPU
 - ARM® quad-core Cortex®-A57 CPU Complex
- 4GB LPDDR4 memory
- 16GB eMMC 5.1 storage
- Gigabit Ethernet (10/100/1000 Mbps)
- PMIC, regulators, power and voltage monitors
- 260-pin keyed connector (exposes both high-speed and low-speed industry standard I/O)
- On-chip temperature sensors

1.1 Maxwell GPU

The Graphics Processing Cluster (GPC) is a dedicated hardware block for rasterization, shading, texturing, and compute; most of the GPU's core graphics functions are performed inside the GPC. Within the GPC there are multiple Streaming Multiprocessor (SM) units and a Raster Engine. Each SM includes a Polymorph Engine and Texture Units; raster operations remain aligned with L2 cache slices and memory controllers

The Maxwell GPU architecture introduced an all-new design for the SM, redesigned all unit and crossbar structures, optimized data flows, and significantly improved power management. The SM scheduler architecture and algorithms were rewritten to be more intelligent and avoid unnecessary stalls, while further reducing the energy per instruction required for scheduling. The organization of the SM also changed; each Maxwell SM (called SMM) is now partitioned into four separate processing blocks, each with its own instruction buffer, scheduler and 32 CUDA cores.

The SMM CUDA cores perform pixel/vertex/geometry shading and physics/compute calculations. Texture units perform texture filtering and load/store units fetch and save data to memory. Special Function Units (SFUs) handle transcendental and graphics interpolation instructions. Finally, the Polymorph Engine handles vertex fetch, tessellation, viewport transform, attribute setup, and stream output. The SMM geometry and pixel processing performance make it highly suitable for rendering advanced user interfaces and complex gaming applications; the power efficiency of the Maxwell GPU enables this performance on devices with power-limited environments.

Features:

- End-to-end lossless compression
- Tile Caching
- Support for OpenGL 4.6, OpenGL ES 3.2, Vulkan 1.1, DirectX 12, CUDA 10 (FP16)
- Adaptive Scalable Texture Compression (ATSC) LDR profile supported
- Iterated blend, ROP OpenGL-ES blend modes
- 2D BLIT from 3D class avoids channel switch
- 2D color compression
- Constant color render SM bypass
- 2x, 4x, 8x MSAA with color and Z compression
- Non-power-of-2 and 3D textures, FP16 texture filtering



- FP16 shader support
- Geometry and Vertex attribute Instancing
- Parallel pixel processing
- Early-z reject: Fast rejection of occluded pixels acts as multiplier on pixel shader and texture performance while saving power and bandwidth
- Video protection region
- Power saving: Multiple levels of clock gating for linear scaling of power

GPU frequency and voltage are actively managed by Tegra Power and Thermal Management Software and influenced by workload. Frequency may be throttled at higher temperatures (above a specified threshold) resulting in a behavior that reduces the GPU operating frequency. Observed chip-to-chip variance is due to NVIDIA ability to maximize performance (DVFS) on a per-chip basis, within the available power budget.

1.2 CPU Complex

The CPU complex is a high-performance Multi-Core SMP cluster of four ARM Cortex-A57 CPUs with 2MB of L2 cache (shared by all cores). Features include:

- Superscalar, variable-length, out-of-order pipeline
- Dynamic branch prediction with Branch Target Buffer (BTB) and Global History Buffer RAMs, a return stack, and an indirect predictor
- 48-entry fully-associative L1 instruction TLB with native support for 4KB, 64KB, and 1MB page sizes.
- 32-entry fully-associative L1 data TLB with native support for 4KB, 64KB, and 1MB pages sizes.
- 4-way set-associative unified 1024-entry Level 2 (L2) TLB in each processor
- 48Kbyte I-cache and 32Kbyte D-cache for each core.
- Full implementation of ARMv8 architecture instruction set
- Embedded Trace Microcell (ETM) based on the ETMv4 architecture
- Performance Monitor Unit (PMU) based on the PMUv3 architecture
- Cross Trigger Interface (CTI) for multiprocessor debugging
- Cryptographic Engine for crypto function support
- Interface to an external Generic Interrupt Controller (vGIC-400)
- Power management with multiple power domains

CPU frequency and voltage are actively managed by Tegra Power and Thermal Management Software and influenced by workload. Frequency may be throttled at higher temperatures (above a specified threshold) resulting in a behavior that reduces the CPU operating frequency. Observed chip-to-chip variance is due to NVIDIA ability to maximize performance (DVFS) on a per-chip basis, within the available power budget.

1.2.1 Snoop Control Unit and L2 Cache

The CPU cluster includes an integrated snoop control unit (SCU) that maintains coherency between the CPUs within the cluster and a tightly coupled L2 cache that is shared between the CPUs within the cluster. The L2 cache also provides a 128-bit AXI master interface to access DRAM. L2 cache features include:

- 2MB L2
- Fixed line length of 64 bytes
- 16-way set-associative cache structure



- Duplicate copies of the L1 data cache directories for coherency support
- Hardware pre-fetch support
- ECC support

1.2.2 Performance Monitoring

The performance monitoring unit (part of MPCore non-CPU logic) provides six counters, each of which can count any of the events in the processor. The unit gathers various statistics on the operation of the processor and memory system during runtime, based on ARM PMUv3 architecture.

1.3 High-Definition Audio-Video Subsystem

The audio-video subsystem off-loads audio and video processing activities from the CPU subsystem resulting in faster, fully concurrent, highly efficient operation.

1.3.1 Multi-Standard Video Decoder

The video decoder accelerates video decode, supporting low resolution content, Standard Definition (SD), High Definition (HD) and UltraHD (2160p, or 4k video) profiles. The video decoder is designed to be extremely power efficient without sacrificing performance.

The video decoder communicates with the memory controller through the video DMA which supports a variety of memory format output options. For low power operations, the video decoder can operate at the lowest possible frequency while maintaining real-time decoding using dynamic frequency scaling techniques.

Video standards supported:

- H.265: Main10, Main
- WEBM VP9 and VP8
- H.264: Baseline (no FMO/ASO support), Main, High, Stereo SEI (half-res)
- VC-1: Simple, Main, Advanced
- MPEG-4: Simple (with B frames, interlaced; no DP and RVLC)
- H.263: Profile 0
- DiVX: 4/5/6
- XviD Home Theater
- MPEG-2: MP

1.3.2 Multi-Standard Video Encoder

The multi-standard video encoder enables full hardware acceleration of various encoding standards. It performs high-quality video encoding operations for applications such as video recording and video conferencing. The encode processor is designed to be extremely power-efficient without sacrificing performance.

Video standards supported:

- H.265 Main Profile: I-frames and P-frames (No B-frames)
- H.264 Baseline/Main/High Profiles: IDR/I/P/B-frame support, MVC
- VP8
- MPEG4 (ME only)
- MPEG2 (ME only)
- VC1 (ME only): No B frame, no interlaced



NVIDIA.

Jetson Nano System-on-Module
Maxwell GPU + ARM Cortex-A57 + 4GB LPDDR4 + 16GB eMMC

1.3.3 JPEG Processing Block

The JPEG processing block is responsible for JPEG (de)compression calculations (based on JPEG still image standard), image scaling, decoding (YUV420, YUV422H/V, YUV444, YUV400) and color space conversion (RGB to YUV; decode only).

Input (encode) formats:

- Pixel width: 8bpc
- Subsample format: YUV420
- Resolution up to 16K x 16K
- Pixel pack format
 - Semi-planar/planar for 420

Output (decode) formats:

- Pixel width 8bpc
- Resolution up to 16K x 16K
- Pixel pack format
 - Semi-planar/planar for YUV420
 - YUY2/planar for 422H/422V
 - Planar for YUV444
 - Interleave for RGBA

1.3.4 Video Image Compositor (VIC)

The Video Image Compositor implements various 2D image and video operations in a power-efficient manner. It handles various system UI scaling, blending and rotation operations, video post-processing functions needed during video playback, and advanced de-noising functions used for camera capture.

Features:

- Color Decompression
- High-quality Deinterlacing
- Inverse Telecine
- Temporal Noise Reduction
 - High-quality video playback
 - Reduces camera sensor noise
- Scaling
- Color Conversion
- Memory Format Conversion
- Blend/Composite
- 2D Bit BLIT operation
- Rotation



1.4 Image Signal Processor (ISP)

The ISP module takes data from the VI/CSI module or memory in raw Bayer format and processes it to YUV output. The imaging subsystem supports raw (Bayer) image sensors up to 24 million pixels. Advanced image processing is used to convert input to YUV data and remove artifacts introduced by high megapixel CMOS sensors and optics with up to 30-degree CRA.

Features:

- Flexible post-processing architecture for supporting custom computer vision and computational imaging operations
- Bayer domain hardware noise reduction
- Per-channel black-level compensation
- High-order lens-shading compensation
- 3 x 3 color transform
- Bad pixel correction
- Programmable coefficients for de-mosaic with color artifact reduction
 - Color Artifact Reduction: a two-level (horizontal and vertical) low-pass filtering scheme that is used to reduce/remove any color artifacts that may result from Bayer signal processing and the effects of sampling an image.
- Enhanced down scaling quality
- Edge Enhancement
- Color and gamma correction
- Programmable transfer function curve
- Color-space conversion (RGB to YUV)
- Image statistics gathering (per-channel)
 - Two 256-bin image histograms
 - Up to 4,096 local region averages
 - AC flicker detection (50Hz and 60Hz)
 - Focus metric block

1.5 Display Controller Complex

The Display Controller Complex integrates two independent display controllers. Each display controller is capable of interfacing to an external display device and can drive the same or different display contents at different resolutions and refresh rates. Each controller supports a cursor and three windows (Window A, B, and C); controller A supports two additional simple windows (Window D, T). The display controller reads rendered graphics or video frame buffers in memory, blends them and sends them to the display.

Features:

- Two heads. Each can be mapped to one of:
 - 1x DSI, 1x eDP/DP (Limited Functionality: No Audio)
 - 1x HDMI/DP (Full Functionality)
- 90, 180, 270-degree image transformation uses both horizontal and vertical flips (controller A only)
- Byte-swapping options on 16-bit and 32-bit boundary for all color depths
- NVIDIA Pixel Rendering Intensity and Saturation Management™ (PRISM)
- 256 x 256 cursor size
- Color Management Unit for color decompression and to enhance color accuracy (compensate for the color error specific to the display panel being used)



- Scaling and tiling in hardware for lower power operation
- Full color alpha-blending
- Captive panels
 - Secure window (Win T) for TrustZone
 - Supports cursor and up to four windows (Win A, B, C, and D)
 - 1x 2-lane MIPI DSI
 - Supports MIPI D-PHY rates up to 1.5Gbps
 - 4-lane eDP with AUX channel
 - Independent resolution and pixel clock
 - Supports display rotation and scaling in hardware
- External displays
 - Supports cursor and three windows (Window A, B, and C)
 - 1x HDMI (2.0) or DisplayPort (HBR2) interface
 - Supports display scaling in hardware

1.6 Memory

The Jetson Nano integrates 4GB of LPDDR4 over a four-channel x 16-bit interface. Memory frequency options are 204MHz and 1600MHz; maximum frequency of 1600MHz has a theoretical peak memory bandwidth of 25.6GB/s.

The Memory Controller (MC) maximizes memory utilization while providing minimum latency access for critical CPU requests. An arbiter is used to prioritize requests, optimizing memory access efficiency and utilization and minimizing system power consumption. The MC provides access to main memory for all internal devices. It provides an abstract view of memory to its clients via standardized interfaces, allowing the clients to ignore details of the memory hierarchy. It optimizes access to shared memory resources, balancing latency and efficiency to provide best system performance, based on programmable parameters.

Features:

- TrustZone (TZ) Secure and OS-protection regions
- System Memory Management Unit
- Dual CKE signals for dynamic power down per device
- Dynamic Entry/Exit from Self -Refresh and Power Down states

The MC can sustain high utilization over a very diverse mix of requests. For example, the MC is prioritized for bandwidth (BW) over latency for all multimedia blocks (the multimedia blocks have been architected to prefetch and pipeline their operations to increase latency tolerance); this enables the MC to optimize performance by coalescing, reordering, and grouping requests to minimize memory power. DRAM also has modes for saving power when it is either not being used, or during periods of specific types of use.



NVIDIA.

2.0 Power and System Management

The Jetson Nano module operates from a single power source (VDD_IN) with all internal module voltages and I/O voltages generated from this input. This enables the on-board power management controller to implement a tiered structure of power and clock gating in a complex environment that optimizes power consumption based on workload:

- **Power Management Controller (PMC) and Real Time Clock (RTC):** These blocks reside in an Always On (not power gated) partition. The PMC provides an interface to an external power manager IC or PMU. It primarily controls voltage transitions for the SoC as it transitions to/from different low power modes; it also acts as a slave receiving dedicated power/clock request signals as well as wake events from various sources (e.g., SPI, I2C, RTC, USB attach) which can wake the system from a deep-sleep state. The RTC maintains the ability to wake the system based on either a timer event or an external trigger (e.g., key press).
- **Power Gating:** The SoC aggressively employs power-gating (controlled by PMC) to power-off modules which are idle. CPU cores are on a separate power rail to allow complete removal of power and eliminate leakage. Each CPU can be power gated independently. Software provides context save/restore to/from DRAM.
- **Clock Gating:** Used to reduce dynamic power in a variety of power states.
- **Dynamic Voltage and Frequency Scaling (DVFS):** Raises voltages and clock frequencies when demand requires, lowers them when less is sufficient, and removes them when none is needed. DVFS is used to change the voltage and frequencies in the following power domains: CPU, CORE, and GPU.

Table 1 Power and System Control Pin Descriptions

Pin	Name	Direction	Type	PoR	Description
251 252 253 254 255 256 257 258 259 260	VDD_IN	Input	5.0V		Power: Main DC input, supplies PMIC and other regulators
235	PMIC_BBAT	Bidirectional	1.65V-5.5V		Power: PMIC Battery Back-up. Optionally used to provide back-up power for the Real-Time Clock (RTC).
240	SLEEP/WAKE*	Input	CMOS – 5.0V	PU	Sleep / Wake. Configured as GPIO for optional use to place system in sleep mode or wake system from sleep.
214	FORCE_RECOVERY*	Input	CMOS – 1.8V	PU	Force Recovery: strap pin
237	POWER_EN	Input	CMOS – 5.0V		Module on/off: high = on, low = off.
233	SHUTDOWN_REQ*	Output	CMOS – 5.0V	z	Shutdown Request: used by the module to request a shutdown from the carrier board (POWER_EN low). 100kΩ pull-up to VDD_IN (5V) on the module.
239	SYS_RESET*	Bidirectional	Open Drain, 1.8V	1	Module Reset. Reset to the module when driven low by the carrier board. When module power sequence is complete used as carrier board supply enable. Used to ensure proper power on/off sequencing between module and carrier board supplies. 4.7kΩ pull-up to 1.8V on the module.
178	MOD_SLEEP*	Output	CMOS – 1.8V		Indicates the module sleep status. Low is in sleep mode, high is normal operation. This pin is controlled by system software and should not be modified.



NVIDIA.

2.1 Power Rails

VDD_IN must be supplied by the carrier board that the Jetson Nano is designed to connect to. It must meet the required electrical specifications detailed in Section 5. All Jetson Nano interfaces are referenced to on-module voltage rails; no I/O voltage is required to be supplied to the module. See the *Jetson Nano Product Design Guide* for details of connecting to each of the interfaces.

2.2 PMIC_BBAT

An optional back up battery can be attached to the PMIC_BBAT module input. It is used to maintain the RTC voltage when VDD_IN is not present. This pin is connected directly to the onboard PMIC. When a backup cell is connected to the PMIC, the RTC will retain its contents and can be configured to charge the backup cell. RTC accuracy is 2 seconds/day under typical room temperature conditions (only).

The following backup cells may be attached to the PMIC_BBAT pin:

- Super Capacitor (gold cap, double layer electrolytic)
- Standard capacitors (tantalum)
- Rechargeable Lithium Manganese cells

A backup cell **MUST** provide a voltage in the range 2.5V to 3.5V. The backup cell is charged with a constant current, constant voltage charger that can be configured between 2.5V and 3.5V (constant voltage) output and 50 μ A to 800 μ A (constant current).

Table 2: PMIC_BBAT Pin Descriptions

Pin	Name	Description	Direction	Pin Type
235	PMIC_BBAT	PMIC Battery Back-up. Optionally used to provide back-up power for the Real-Time Clock (RTC). Connects to Lithium Cell or super capacitor on Carrier Board. PMIC is supply when charging cap or coin cell. Super cap or coin cell is source when system is disconnected from power. Constant current of 2.0 μ A for 2.5V; 2.3 μ A for 3.3V typical; 4.2 μ A maximum.	Bidir	1.65V-5.5V

2.3 Power Domains/Islands

Power domains and power islands are used to optimize power consumption for various low-power use cases and limiting leakage current. The RTC domain is always on, CORE/CPU/GPU domains can be turned on and off. The CPU, CORE and GPU power domains also contain power-gated islands which are used to power individual modules (as needed) within each domain. Clock-gating is additionally applied during powered-on but idle periods to further reduce unnecessary power consumption. Clock-gating can be applied to both power-gated and non-power-gated islands (NPG).

Table 3 Power Domains

Power Domain	Power Island in Domain	Modules in Power Island
RTC (VDD_RTC)	N/A	PMC (Power Management Controller)
		RTC (Real Time Clock)
CORE (VDD_SOC)	NPG (Non-Power-Gated)	AHB, APB Bus, AVP, Memory Controller (MC/EMC), USB 2.0, SDMMC
	VE, VE2	ISPs (image signal processing) A and B, VI (video input), CSI (Camera Serial Interface)
	NVENC	Video Encode
	NVDEC	Video Decode
	NVJPG	JPG accelerator and additional Video Decode
	PCX	PCIe

Power Domain	Power Island in Domain	Modules in Power Island
	SOR	HDMI, DSI, DP
	IRAM	IRAM
	DISP-A, DISP-B	Display Controllers A and B
	XUSBA, XUSBB, XUSBC	USB 3.0
	VIC	VIC (Video Image Compositor)
	ADSP	APE (Audio Processing Engine)
	DFD	Debug logic
GPU (VDD_GPU)	GPU	3D, FE, PD, PE, RAST, SM, ROP
CPU (VDD_CPU)	CPU 0	CPU 0
	CPU 1	CPU 1
	CPU 2	CPU 2
	CPU 3	CPU 3
	Non-CPU	L2 Cache for Main CPU complex
	TOP	Top level logic

2.4 Power Management Controller (PMC)

The PMC power management features enable both high-speed operation and very low-power standby states. The PMC primarily controls voltage transitions for the SoC as it transitions to/from different low-power modes; it also acts as a slave receiving dedicated power/clock request signals as well as wake events from various sources (e.g., SPI, I2C, RTC, USB attach) which can wake the system from deep sleep state. The PMC enables aggressive power-gating capabilities on idle modules and integrates specific logic to maintain defined states and control power domains during sleep and deep sleep modes.

2.4.1 Resets

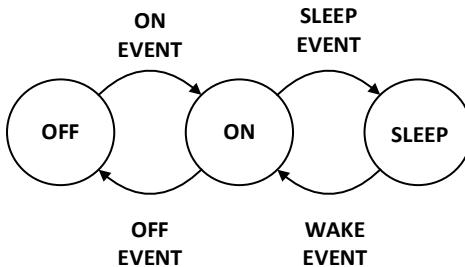
The PMC receives the primary reset event (from SYS_RESET*) and generates various resets for: PMC, RTC, and CAR. From the PMC provided reset, the Clock and Reset (CAR) controller generates resets for most of the blocks in the module. In addition to reset events, the PMC receives other events (e.g., thermal, WatchDog Timer (WDT), software, wake) which also result in variants of system reset.

The RTC block includes an embedded real-time clock and can wake the system based on either a timer event or an external trigger (e.g., key press).

2.4.2 System Power States and Transitions

The Jetson module operates in three main power modes: OFF, ON, and SLEEP. The module transitions between these states are based on various events from hardware or software. Figure 1 shows the transitions between these states.

Figure 1 Power State Diagram





2.4.2.1 ON State

The ON power state is entered from either OFF or SLEEP states. In this state the Jetson module is fully functional and operates normally. An ON event has to occur for a transition between OFF and ON states. The only ON EVENT currently used is a low to high transition on the POWER_EN pin. This must occur with VDD_IN connected to a power rail, and POWER_EN is asserted (at a logic1). The POWER_EN control is the carrier board indication to the Jetson module that the VDD_VIN power is good. The Carrier board should assert this high only when VDD_IN has reached its required voltage level and is stable. This prevents the Jetson module from powering up until the VDD_IN power is stable.

NOTE: The Jetson Nano module does include an Auto-Power-On option; a system input that enables the module to power on if asserted. For more information on available signals and broader system usage, see the *Jetson Nano Product Design Guide*.

2.4.2.2 OFF State

The OFF state is the default state when the system is not powered. It can only be entered from the ON state, through an OFF event. OFF Events are listed in the table below.

Table 4 OFF State Events

Event	Details	Preconditions
HW Shutdown	Set POWER_EN pin to zero for at least 100µS, the internal PMIC will start shutdown sequence	In ON State
SW Shutdown	Software initiated shutdown	ON state, Software operational
Thermal Shutdown	If the internal temperature of the Jetson module reaches an unsafe temperature, the hardware is designed to initiate a shutdown	Any power state

2.4.2.3 SLEEP State

The Sleep state can only be entered from the ON state. This state allows the Jetson module to quickly resume to an operational state without performing a full boot sequence. In this state the Jetson module operates in low power with enough circuitry powered to allow the device to resume and re-enter the ON state. During this state the output signals from Jetson module are maintained at their logic level prior to entering the state (i.e., they do not change to a 0V level).

The SLEEP state can only be entered directly by software. For example, operating within an OS, with no operations active for a certain time can trigger the OS to initiate a transition to the SLEEP state.

To Exit the SLEEP state a WAKE event must occur. WAKE events can occur from within the Jetson module or from external devices through various pins on the Jetson Nano connector. A full list of Wake enabled pins is available in the pinmux.

Table 5 SLEEP State Events

Event	Details
RTC WAKE up	Timers within the Jetson module can be programmed, on SLEEP entry. When these expire they create a WAKE event to exit the SLEEP state.
Thermal Condition	If the Jetson module internal temperature exceeds programmed hot and cold limits the system is forced to wake up, so it can report and take appropriate action (shut down for example)
USB VBUS detection	If VBUS is applied to the system (USB cable attached) then the device can be configured to Wake and enumerate

2.5 Thermal and Power Monitoring

The Jetson Nano is designed to operate under various workloads and environmental conditions. It has been designed so that an active or passive heat sinking solution can be attached. The module contains various methods through hardware and software to limit the internal temperature to within operating limits. See the *Jetson Nano Thermal Design Guide* for more details.

2.6 Power Sequencing

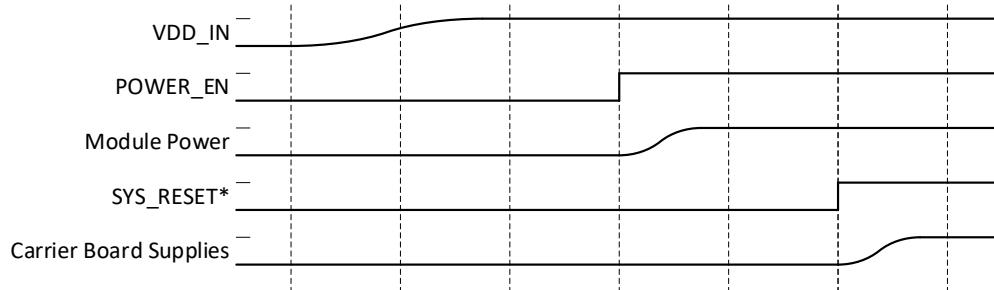
The Jetson Nano module is required to be powered on and off in a known sequence. Sequencing is determined through a set of control signals; the SYS_RESET* signal (when deasserted) is used to indicate when the carrier board can power on. The following sections provide an overview of the power sequencing steps between the carrier board and Jetson Nano module. Refer to the *Jetson Nano Product Design Guide* for system level details on the application of power, power sequencing, and monitoring. The Jetson Nano module and the product carrier board must be power sequenced properly to avoid potential damage to components on either the module or the carrier board system.

2.6.1 Power Up

During power up, the carrier board must wait until the signal SYS_RESET* is deasserted from the Jetson module before enabling its power; the Jetson module will deassert the SYS_RESET* signal to enable the complete system to boot.

NOTE: I/O pins cannot be high (>0.5V) before SYS_RESET* goes high. When SYS_RESET* is low, the maximum voltage applied to any I/O pin is 0.5V. For more information, refer to the *Jetson Nano Product Design Guide*.

Figure 2 Power-up Sequence (No Power Button – Auto-Power-On Enabled)



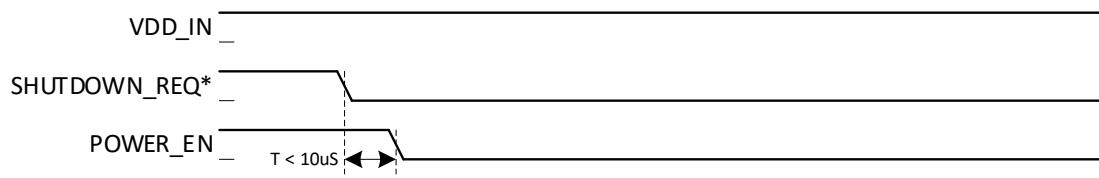
2.6.2 Power Down

In a shutdown event the Jetson module asserts SHUTDOWN_REQ*. The SHUTDOWN_REQ* must be serviced by the carrier board to toggle POWER_EN from high to low, even in cases of sudden power loss. The Jetson module starts the power off sequence when POWER_EN is deasserted; SYS_RESET* is asserted by the Jetson module, allowing the carrier board to put any components into a known state and power down.



Jetson Nano System-on-Module
Maxwell GPU + ARM Cortex-A57 + 4GB LPDDR4 + 16GB eMMC

Figure 3 Power Down Sequence (Initiated by SHUTDOWN_REQ* Assertion)





3.0 Pin Descriptions

The primary interface to Jetson Nano is via a 260-pin SO-DIMM connector. Connector exposes power, ground, high-speed and low-speed industry standard I/O connections. See the *NVIDIA Jetson Nano Product Design Guide* for details on integrating the module and mating connector into product designs.

The I/O pins on the SO-DIMM are comprised of both Single Function I/O (SFIO) and Multi-Purpose digital I/O (MPIO) pins. Each MPIO can be configured to act as a GPIO or it can be assigned for use by a particular I/O controller. Though each MPIO has up to five functions (GPIO function and up to four SFIO functions), a given MPIO can only act as a single function at a given point in time. The functions for each pin on the Jetson module are fixed to a single SFIO function or as a GPIO. The different MPIO pins share a similar structure, but there are several varieties of such pins. The varieties are designed to minimize the number of on-board components (such as level shifters or pull-up resistors) required in Jetson Nano designs.

MPIO pin types:

- ST (standard) pins are the most common pins on the chip. They are used for typical General Purpose I/O.
- DD (dual-driver) pins are similar to the ST pins. A DD pin can tolerate its I/O pin being pulled up to 3.3V (regardless of supply voltage) if the pin's output-driver is set to open-drain mode. There are special power-sequencing considerations when using this functionality.

NOTE: The output of DD pins cannot be pulled High during deep-power-down (DPD).

- CZ (controlled output impedance) pins are optimized for use in applications requiring tightly controlled output impedance. They are similar to ST pins except for changes in the drive strength circuitry and in the weak pull-ups/downs. CZ pins are included on the VDDIO_SDMMC3 (Module SDMMC pins) power rail; also includes a CZ_COMP pin. Circuitry within the Jetson module continually matches the output impedance of the CZ pins to the on-board pull-up/-down resistors attached to the CZ_COMP pins.
- LV_CZ (low voltage-controlled impedance) pins are similar to CZ pins but are optimized for use with a 1.2V supply voltage (and signaling level). They support a 1.8V supply voltage (and signaling level) as a secondary mode. The Jetson nano uses LV_CZ pins for SPI interfaces operating at 1.8V.
- DP_AUX pin is used as an Auxiliary control channel for the DisplayPort which needs differential signaling. Because the same I/O block is used for DisplayPort and HDMI to ensure the control path to the display interface is minimized, the DP_AUX pins can operate in open-drain mode so that HDMI's control path (i.e., DDC interface which needs I2C) can also be used in the same pin.

Each MPIO pin consists of:

- An output driver with tristate capability, drive strength controls and push-pull mode, open-drain mode, or both
- An input receiver with either Schmitt mode, CMOS mode, or both
- A weak pull-up and a weak pull-down

MPIO pins are partitioned into multiple “pin control groups” with controls being configured for the group. During normal operation, these per-pin controls are driven by the pinmux controller registers. During deep sleep, the PMC bypasses and then resets the pinmux controller registers. Software reprograms these registers as necessary after returning from deep sleep.

Refer to the *Tegra X1 (SoC) Technical Reference Manual* for more information on modifying pin controls.

3.1 MPIO Power-on Reset Behavior

Each MPIO pin has a deterministic power-on reset (PoR) state. The particular reset state for each pin is chosen to minimize the need of on-board components like pull-up resistors in a Jetson Nano-based system. For example, the on-chip weak pull-ups are enabled during PoR for pins which are usually used to drive active-low chip selects.

3.2 MPIO Deep Sleep Behavior

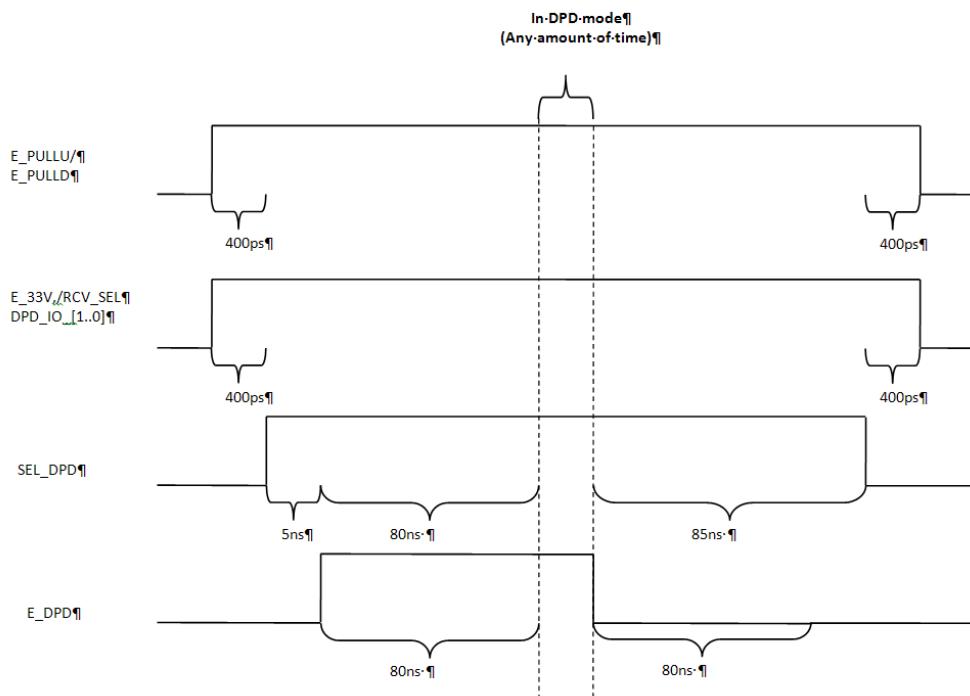
Deep Sleep is an ultra-low-power standby state in which the Jetson Nano maintains much of its I/O state while most of the chip is powered off. The following lists offer a simplified description of the deep sleep entry and exit concentrating on those aspects which relate to the MPIO pins. During deep sleep most of the pins are put in a state called Deep Power Down (DPD). The sequence for entering to DPD is same across pins. Specific variations are there in some pins in terms of type of features that are available in DPD.

NOTE: The output of DD pins cannot be pulled High during deep-power-down (DPD).
 OD pins do NOT retain their output during DPD. OD pins should NOT be configured as GPIOs in a platform where they are expected to hold a value during DPD.

ALL MPIO pins do NOT have identical behavior during deep sleep. They differ with regard to:

- Input buffer behavior during deep sleep
 - Forcibly disabled OR
 - Enabled for use as a “GPIO wake event” OR
 - Enabled for some other purpose (e.g., a “clock request” pin)
- Output buffer behavior during deep sleep
 - Maintain a static programmable (0, 1, or tristate) constant value OR
 - Capable of changing state (i.e., dynamic while the chip is still in deep sleep)
- Weak pull-up/pull-down behavior during deep sleep
 - Forcibly disabled OR
 - Can be configured
- Pins that do not enter deep sleep
 - Some of the pins whose outputs are dynamic during deep sleep are of special type and they do not enter deep sleep (e.g., pins that are associated with PMC logic do not enter deep sleep, pins that are associated with JTAG do not enter into deep sleep any time).

Figure 4 DPD Wait Times





3.3 GPIO Pins

The Jetson Nano has multiple dedicated GPIOs. Each GPIO can be individually configurable as an Output, Input, or Interrupt source with level/edge controls. The pins listed in the following table are dedicated GPIOs; some with alternate SFIO functionality. Many other pins not included in this list are capable of being configured as GPIOs instead of the SFIO functionality the pin name suggests (e.g., UART, SPI, I²S, etc.). All pins that can support GPIO functionality have this exposed in the Pinmux.

Table 6 Dedicated GPIO Pin Descriptions

Pin	Name	Direction	Type	PoR	Alternate Function
87	GPIO00	Bidirectional	Open-Drain [DD]	0	USB VBUS Enable (USB_VBUS_EN0)
118	GPIO01	Bidirectional	CMOS – 1.8V [ST]	pd	Camera MCLK #2 (CLK)
124	GPIO02	Bidirectional	CMOS – 1.8V [ST]	pd	
126	GPIO03	Bidirectional	CMOS – 1.8V [ST]	pd	
127	GPIO04	Bidirectional	CMOS – 1.8V [ST]	pd	
128	GPIO05	Bidirectional	CMOS – 1.8V [ST]	pd	
130	GPIO06	Bidirectional	CMOS – 1.8V [ST]	pd	
206	GPIO07	Bidirectional	CMOS – 1.8V [ST]	pd	Pulse Width Modulation Signal (PWM)
208	GPIO08	Bidirectional	CMOS – 1.8V [ST]	pd	Fan Tachometer
211	GPIO09	Bidirectional	CMOS – 1.8V [ST]	pd	Audio Clock (AUD_MCLK)
212	GPIO10	Bidirectional	CMOS – 1.8V [ST]	pd	
216	GPIO11	Bidirectional	CMOS – 1.8V [ST]	pd	Camera MCLK #3
218	GPIO12	Bidirectional	CMOS – 1.8V [ST]	pd	
228	GPIO13	Bidirectional	CMOS – 1.8V [ST]	pd	Pulse Width Modulation Signal
230	GPIO14	Bidirectional	CMOS – 1.8V [ST]	pd	Pulse Width Modulation Signal
114	CAM0_PWDN	Bidirectional	CMOS – 1.8V [ST]	pd	
120	CAM1_PWDN	Bidirectional	CMOS – 1.8V [ST]	pd	

4.0 Interface Descriptions

The following sections outline the interfaces available on the Jetson Nano module and details the module pins used to interact with and control each interface. See the *Tegra X1 Series SoC Technical Reference Manual* for complete functional descriptions, programming guidelines and register listings for each of these blocks.

4.1 USB

Standard	Notes
<i>Universal Serial Bus Specification Revision 3.0</i>	Refer to specification for related interface timing details.
<i>Universal Serial Bus Specification Revision 2.0</i>	USB Battery Charging Specification, version 1.0; including Data Contact Detect protocol Modes: Host and Device Speeds: Low, Full, and High Refer to specification for related interface timing details.
<i>Enhanced Host Controller Interface Specification for Universal Serial Bus revision 1.0</i>	Refer to specification for related interface timing details.

An xHCI/Device controller (named XUSB) supports the xHCI programming model for scheduling transactions and interface managements as a host that natively supports USB 3.0, USB 2.0, and USB 1.1 transactions with its USB 3.0 and USB 2.0 interfaces. The XUSB controller supports USB 2.0 L1 and L2 (suspend) link power management and USB 3.0 U1, U2, and U3 (suspend) link power managements. The XUSB controller supports remote wakeup, wake on connect, wake on disconnect, and wake on overcurrent in all power states, including sleep mode.

USB 2.0 Ports

Each USB 2.0 port operates in USB 2.0 High Speed mode when connecting directly to a USB 2.0 peripheral and operates in USB 1.1 Full- and Low-Speed modes when connecting directly to a USB 1.1 peripheral. All USB 2.0 ports operating in High Speed mode share one High-Speed Bus Instance, which means 480 Mb/s theoretical bandwidth is distributed across these ports. All USB 2.0 ports operating in Full- or Low-Speed modes share one Full/Low-Speed Bus Instance, which means 12 Mb/s theoretical bandwidth is distributed across these ports.

USB 3.0 Port

The USB 3.0 port only operates in USB 3.0 Super Speed mode (5 Gb/s theoretical bandwidth).

Table 7 USB 2.0 Pin Descriptions

Pin	Name	Direction	Type	Description
87	GPIO0	Input	USB VBUS, 5V	USB 0 VBUS Detect (USB_VBUS_EN0). Do not feed 5V directly into this pin; see the <i>Jetson Nano Product Design Guide</i> for complete details.
109 111	USB0_D_N USB0_D_P	Bidirectional	USB PHY	USB 2.0 Port 0 Data
115 117	USB1_D_N USB1_D_P	Bidirectional	USB PHY	USB 2.0 Port 1 Data
121 123	USB2_D_N USB2_D_P	Bidirectional	USB PHY	USB 2.0 Port 2 Data

Table 8 USB 3.0 Pin Descriptions

Pin	Name	Direction	Type	Description
163 161	USBSS_RX_P USBSS_RX_N	Input	USB SS PHY	USB 3.0 SS Receive



NVIDIA.

Jetson Nano System-on-Module
Maxwell GPU + ARM Cortex-A57 + 4GB LPDDR4 + 16GB eMMC

Pin	Name	Direction	Type	Description
168	USBSS_TX_P	Output	USB SS PHY	USB 3.0 SS Transmit
166	USBSS_TX_N			

4.2 PCI Express (PCIe)

Standard	Notes
<i>PCI Express Base Specification Revision 2.0</i>	Jetson Nano meets the timing requirements for the Gen2 (5.0 GT/s) data rates. Refer to specification for complete interface timing details. Although NVIDIA validates that the Jetson Nano design complies with the PCIe specification, PCIe software support may be limited.

The Jetson module integrates a single PCIe Gen2 controller supporting:

- Connections to a single (x1/2/4) endpoint
- Upstream and downstream AXI interfaces that serve as the control path from the Jetson Nano to the external PCIe device.
- Gen1 (2.5 GT/s/lane) and Gen2 (5.0 GT/s/lane) speeds.

NOTE: Upstream Type 1 Vendor Defined Messages (VDM) should be sent by the Endpoint Port (EP) if the Root Port (RP) also belongs to same vendor/partner; otherwise the VDM is silently discarded.

See the *Jetson Nano Product Design Guide* for supported USB 3.0/PCIe configuration and connection examples.

Table 9 PCIe Pin Descriptions

Pin	Name	Direction	Type	PoR	Description
179	PCIE_WAKE*	Input	Open Drain 3.3V	z	PCI Express Wake This signal is used as the PCI Express defined WAKE# signal. When asserted by a PCI Express device, it is a request that system power be restored. No interrupt or other consequences result from the assertion of this signal. On module 100kΩ pull-up to 3.3V
160	PCIE0_CLK_N	Output	PCIe PHY	0	PCIe Reference Clock
162	PCIE0_CLK_P			0	
180	PCIE0_CLKREQ*	Bidirectional	Open Drain 3.3V	z	PCIe Reference Clock Request This signal is used by a PCIe device to indicate it needs the PCIE0_CLK_N and PCIE0_CLK_P to actively drive reference clock. On module 47kΩ pull-up to 3.3V
181	PCIE0_RST*	Output	Open Drain 3.3V	0	PCIe Reset This signal provides a reset signal to all PCIe links. It must be asserted 100 ms after the power to the PCIe slots has stabilized. On module 47kΩ pull-up to 3.3V
157	PCIE0_RX3_P	Input	PCIe PHY		PCIe Receive (Lane 3)
155	PCIE0_RX3_N				
151	PCIE0_RX2_P	Input	PCIe PHY		PCIe Receive (Lane 2)
149	PCIE0_RX2_N				
139	PCIE0_RX1_P	Input	PCIe PHY		PCIe Receive (Lane 1)
137	PCIE0_RX1_N				
133	PCIE0_RX0_P	Input	PCIe PHY		PCIe Receive (Lane 0)
131	PCIE0_RX0_N				
156	PCIE0_TX3_P	Output	PCIe PHY		PCIe Transmit (Lane 3)
154	PCIE0_TX3_N				



NVIDIA.

Pin	Name	Direction	Type	PoR	Description
150	PCIE0_TX2_P	Output	PCIe PHY		PCIe Transmit (Lane 2)
148	PCIE0_TX2_N				
142	PCIE0_TX1_P	Output	PCIe PHY		PCIe Transmit (Lane 1)
140	PCIE0_TX1_N				
136	PCIE0_TX0_P	Output	PCIe PHY		PCIe Transmit (Lane 0)
134	PCIE0_TX0_N				

4.3 Display Interfaces

The Jetson Nano Display Controller Complex integrates a MIPI-DSI interface and Serial Output Resource (SOR) to collect pixels from the output of the display pipeline, format/encode them to desired format, and then streams to various output devices. The SOR consists of several individual resources which can be used to interface with different display devices such as HDMI, DP, or eDP.

4.3.1 MIPI Display Serial Interface (DSI)

The Display Serial Interface (DSI) is a serial bit-stream replacement for the parallel MIPI DPI and DBI display interface standards. DSI reduces package pin-count and I/O power consumption. DSI support enables both display controllers to connect to an external display(s) with a MIPI DSI receiver. The DSI transfers pixel data from the internal display controller to an external third-party LCD module.

Features:

- PHY Layer
 - Start / End of Transmission. Other out-of-band signaling
 - Per DSI interface: one Clock Lane; two Data Lanes
 - Supports link configuration – 1x 2
 - Maximum link rate 1.5Gbps as per MIPI D-PHY 1.1v version
 - Maximum 10MHz LP receive rate
- Lane Management Layer with Distributor
- Protocol Layer with Packet Constructor
- Supports MIPI DSI 1.0.1v version mandatory features
- Command Mode (One-shot) with Host and/or display controller as master
- Clocks
 - Bit Clock: Serial data stream bit-rate clock
 - Byte Clock: Lane Management Layer Byte-rate clock
 - Application Clock: Protocol Layer Byte-rate clock.
- Error Detection / Correction
 - ECC generation for packet Headers
 - Checksum generation for Long Packets
- Error recovery
- High-Speed Transmit timer
- Low-Power Receive timer
- Turnaround Acknowledge Timeout



Table 10 DSI Pin Descriptions

Pin	Name	Direction	Type	Description
76	DSI_CLK_N	Output	MIPI D-PHY	Differential output clock for DSI interface
78	DSI_CLK_P			
82	DSI_D1_N	Output	MIPI D-PHY	Differential data lanes for DSI interface.
84	DSI_D1_P			
70	DSI_D0_N	Bidirectional	MIPI D-PHY	Differential data lanes for DSI interface. DSI lane can read data back from the panel side in low power (LP) mode.
72	DSI_D0_P			

4.3.2 High-Definition Multimedia Interface (HDMI) and DisplayPort (DP) Interfaces

Standard	Notes
<i>High-Definition Multimedia Interface (HDMI) Specification, version 2.0</i>	> 340MHz pixel clock Scrambling support Clock/4 support (1/40 bit-rate clock)

The HDMI and DP interfaces share the same set of interface pins. A new transport mode was introduced in HDMI 2.0 to enable link clock frequencies greater than 340MHz and up to 600MHz. For transfer rates above 340MHz, there are two main requirements:

- All link data, including active pixel data, guard bands, data islands and control islands must be scrambled.
- The TMDS clock lane must toggle at CLK/4 instead of CLK. Below 340MHz, the clock lane toggles as normal (independent of the state of scrambling).

Features:

- HDMI
 - HDMI 2.0 mode (3.4Gbps < data rate <= 6Gbps)
 - HDMI 1.4 mode (data rate<=3.4Gbps)
 - Multi-channel audio from HDA controller, up to eight channels 192kHz 24-bit.
 - Vendor Specific Info-frame (VSI) packet transmission
 - 24-bit RGB pixel formats
 - Transition Minimized Differential Signaling (TMDS) functional up to 340MHz pixel clock rate
- DisplayPort
 - Display Port mode: interface is functional up to 540MHz pixel clock rate (i.e., 1.62GHz for RBR, 2.7GHz for HBR, and 5.4GHz for HBR2).
 - 8b/10b encoding support
 - External Dual Mode standard support
 - Audio streaming support

Table 11 HDMI Pin Descriptions

Pin	Name	Direction	Type	Description
83	DP1_TXD3_P	Differential Output	AC-Coupled on Carrier Board [DP]	DP Data lane 3 or HDMI Differential Clock. AC coupling required on carrier board. For HDMI, pull-downs (with disable) also required on carrier board.
81	DP1_TXD3_N			
77	DP1_TXD2_P	Differential Output	AC-Coupled on Carrier Board [DP]	HDMI Differential Data lanes 2:0. AC coupling required on carrier board. For HDMI, pull-downs (with disable) also required on carrier board.
75	DP1_TXD2_N			
71	DP1_TXD1_P			HDMI: DP1_TXD2_[P,N] = HDMI Lane 0
69	DP1_TXD1_N			DP1_TXD1_[P,N] = HDMI Lane 1
65	DP1_TXD0_P			



NVIDIA.

Pin	Name	Direction	Type	Description
63	DP1_TXD0_N			DP1_TXD0_[P,N] = HDMI Lane 2
96	DP1_HPD	Input	CMOS – 1.8V [ST]	HDMI Hot Plug detection. Level shifter required as this pin is not 5V tolerant.
94	HDMI_CEC	Bidirectional	Open Drain, 1.8V [DD]	Consumer Electronics Control (CEC) one-wire serial bus. NVIDIA provides low level CEC APIs (read/write). These are not supported in earlier Android releases. For additional CEC support, 3rd party libraries need to be made available.
100	DP1_AUX_P	Bidirectional	Open-Drain, 1.8V (3.3V tolerant - DDC) [DP_AUX]	DDC Serial Clock for HDMI. Level shifter required; pin is not 5V tolerant.
98	DP1_AUX_N	Bidirectional	Open-Drain, 1.8V (3.3V tolerant - DDC)	DDC Serial Data. Level shifter required; pin is not 5V tolerant.

Table 12 DisplayPort on DP1 Pin Descriptions

Pin	Name	Direction	Type	Description
83	DP1_TXD3_P			DisplayPort 1 Differential Data lanes 2:0. AC coupling required on carrier board.
81	DP1_TXD3_N			DP1_TXD2_[P,N] = DP Lane 2
77	DP1_TXD2_P			DP1_TXD1_[P,N] = DP Lane 1
75	DP1_TXD2_N			DP1_TXD0_[P,N] = DP Lane 0
71	DP1_TXD1_P			
69	DP1_TXD1_N			
65	DP1_TXD0_P			
63	DP1_TXD0_N			
96	DP1_HPD	Input	CMOS – 1.8V [ST]	DisplayPort 1 Hot Plug detection. Level shifter required and must be non-inverting.
100	DP1_AUX_P	Bidirectional	Open-Drain, 1.8V [DP_AUX]	DisplayPort 1 auxiliary channels. AC coupling required on carrier board.
98	DP1_AUX_N			

4.3.3 Embedded DisplayPort (eDP) Interface

Standard	Notes
Embedded DisplayPort 1.4	Supported eDP 1.4 features: <ul style="list-style-type: none"> ▪ Additional link rates ▪ Enhanced framing ▪ Power sequencing ▪ Reduced aux timing ▪ Reduced main voltage swing

eDP is a mixed-signal interface consisting of four differential serial output lanes and one PLL. This PLL is used to generate a high frequency bit-clock from an input pixel clock enabling the ability to handle 10-bit parallel data per lane at the pixel rate for the desired mode. Embedded DisplayPort (eDP) modes (1.6GHz for RBR, 2.16GHz, 2.43GHz, 2.7GHz for HBR, 3.42GHz, 4.32GHz and 5.4GHz for HBR2).

NOTE: eDP has been tested according to DP1.2b PHY CTS even though eDPv1.4 supports lower swing voltages and additional intermediate bit rates. This means the following nominal voltage levels (400mV, 600mV, 800mV, 1200mV) and data rates (RBR, HBR, HBR2) are tested. This interface can be tuned to drive lower voltage swings below 400mV and can be programmed to other intermediate bit rates as per the requirements of the panel and the system designer.

DisplayPort on DP0 is limited to display functionality only; no HDCP or audio support.



NVIDIA.

Jetson Nano System-on-Module
Maxwell GPU + ARM Cortex-A57 + 4GB LPDDR4 + 16GB eMMC

Table 13 eDP (or DisplayPort on DP0) Pin Descriptions

Pin	Name	Direction	Type	Description
59	DP0_TXD3_P	Differential Output	AC-Coupled on Carrier Board [DP]	DP0 Differential Data. AC coupling & pull-downs (with disable) required on carrier board. DP0_TXD3_[P,N] = DisplayPort 0 Data Lane 3
57	DP0_TXD3_N			DP0_TXD2_[P,N] = DisplayPort 0 Data Lane 2
53	DP0_TXD2_P			DP0_TXD1_[P,N] = DisplayPort 0 Data Lane 1
51	DP0_TXD2_N			DP0_TXD0_[P,N] = DisplayPort 0 Data Lane 0
47	DP0_TXD1_P			
45	DP0_TXD1_N			
41	DP0_TXD0_P			
39	DP0_TXD0_N			
88	DP0_HPD	Input	CMOS – 1.8V [ST]	DP0 Hot Plug detection. Level shifter required as this pin is not 5V tolerant
92	DP0_AUX_P	Bidirectional	AC-Coupled on Carrier Board [DP_AUX]	DP0 auxiliary channels. AC coupling required on Carrier board.
90	DP0_AUX_N			

4.4 MIPI Camera Serial Interface (CSI) / VI (Video Input)

Standard
MIPI CSI 2.0 Receiver specification
MIPI D-PHY® v1.2 Physical Layer specification

The Camera Serial Interface (CSI) is based on MIPI CSI 2.0 standard specification and implements the CSI receiver which receives data from an external camera module with CSI transmitter. The Video Input (VI) block receives data from the CSI receiver and prepares it for presentation to system memory or the dedicated image signal processor (ISP) execution resources.

Features:

- Supports both x4-lane and x2-lane sensor camera configurations:
 - x4 only configuration (up to three active streams)
 - x4 + x2 configurations (up to four active streams)
- Supported input data formats:
 - RGB: RGB888, RGB666, RGB565, RGB555, RGB444
 - YUV: YUV422-8b, YUV420-8b (legacy), YUV420-8b, YUV444-8b
 - RAW: RAW6, RAW7, RAW8, RAW10, RAW12, RAW14
 - DPCM: user defined
 - User defined: JPEG8
 - Embedded: Embedded control information
- Supports single-shot mode
- Physical Interface (MIPI D-PHY) Modes of Operation
 - High Speed Mode – High-speed differential signaling up to 1.5Gbps; burst transmission for low power
 - Low Power Control – Single-ended 1.2V CMOS level; low-speed signaling for handshaking.
 - Low Power Escape – Low-speed signaling for data, used for escape command entry only.

If the two streams come from a single source, then the streams are separated using a filter indexed on different data types. In case of separation using data types, the normal data type is separated from the embedded data type.



NVIDIA.

Jetson Nano System-on-Module
Maxwell GPU + ARM Cortex-A57 + 4GB LPDDR4 + 16GB eMMC

Table 14 CSI Pin Descriptions

Pin	Name	Direction	Type	Description
10	CSI0_CLK_N	Input	MIPI D-PHY	CSI 0 Clock-
12	CSI0_CLK_P	Input	MIPI D-PHY	CSI 0 Clock+
4	CSI0_D0_N	Input	MIPI D-PHY	CSI 0 Data 0-
6	CSI0_D0_P	Input	MIPI D-PHY	CSI 0 Data 0+
16	CSI0_D1_N	Input	MIPI D-PHY	CSI 0 Data 1-
18	CSI0_D1_P	Input	MIPI D-PHY	CSI 0 Data 1+
3	CSI1_D0_N	Input	MIPI D-PHY	CSI 1 Data 0-
5	CSI1_D0_P	Input	MIPI D-PHY	CSI 1 Data 0+
15	CSI1_D1_N	Input	MIPI D-PHY	CSI 1 Data 1-
17	CSI1_D1_P	Input	MIPI D-PHY	CSI 1 Data 1+
28	CSI2_CLK_N	Input	MIPI D-PHY	CSI 2 Clock-
30	CSI2_CLK_P	Input	MIPI D-PHY	CSI 2 Clock+
22	CSI2_D0_N	Input	MIPI D-PHY	CSI 2 Data 0-
24	CSI2_D0_P	Input	MIPI D-PHY	CSI 2 Data 0+
34	CSI2_D1_N	Input	MIPI D-PHY	CSI 2 Data 1-
36	CSI2_D1_P	Input	MIPI D-PHY	CSI 2 Data 1+
27	CSI3_CLK_N	Input	MIPI D-PHY	CSI 3 Clock-
29	CSI3_CLK_P	Input	MIPI D-PHY	CSI 3 Clock+
21	CSI3_D0_N	Input	MIPI D-PHY	CSI 3 Data 0-
23	CSI3_D0_P	Input	MIPI D-PHY	CSI 3 Data 0+
33	CSI3_D1_N	Input	MIPI D-PHY	CSI 3 Data 1-
35	CSI3_D1_P	Input	MIPI D-PHY	CSI 3 Data 1+
52	CSI4_CLK_N	Input	MIPI D-PHY	CSI 4 Clock-
54	CSI4_CLK_P	Input	MIPI D-PHY	CSI 4 Clock+
46	CSI4_D0_N	Input	MIPI D-PHY	CSI 4 Data 0-
48	CSI4_D0_P	Input	MIPI D-PHY	CSI 4 Data 0+
58	CSI4_D1_N	Input	MIPI D-PHY	CSI 4 Data 1-
60	CSI4_D1_P	Input	MIPI D-PHY	CSI 4 Data 1+
40	CSI4_D2_N	Input	MIPI D-PHY	CSI 4 Data 2-
42	CSI4_D2_P	Input	MIPI D-PHY	CSI 4 Data 2+
64	CSI4_D3_N	Input	MIPI D-PHY	CSI 4 Data 3-
66	CSI4_D3_P	Input	MIPI D-PHY	CSI 4 Data 3+

Table 15 Camera Clock and Control Pin Descriptions

Pin	Name	I/O	Pin Type	PoR	Description
213	CAM_I2C_SCL	Bidirectional	Open Drain – 3.3V [DD]	z	Camera I2C Clock
215	CAM_I2C_SDA	Bidirectional	Open Drain – 3.3V [DD]	z	Camera I2C Data



NVIDIA.

Pin	Name	I/O	Pin Type	PoR	Description
116	CAM0_MCLK	Output	CMOS – 1.8V [ST]	PD	Camera 1 Reference Clock
114	CAM0_PWDN	Output	CMOS – 1.8V [ST]	PD	Camera 1 Powerdown or GPIO
122	CAM1_MCLK	Output	CMOS – 1.8V [ST]	PD	Camera 2 Reference Clock
120	CAM1_PWDN	Output	CMOS – 1.8V [ST]	PD	Camera 2 Powerdown or GPIO

4.5 SD / SDIO

Standard	Notes
<i>SD Specifications Part A2 SD Host Controller Standard Specification Version 4.00</i>	
<i>SD Specifications Part 1 Physical Layer Specification Version 4.00</i>	
<i>SD Specifications Part E1 SDIO Specification Version 4.00</i>	Support for SD 4.0 Specification without UHS-II
<i>Embedded Multimedia Card (eMMC), Electrical Standard 5.1</i>	

The SecureDigital (SD)/Embedded MultiMediaCard (eMMC) controller is used to support the on-module eMMC and a single SDIO interface made available for use with SDIO peripherals; it supports Default and High-Speed modes.

The SDMMC controller has a direct memory interface and is capable of initiating data transfers between memory and external device. The SDMMC controller supports both the SD and eMMC bus protocol and has an APB slave interface to access configuration registers. Interface is intended for supporting various compatible peripherals with an SD/MMC interface.

Table 16 SD/SDIO Controller I/O Capabilities

Controller	Bus Width	Supported Voltages (V)	I/O bus clock (MHz)	Max Bandwidth (MBps)	Notes
SD/SDIO Card	4	1.8 / 3.3	208	104	Available at connector for SDIO or SD Card use
eMMC	8	1.8	200	400	On-module eMMC

Table 17 SD/SDIO Pin Descriptions

Pin	Name	I/O	Pin Type	PoR	Description
229	SDMMC_CLK	Output	CMOS – 1.8V / 3.3V [CZ]	PD	SDIO/MMC Clock
227	SDMMC_CMD	Bidirectional	CMOS – 1.8V / 3.3V [CZ]	PU	SDIO/MMC Command
225	SDMMC_DAT3	Bidirectional	CMOS – 1.8V / 3.3V [CZ]	PU	SDIO/MMC Data bus
223	SDMMC_DAT2				
221	SDMMC_DAT1				
219	SDMMC_DAT0				

Note: Pin voltage is determined by LDO on module setting.



NVIDIA.

Jetson Nano System-on-Module
Maxwell GPU + ARM Cortex-A57 + 4GB LPDDR4 + 16GB eMMC

4.6 Inter-IC Sound (I²S)

Standard
Inter-IC Sound (I ² S) specification

The I²S controller transports streaming audio data between system memory and an audio codec. The I²S controller supports I²S format, Left-justified Mode format, Right-justified Mode format, and DSP mode format, as defined in the Philips inter-IC-sound (I²S) bus specification.

The I²S and PCM (master and slave modes) interfaces support clock rates up to 24.5760MHz.

The I²S controller supports point-to-point serial interfaces for the I²S digital audio streams. I²S-compatible products, such as compact disc players, digital audio tape devices, digital sound processors, and those with digital TV sound may be directly connected to the I²S controller. The controller also supports the PCM and telephony mode of data-transfer. Pulse-Code-Modulation (PCM) is a standard method used to digitize audio (particularly voice) patterns for transmission over digital communication channels. The Telephony mode is used to transmit and receive data to and from an external mono CODEC in a slot-based scheme of time-division multiplexing (TDM). The I²S controller supports bidirectional audio streams and can operate in half-duplex or full-duplex mode.

Features:

- Basic I²S modes to be supported (I²S, RJM, LJM and DSP) in both Master and Slave modes.
- PCM mode with short (one-bit-clock wide) and long-fsync (two bit-clocks wide) in both master and slave modes.
- NW-mode with independent slot-selection for both Tx and Rx
- TDM mode with flexibility in number of slots and slot(s) selection.
- Capability to drive-out a High-z outside the prescribed slot for transmission
- Flow control for the external input/output stream.

Table 18 Audio Pin Descriptions

Pin	Name	Direction	Type	PoR	Description
211	GPIO09	Output	CMOS – 1.8V [ST]	PD	Audio Codec Master Clock (AUD_MCLK)
195	I2S0_DIN	Input	CMOS – 1.8V [CZ]	PD	I ² S Audio Port 0 Data In
193	I2S0_DOUT	Output	CMOS – 1.8V [CZ]	PD	I ² S Audio Port 0 Data Out
197	I2S0_FS	Bidirectional	CMOS – 1.8V [CZ]	PD	I ² S Audio Port 0 Frame Select (Left/Right Clock)
199	I2S0_SCLK	Bidirectional	CMOS – 1.8V [CZ]	PD	I ² S Audio Port 0 Clock
222	I2S1_DIN	Input	CMOS – 1.8V [ST]	PD	I ² S Audio Port 1 Data In
220	I2S1_DOUT	Output	CMOS – 1.8V [ST]	PD	I ² S Audio Port 1 Data Out
224	I2S1_FS	Bidirectional	CMOS – 1.8V [ST]	PD	I ² S Audio Port 1 Frame Select (Left/Right Clock)
226	I2S1_SCLK	Bidirectional	CMOS – 1.8V [ST]	PD	I ² S Audio Port 1 Clock



NVIDIA.

4.7 Miscellaneous Interfaces

4.7.1 Inter-Chip Communication (I²C)

Standard
NXP inter-IC-bus (I ² C) specification

This general purpose I²C controller allows system expansion for I²C -based devices as defined in the NXP inter-IC-bus (I²C) specification. The I²C bus supports serial device communications to multiple devices; the I²C controller handles clock source negotiation, speed negotiation for standard and fast devices, 7-bit slave address support according to the I²C protocol and supports master and slave mode of operation.

The I²C controller supports the following operating modes: Master – Standard-mode (up to 100Kbit/s), Fast-mode (up to 400 Kbit/s), Fast-mode plus (Fm+, up to 1Mbit/s); Slave – Standard-mode (up to 100Kbit/s), Fast-mode (up to 400 Kbit/s), Fast-mode plus (Fm+, up to 1Mbit/s).

Table 19 I²C Pin Descriptions

Pin	Name	I/O	Pin Type	PoR	Description
185 187	I2C0_SCL I2C0_SDA	Bidirectional	Open Drain – 3.3V [DD]	z z	Only 3.3V devices supported without level shifter. I ² C 0 Clock/Data pins. On module 2.2kΩ pull-up to 3.3V.
189 191	I2C1_SCL I2C1_SDA	Bidirectional	Open Drain – 3.3V [DD]	z z	Only 3.3V devices supported without level shifter. I ² C 1 Clock/Data pins. On module 2.2kΩ pull-up to 3.3V.
232 234	I2C2_SCL I2C2_SDA	Bidirectional	Open Drain – 1.8V [DD]	z z	Only 1.8V devices supported without level shifter. I ² C 2 Clock/Data pins. On module 2.2kΩ pull-up to 1.8V.
213 215	CAM_I2C_SCL CAM_I2C_SDA	Bidirectional	Open Drain – 3.3V [DD]	z z	Only 3.3V devices supported without level shifter. Camera I ² C Clock/Data pins. On module 4.7kΩ pull-up to 3.3V.

4.7.2 Serial Peripheral Interface (SPI)

The SPI controllers operate up to 65Mbps in master mode and 45Mbps in slave mode. It allows a duplex, synchronous, serial communication between the controller and external peripheral devices. It consists of four signals, SS_N (Chip select), SCK (clock), MOSI (Master data out and Slave data in) and MISO (Slave data out and master data in). The data is transferred on MOSI or MISO based on the data transfer direction on every SCK edge. The receiver always receives the data on the other edge of SCK.

Features:

- Independent Rx FIFO and Tx FIFO.
- Software controlled bit-length supports packet sizes of 1 to 32 bits.
- Packed mode support for bit-length of 7 (8-bit packet size) and 15 (16-bit packet size).
- SS_N can be selected to be controlled by software, or it can be generated automatically by the hardware on packet boundaries.
- Receive compare mode (controller listens for a specified pattern on the incoming data before receiving the data in the FIFO).
- Simultaneous receive and transmit supported
- Supports Master mode. Slave mode has not been validated



Table 20 SPI Pin Descriptions

Pin	Name	Direction	Type	PoR	Description
95	SPI0_CS0*	Bidirectional	CMOS – 1.8V [LV-CZ]	PU	SPI 0 Chip Select 0
97	SPI0_CS1*	Bidirectional	CMOS – 1.8V [LV-CZ]	PU	SPI 0 Chip Select 1
93	SPI0_MISO	Bidirectional	CMOS – 1.8V [LV-CZ]	PD	SPI 0 Master In / Slave Out
89	SPI0_MOSI	Bidirectional	CMOS – 1.8V [LV-CZ]	PD	SPI 0 Master Out / Slave In
91	SPI0_SCK	Bidirectional	CMOS – 1.8V [LV-CZ]	PD	SPI 0 Clock
110	SPI1_CS0*	Bidirectional	CMOS – 1.8V [CZ]	PU	SPI 1 Chip Select 0
112	SPI1_CS1*	Bidirectional	CMOS – 1.8V [CZ]	PU	SPI 1 Chip Select 1
108	SPI1_MISO	Bidirectional	CMOS – 1.8V [CZ]	PD	SPI 1 Master In / Slave Out
104	SPI1_MOSI	Bidirectional	CMOS – 1.8V [CZ]	PD	SPI 1 Master Out / Slave In
106	SPI1_SCK	Bidirectional	CMOS – 1.8V [CZ]	PD	SPI 1 Clock

Figure 5 SPI Master Timing Diagram

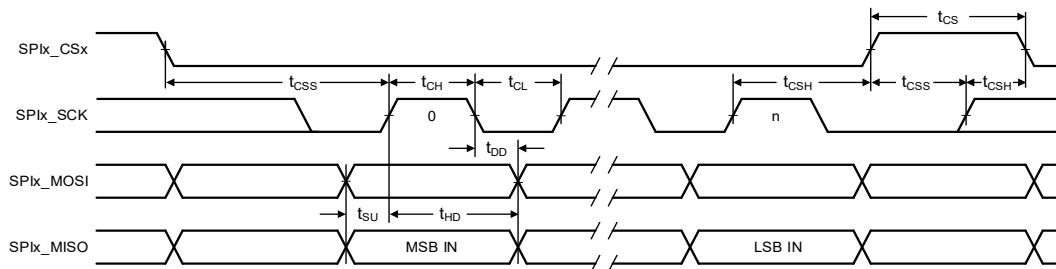
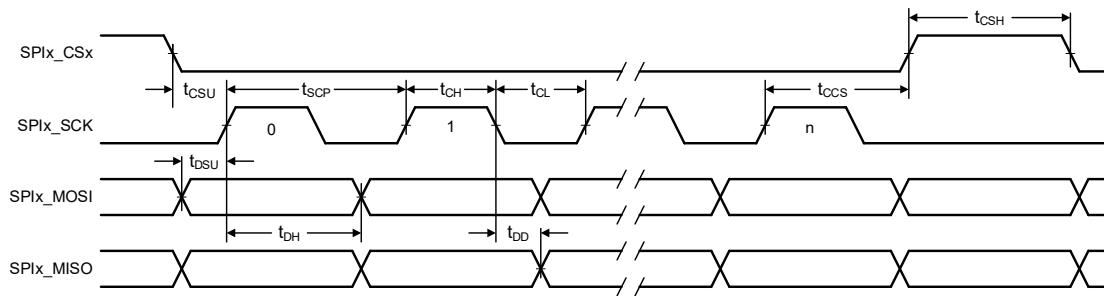


Table 21 SPI Master Timing Parameters

Symbol	Parameter	Minimum	Maximum	Unit
Fsck	SPIx_SCK clock frequency		65	MHz
Psck	SPIx_SCK period	1/Fsck		ns
t _{CH}	SPIx_SCK high time	50%Psck -10%	50%Psck +10%	ns
t _{CL}	SPIx_SCK low time	50%Psck -10%	50%Psck +10%	ns
t _{CFT}	SPIx_SCK rise time (slew rate)	0.1		V/ns
t _{CFT}	SPIx_SCK fall time (slew rate)	0.1		V/ns
t _{SU}	SPIx_MISO setup to SPIx_SCK rising edge	2		ns
t _{HD}	SPIx_MISO hold from SPIx_SCK rising edge	3		ns
t _{DD}	SPIx_MOSI delay from SPIx_SCK falling edge	0	4	ns
t _{css}	SPIx_CSx setup time	2		ns
t _{cs}	SPIx_CSx hold time	3		ns
t _{cs}	SPIx_CSx high time	10		ns

Note: Polarity of SCLK is programmable. Data can be driven or input relative to either the rising edge (shown above) or falling edge.

Figure 6 SPI Slave Timing Diagram

Table 22 SPI Slave Timing Parameters

Symbol	Parameter	Minimum	Maximum	Unit
t_{SCP}	SPIx_SCK period	$2^*(t_{SDD} + t_{MSU}^1)$		ns
t_{SCH}	SPIx_SCK high time	$t_{SDD} + t_{MSU}^1$		ns
t_{SCL}	SPIx_SCK low time	$t_{SDD} + t_{MSU}^1$		ns
t_{SCSU}	SPIx_CSx setup time	1		t_{SCP}
t_{SCSH}	SPIx_CSx high time	1		t_{SCP}
t_{SCCS}	SPIx_SCK rising edge to SPIx_CSx rising edge	1	1	t_{SCP}
t_{SDSU}	SPIx_MOSI setup to SPIx_SCK rising edge	1	1	ns
t_{SDH}	SPIx_MOSI hold from SPIx_SCK rising edge	2	11	ns

1. t_{MSU} is the setup time required by the external master

Note: Polarity of SCLK is programmable. Data can be driven or input relative to either the rising edge (shown above) or falling edge.

4.7.3 UART

UART controller provides serial data synchronization and data conversion (parallel-to-serial and serial-to-parallel) for both receiver and transmitter sections. Synchronization for serial data stream is accomplished by adding start and stop bits to the transmit data to form a data character. Data integrity is accomplished by attaching a parity bit to the data character. The parity bit can be checked by the receiver for any transmission bit errors.

NOTE: The UART receiver input has low baud rate tolerance in 1-stop bit mode. External devices must use 2 stop bits.

In 1-stop bit mode, the Tegra UART receiver can lose sync between Tegra receiver and the external transmitter resulting in data errors/corruption. In 2-stop bit mode, the extra stop bit allows the Tegra UART receiver logic to align properly with the UART transmitter.

Features:

- Synchronization for the serial data stream with start and stop bits to transmit data and form a data character
- Supports both 16450- and 16550-compatible modes. Default mode is 16450
- Device clock up to 200MHz, baud rate of 12.5Mbits/second
- Data integrity by attaching parity bit to the data character
- Support for word lengths from five to eight bits, an optional parity bit and one or two stop bits
- Support for modem control inputs
- DMA capability for both Tx and Rx
- 8-bit x 36 deep Tx FIFO



NVIDIA.

Jetson Nano System-on-Module
Maxwell GPU + ARM Cortex-A57 + 4GB LPDDR4 + 16GB eMMC

- 11-bit x 36 deep Rx FIFO. Three bits of 11 bits per entry log the Rx errors in FIFO mode (break, framing, and parity errors as bits 10, 9, 8 of FIFO entry)
- Auto sense baud detection
- Timeout interrupts to indicate if the incoming stream stopped
- Priority interrupts mechanism
- Flow control support on RTS and CTS
- Internal loopback
- SIR encoding/decoding (3/16 or 4/16 baud pulse widths to transmit bit zero)

Table 23 UART Pin Descriptions

Pin	Name	Direction	Type	PoR	Description
99	UART0_TXD	Output	CMOS – 1.8V [ST]	PD	UART 0 Transmit
101	UART0_RXD	Input	CMOS – 1.8V [ST]	PU	UART 0 Receive
103	UART0_RTS*	Output	CMOS – 1.8V [ST]	PD	UART 0 Request to Send
105	UART0_CTS*	Input	CMOS – 1.8V [ST]	PD	UART 0 Clear to Send
203	UART1_TXD	Output	CMOS – 1.8V [ST]	PD	UART 1 Transmit
205	UART1_RXD	Input	CMOS – 1.8V [ST]	PD	UART 1 Receive
207	UART1_RTS*	Output	CMOS – 1.8V [ST]	PD	UART 1 Request to Send
209	UART1_CTS*	Input	CMOS – 1.8V [ST]	PD	UART 1 Clear to Send
236	UART2_TXD	Output	CMOS – 1.8V [ST]	PD	UART 2 Transmit
238	UART2_RXD	Input	CMOS – 1.8V [ST]	PD	UART 2 Receive

4.7.4 Gigabit Ethernet

The Jetson Nano integrates a Realtek RTL81119ICG Gigabit Ethernet controller. The on-module Ethernet controller supports:

- 10/100/1000 Mbps Gigabit Ethernet
- IEEE 802.3u Media Access Controller (MAC)

Table 24 Gigabit Ethernet Pin Descriptions

Pin	Name	Direction	Type	Description
194	GBE_LED_ACT	Output		Activity LED (yellow) enable
188	GBE_LED_LINK	Output		Link LED (green) enable. Link LED only illuminates if link established is 1000. 100/10 will not cause the Link LED to light up.
184 186	GBE_MDI0_N GBE_MDI0_P	Bidirectional	MDI	GbE Transformer Data 0
190 192	GBE_MDI1_N GBE_MDI1_P	Bidirectional	MDI	GbE Transformer Data 1
196 198	GBE_MDI2_N GBE_MDI2_P	Bidirectional	MDI	GbE Transformer Data 2
202 204	GBE_MDI3_N GBE_MDI3_P	Bidirectional	MDI	GbE Transformer Data 3



4.7.5 Fan

The Jetson Nano includes PWM and Tachometer functionality to enable fan control as part of a thermal solution. The Pulse Width Modulator (PWM) controller is a frequency divider with a varying pulse width. The PWM runs off a device clock programmed in the Clock and Reset controller and can be any frequency up to the device clock maximum speed of 48MHz. The PWFM gets divided by 256 before being subdivided based on a programmable value.

Table 25 Fan Pin Descriptions

Pin	Name	Direction	Type	PoR	Description
230	GPIO14	Output	CMOS – 1.8V [ST]	PD	Fan PWM
208	GPIO08	Input	CMOS – 1.8V [ST]	PD	Fan Tachometer

4.7.6 Debug

A debug interface is supported via JTAG on-module test points or serial interface over UART1. The JTAG interface can be used for SCAN testing or communicating with integrated CPU. See the *NVIDIA Jetson Nano Product Design Guide* for more information.

Table 26 Debug Pin Descriptions

Pin	Name	I/O	Pin Type	PoR	Description
-	JTAG_RTCK	Output	CMOS – 1.8V [JT_RST]	0	Return Test Clock
-	JTAG_TCK	Input	CMOS – 1.8V [JT_RST]	z	Test Clock
-	JTAG_TDI	Input	CMOS – 1.8V [JT_RST]	PU	Test Data In
-	JTAG_TDO	Output	CMOS – 1.8V [ST]	z	Test Data Out
-	JTAG_TMS	Input	CMOS – 1.8V [JT_RST]	PU	Test Mode Select
-	JTAG_GP0	Input	CMOS – 1.8V [JT_RST]	PD	Test Reset
236	UART2_TXD	Output	CMOS – 1.8V [ST]	PD	Debug UART Transmit
238	UART2_RXD	Input	CMOS – 1.8V [ST]	PD	Debug UART Receive



5.0 Physical / Electrical Characteristics

5.1 Operating and Absolute Maximum Ratings

The parameters listed in following table are specific to a temperature range and operating voltage. Operating the Jetson Nano module beyond these parameters is not recommended. Exceeding these conditions for extended periods may adversely affect device reliability.

WARNING: Exceeding the listed conditions may damage and/or affect long-term reliability of the part.
The Jetson Nano module should never be subjected to conditions extending beyond the ratings listed below.

Table 27 Recommended Operating Conditions

Symbol	Parameter	Minimum	Typical	Maximum	Unit
VDD _{DC}	VDD_IN	4.75	5.0	5.25	V
	PMIC_BBAT	1.65		5.5	V

Absolute maximum ratings describe stress conditions. These parameters do not set minimum and maximum operating conditions that will be tolerated over extended periods of time. If the device is exposed to these parameters for extended periods of time, no guarantee is made and device reliability may be affected. It is not recommended to operate the Jetson Nano module under these conditions.

Table 28 Absolute Maximum Ratings

Symbol	Parameter	Minimum	Maximum	Unit	Notes
VDD _{MAX}	VDD_IN	-0.5	5.5	V	
	PMIC_BBAT	-0.3	6.0	V	
IDD _{MAX}	VDD_IN I _{max}		5	A	
V _{M_PIN}	Voltage applied to any powered I/O pin	-0.5	VDD + 0.5	V	VDD + 0.5V when CARRIER_PWR_ON high & associated I/O rail powered. I/O pins cannot be high (>0.5V) before CARRIER_PWR_ON goes high. When CARRIER_PWR_ON is low, the maximum voltage applied to any I/O pin is 0.5V
	DD pins configured as open drain	-0.5	3.63	V	The pin's output-driver must be set to open-drain mode
T _{OP}	Operating Temperature	-25	97	°C	See the <i>Jetson Nano Thermal Design Guide</i> for details.
T _{STG}	Storage Temperature (ambient)	-40	80	°C	
M _{MAX}	Mounting Force		4.0	kgf	kilogram-force (kgf). Maximum force applied to PCB. See the <i>Jetson Nano Thermal Design Guide</i> for additional details on mounting a thermal solution.



5.2 Digital Logic

Voltages less than the minimum stated value can be interpreted as an undefined state or logic level low which may result in unreliable operation. Voltages exceeding the maximum value can damage and/or adversely affect device reliability.

Table 29. CMOS Pin Type DC Characteristics

Symbol	Description	Minimum	Maximum	Units
V_{IL}	Input Low Voltage	-0.5	0.25 x VDD	V
V_{IH}	Input High Voltage	0.75 x VDD	0.5 + VDD	V
V_{OL}	Output Low Voltage ($I_{OL} = 1\text{mA}$)	---	0.15 x VDD	V
V_{OH}	Output High Voltage ($I_{OH} = -1\text{mA}$)	0.85 x VDD	---	V

Table 30 Open Drain Pin Type DC Characteristics

Symbol	Description	Minimum	Maximum	Units
V_{IL}	Input Low Voltage	-0.5	0.25 x VDD	V
V_{IH}	Input High Voltage	0.75 x VDD	3.63	V
V_{OL}	Output Low Voltage ($I_{OL} = 1\text{mA}$)	---	0.15 x VDD	V
	I2C[1,0] Output Low Voltage ($I_{OL} = 2\text{mA}$) (see note)	---	0.3 x VDD	V
V_{OH}	Output High Voltage ($I_{OH} = -1\text{mA}$)	0.85 x VDD	---	V

Note: I2C[1,0]_[SCL, SDA] pins pull-up to 3.3V through on module 2.2kΩ resistor. I2C2_[SCL, SDA] pins pull-up to 1.8V through on module 2.2kΩ resistor.

5.3 Environmental & Mechanical Screening

Module performance was assessed against a series of industry standard tests designed to evaluate robustness and estimate the failure rate of an electronic assembly in the environment in which it will be used. Mean Time Between Failures (MTBF) calculations are produced in the design phase to predict a product's future reliability in the field.

Table 31 Jetson Nano Reliability Report

Test	Reference Standards / Test Conditions
Temperature Humidity Biased	JESD22-A101 85°C / 85% RH, 168 hours, Power ON
Temperature Cycling	JESD22-A104, IPC9701 -40°C to 105°C, 250 cycles, non-operational
Humidity Steady State	NVIDIA Standard 45°C 90% RH 336hrs, operational
Mechanical Shock – 140G	JESD22-B110 140G, half sine, 1 shock/orientation, 6 orientations total, non-operational
Mechanical Shock – 50G	IEC60068-2-27 50G, half sine, 1 shock/orientation, 6 orientations total, operational
Connector Insertion Cycling	EIA-364 30 cycles
Sine Vibration – 3G	IEC60068-2-6 3G, 10-500 Hz, 1 sweep/axis, 3 axes total, non-operational
Random Vibration – 2G	IEC60068-2-64 10-500 Hz, 2 Grms, 1 hour/axis, non-operational



Jetson Nano System-on-Module
Maxwell GPU + ARM Cortex-A57 + 4GB LPDDR4 + 16GB eMMC

NVIDIA.

Test	Reference Standards / Test Conditions
Random Vibration – 1G	IEC60068-2-64 10-500 Hz, 1 Grms, 1 hour/axis, operational
Hard Boot	NVIDIA Standard Power ON/OFF, ON for 150 sec OFF for 30 sec 1000 cycles at 25°C, 1000 cycles at -40°C
Operational Low Temp	NVIDIA Standard -5°C, 24 hours, operational
Operational High Temp	NVIDIA Standard 40°C, 90%RH, 168 hours, operational
MTBF / Failure Rate: 3,371K Hours	Telcordia SR-332, ISSUE 3 Parts Count (Method I) Controlled Environment (GB), T = 35°C, CL = 90%
MTBF / Failure Rate: 1,836K Hours	Telcordia SR-332, ISSUE 3 Parts Count (Method I) Uncontrolled Environment (GF), T = 35°C, CL = 90%
MTBF / Failure Rate: 957K Hours	Telcordia SR-332, ISSUE 3 Parts Count (Method I) Uncontrolled Environment (GM), T = 35°C, CL = 90%



5.4 Pinout

Signal Name	Pin # Top Odd	Pin # Bottom Even	Signal Name
GND	1	2	GND
CSI1_D0_N	3	4	CSI0_D0_N
CSI1_D0_P	5	6	CSI0_D0_P
GND	7	8	GND
RSVD	9	10	CSI0_CLK_N
RSVD	11	12	CSI0_CLK_P
GND	13	14	GND
CSI1_D1_N	15	16	CSI0_D1_N
CSI1_D1_P	17	18	CSI0_D1_P
GND	19	20	GND
CSI3_D0_N	21	22	CSI2_D0_N
CSI3_D0_P	23	24	CSI2_D0_P
GND	25	26	GND
CSI3_CLK_N	27	28	CSI2_CLK_N
CSI3_CLK_P	29	30	CSI2_CLK_P
GND	31	32	GND
CSI3_D1_N	33	34	CSI2_D1_N
CSI3_D1_P	35	36	CSI2_D1_P
GND	37	38	GND
DP0_TXD0_N	39	40	CSI4_D2_N
DP0_TXD0_P	41	42	CSI4_D2_P
GND	43	44	GND
DP0_TXD1_N	45	46	CSI4_D0_N
DP0_TXD1_P	47	48	CSI4_D0_P
GND	49	50	GND
DP0_RXD2_N	51	52	CSI4_CLK_N
DP0_RXD2_P	53	54	CSI4_CLK_P
GND	55	56	GND
DP0_RXD3_N	57	58	CSI4_D1_N
DP0_RXD3_P	59	60	CSI4_D1_P
GND	61	62	GND
DP1_TXD0_N	63	64	CSI4_D3_N
DP1_TXD0_P	65	66	CSI4_D3_P
GND	67	68	GND
DP1_RXD1_N	69	70	DSI_D0_N
DP1_RXD1_P	71	72	DSI_D0_P
GND	73	74	GND
DP1_RXD2_N	75	76	DSI_CLK_N
DP1_RXD2_P	77	78	DSI_CLK_P
GND	79	80	GND
DP1_RXD3_N	81	82	DSI_D1_N
DP1_RXD3_P	83	84	DSI_D1_P
GND	85	86	GND
GPIO0	87	88	DP0_HPD
SPI0_MOSI	89	90	DP0_AUX_N
SPI0_SCK	91	92	DP0_AUX_P
SPI0_MISO	93	94	HDMI_CEC
SPI0_CS0*	95	96	DP1_HPD
SPI0_CS1*	97	98	DP1_AUX_N
UART0_RXD	99	100	DP1_AUX_P
UART0_RXD	101	102	GND
UART0_RTS*	103	104	SPI1_MOSI
UART0_CTS*	105	106	SPI1_SCK
GND	107	108	SPI1_MISO
USB0_D_N	109	110	SPI1_CS0*
USB0_D_P	111	112	SPI1_CS1*
GND	113	114	CAM0_PWDN
USB1_D_N	115	116	CAM0_MCLK
USB1_D_P	117	118	GPIO01
GND	119	120	CAM1_PWDN
USB2_D_N	121	122	CAM1_MCLK
USB2_D_P	123	124	GPIO02
GND	125	126	GPIO03
GPIO04	127	128	GPIO05
GND	129	130	GPIO06
PCIE0_RX0_N	131	132	GND

Signal Name	Pin # Top Odd	Pin # Bottom Even	Signal Name
PCIE0_RX0_P	133	134	PCIE0_TX0_N
GND	135	136	PCIE0_TX0_P
PCIE0_RX1_N	137	138	GND
PCIE0_RX1_P	139	140	PCIE0_TX1_N
GND	141	142	PCIE0_TX1_P
RSVD	143	144	GND
KEY	KEY	KEY	KEY
RSVD	145	146	GND
GND	147	148	PCIE0_TX2_N
PCIE0_RX2_N	149	150	PCIE0_TX2_P
PCIE0_RX2_P	151	152	GND
GND	153	154	PCIE0_TX3_N
PCIE0_RX3_N	155	156	PCIE0_TX3_P
PCIE0_RX3_P	157	158	GND
GND	159	160	PCIE0_CLK_N
USBSS_RX_N	161	162	PCIE0_CLK_P
USBSS_RX_P	163	164	GND
GND	165	166	USBSS_TX_N
RSVD	167	168	USBSS_TX_P
RSVD	169	170	GND
GND	171	172	RSVD
RSVD	173	174	RSVD
RSVD	175	176	GND
GND	177	178	MOD_SLEEP*
PCIE_WAKE*	179	180	PCIE0_CLKREQ*
PCIE_RST*	181	182	RSVD
RSVD	183	184	GBE_MDIO_N
I2C0_SCL	185	186	GBE_MDIO_P
I2C0_SDA	187	188	GBE_LED_LINK
I2C1_SCL	189	190	GBE_MD1_N
I2C1_SDA	191	192	GBE_MD1_P
I2S0_DOUT	193	194	GBE_LED_ACT
I2S0_DIN	195	196	GBE_MD2_N
I2S0_FS	197	198	GBE_MD2_P
I2S0_SCLK	199	200	GND
GND	201	202	GBE_MD3_N
UART1_RXD	203	204	GBE_MD3_P
UART1_RXD	205	206	GPIO07
UART1_RTS*	207	208	GPIO08
UART1_CTS*	209	210	CLK_32K_OUT
GPIO09	211	212	GPIO10
CAM_I2C_SCL	213	214	FORCE_RECOVERY*
CAM_I2C_SDA	215	216	GPIO11
GND	217	218	GPIO12
SDMMC_DAT0	219	220	I2S1_DOUT
SDMMC_DAT1	221	222	I2S1_DIN
SDMMC_DAT2	223	224	I2S1_FS
SDMMC_DAT3	225	226	I2S1_SCLK
SDMMC_CMD	227	228	GPIO13
SDMMC_CLK	229	230	GPIO14
GND	231	232	I2C2_SCL
SHUTDOWN_REQ*	233	234	I2C2_SDA
PMIC_BBAT	235	236	UART2_RXD
POWER_EN	237	238	UART2_RXD
SYS_RESET*	239	240	SLEEP/WAKE*
GND	241	242	GND
GND	243	244	GND
GND	245	246	GND
GND	247	248	GND
GND	249	250	GND
VDD_IN	251	252	VDD_IN
VDD_IN	253	254	VDD_IN
VDD_IN	255	256	VDD_IN
VDD_IN	257	258	VDD_IN
VDD_IN	259	260	VDD_IN

5.5 Package Drawing and Dimensions

Table 32 Module Dimensions

Description	Minimum	Typical	Maximum	Unit
Connector to opposite side			45	mm
Side (perpendicular to connector) to opposite side			69.6	mm
SoC height	1.36	1.51	1.66	mm

Figure 7 Module Top and Side View with Cover Outline

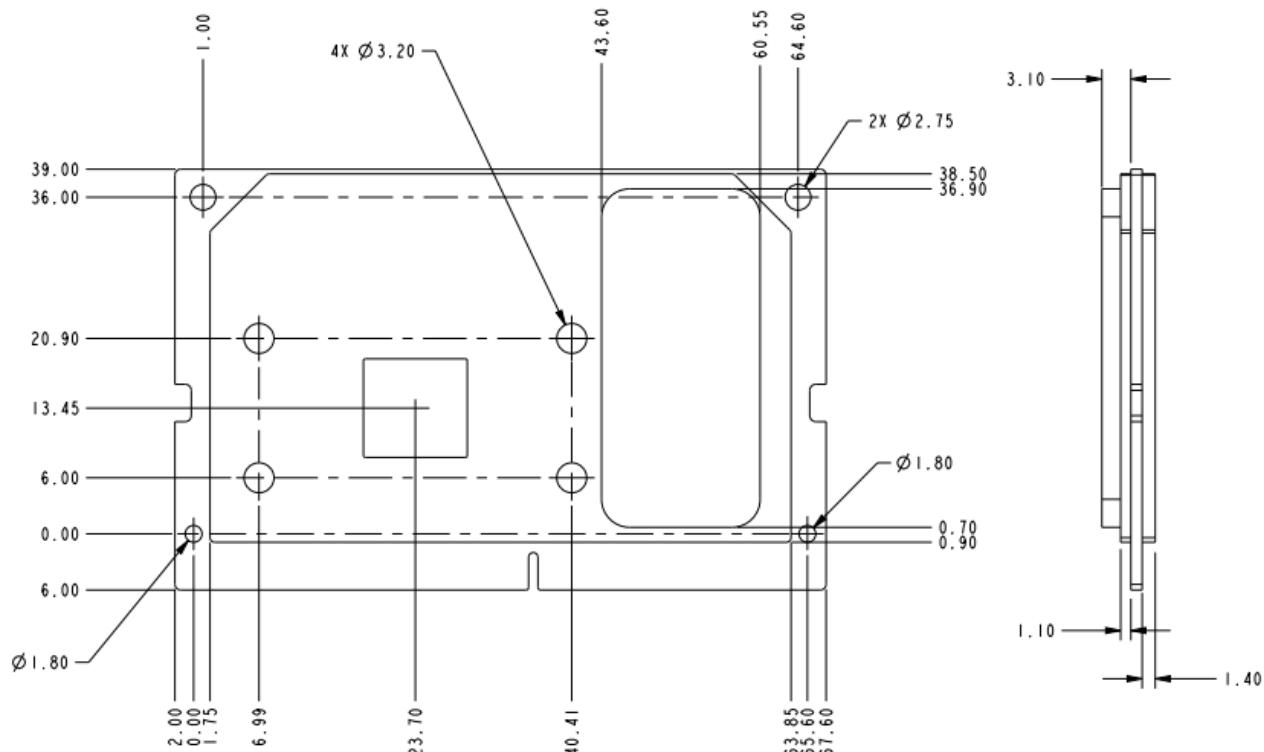
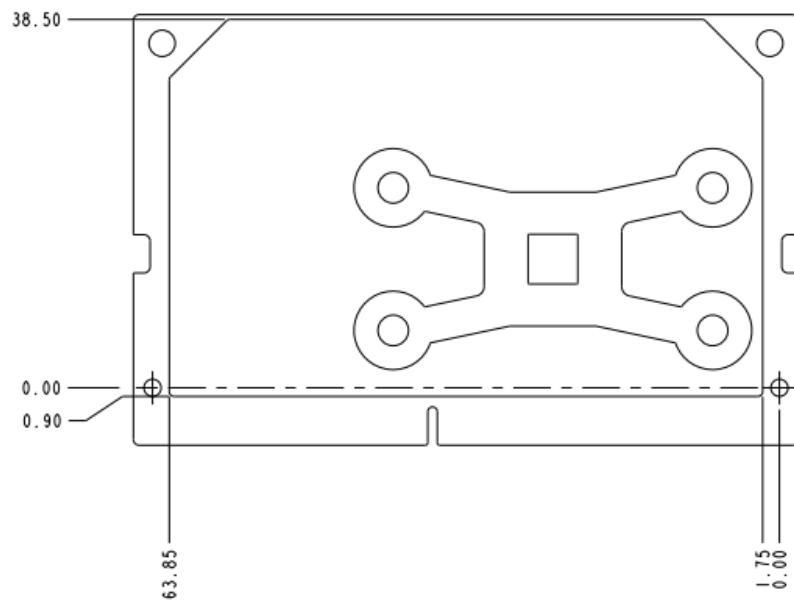
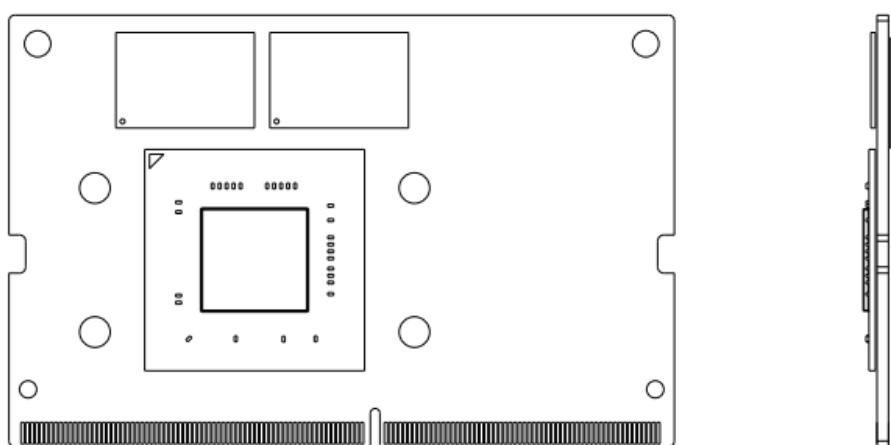
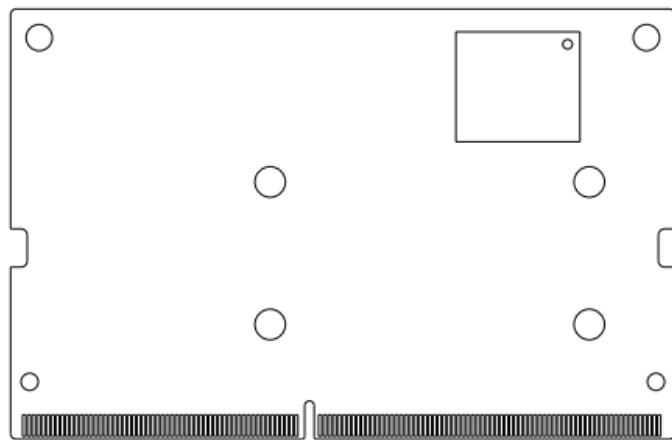


Figure 8 Module Bottom with Cover Outline**Figure 9 Module Top Showing DRAM Placement and Side View**



Jetson Nano System-on-Module
Maxwell GPU + ARM Cortex-A57 + 4GB LPDDR4 + 16GB eMMC

Figure 10 Module Bottom Showing EMMC Placement



Notice

The information provided in this specification is believed to be accurate and reliable as of the date provided. However, NVIDIA Corporation ("NVIDIA") does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This publication supersedes and replaces all other specifications for the product that may have been previously supplied.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and other changes to this specification, at any time and/or to discontinue any product or service without notice. Customer should obtain the latest relevant specification before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer. NVIDIA hereby expressly objects to applying any customer general terms and conditions with regard to the purchase of the NVIDIA product referenced in this specification.

NVIDIA products are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer's own risk.

NVIDIA makes no representation or warranty that products based on these specifications will be suitable for any specified use without further testing or modification. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer's sole responsibility to ensure the product is suitable and fit for the application planned by customer and to do the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer's product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this specification. NVIDIA does not accept any liability related to any default, damage, costs or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this specification, or (ii) customer product designs.

No license, either express or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this specification. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA. Reproduction of information in this specification is permissible only if reproduction is approved by NVIDIA in writing, is reproduced without alteration, and is accompanied by all associated conditions, limitations, and notices.

ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA's aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the NVIDIA terms and conditions of sale for the product.

HDMI

HDMI, the HDMI logo, and High-Definition Multimedia Interface are trademarks or registered trademarks of HDMI Licensing LLC.

ARM

ARM, AMBA and ARM Powered are registered trademarks of ARM Limited. Cortex, MPCore and Mali are trademarks of ARM Limited. All other brands or product names are the property of their respective holders. "ARM" is used to represent ARM Holdings plc; its operating company ARM Limited; and the regional subsidiaries ARM Inc.; ARM KK; ARM Korea Limited.; ARM Taiwan Limited; ARM France SAS; ARM Consulting (Shanghai) Co. Ltd.; ARM Germany GmbH; ARM Embedded Technologies Pvt. Ltd.; ARM Norway, AS and ARM Sweden AB

OpenCL

OpenCL is a trademark of Apple Inc. used under license to the Khronos Group Inc.

Trademarks

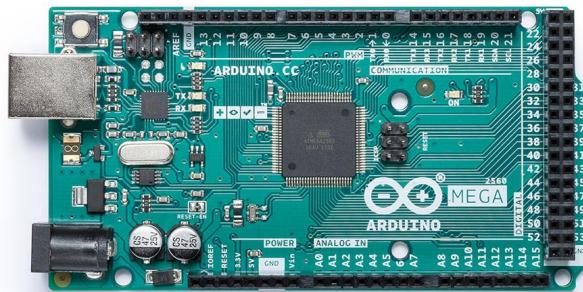
NVIDIA, the NVIDIA logo, and Tegra are trademarks and/or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright

© 2014 – 2022 NVIDIA Corporation. All rights reserved



2 ARDUINO MEGA DATA-SHEET



Description

Arduino® Mega 2560 is an exemplary development board dedicated for building extensive applications as compared to other maker boards by Arduino. The board accommodates the ATmega2560 microcontroller, which operates at a frequency of 16 MHz. The board contains 54 digital input/output pins, 16 analog inputs, 4 UARTs (hardware serial ports), a USB connection, a power jack, an ICSP header, and a reset button.

Target Areas

3D Printing, Robotics, Maker



Features

- **ATmega2560 Processor**

- Up to 16 MIPS Throughput at 16MHz
- 256k bytes (of which 8k is used for the bootloader)
- 4k bytes EEPROM
- 8k bytes Internal SRAM
- 32 × 8 General Purpose Working Registers
- Real Time Counter with Separate Oscillator
- Four 8-bit PWM Channels
- Four Programmable Serial USART
- Controller/Peripheral SPI Serial Interface

- **ATmega16U2**

- Up to 16 MIPS Throughput at 16 MHz
- 16k bytes ISP Flash Memory
- 512 bytes EEPROM
- 512 bytes SRAM
- USART with SPI master only mode and hardware flow control (RTS/CTS)
- Master/Slave SPI Serial Interface

- **Sleep Modes**

- Idle
- ADC Noise Reduction
- Power-save
- Power-down
- Standby
- Extended Standby

- **Power**

- USB Connection
- External AC/DC Adapter

- **I/O**

- 54 Digital
- 16 Analog
- 15 PWM Output



Contents

1 The Board	5
1.1 Application Examples	5
1.2 Accessories	5
1.3 Related Products	5
2 Ratings	6
2.1 Recommended Operating Conditions	6
2.2 Power Consumption	6
3 Functional Overview	6
3.1 Block Diagram	6
3.2 Board Topology	7
3.3 Processor	8
3.4 Power Tree	8
4 Board Operation	9
4.1 Getting Started - IDE	9
4.2 Getting Started - Arduino Web Editor	9
4.3 Sample Sketches	9
4.4 Online Resources	9
5 Connector Pinouts	10
5.1 Analog	11
5.2 Digital	11
5.3 ATMEGA16U2 JP5	13
5.4 ATMEGA16U2 ICSP1	13
5.5 Digital Pins D22 - D53 LHS	13
5.6 Digital Pins D22 - D53 RHS	14
6 Mechanical Information	14
6.1 Board Outline	14
6.2 Board Mount Holes	15
7 Declaration of Conformity CE DoC (EU)	15
8 Declaration of Conformity to EU RoHS & REACH 211 01/19/2021	
9 Conflict Minerals Declaration	17
10 FCC Caution	17
11 Company Information	18
12 Reference Documentation	18





1 The Board

Arduino® Mega 2560 is a successor board of Arduino Mega, it is dedicated to applications and projects that require large number of input output pins and the use cases which need high processing power. The Arduino® Mega 2560 comes with a much larger set of IOs when we compare it with traditional Uno board considering the form factor of both the boards.

1.1 Application Examples

- **Robotics:** Featuring the high processing capacity, the Arduino Mega 2560 can handle the extensive robotic applications. It is compatible with the motor controller shield that enables it to control multiple motors at an instance, thus making it perfect for robotic applications. The large number of I/O pins can accommodate many robotic sensors as well.
- **3D Printing:** Algorithms play a significant role in implementation of 3D printers. Arduino Mega 2560 has the power to process these complex algorithms required for 3D printing. Additionally, the slight changes to the code is easily possible with the Arduino IDE and thus 3D printing programs can be customized according to user requirements.
- **Wi-Fi:** Integrating wireless functionality enhances the utility of the applications. Arduino Mega 2560 is compatible with WiFi shields hence allowing the wireless features for the applications in 3D printing and Robotics.

1.2 Accessories

1.3 Related Products

- Arduino® Uno Rev 3
- Arduino® Nano
- Arduino® DUE without headers

2 Ratings

2.1 Recommended Operating Conditions

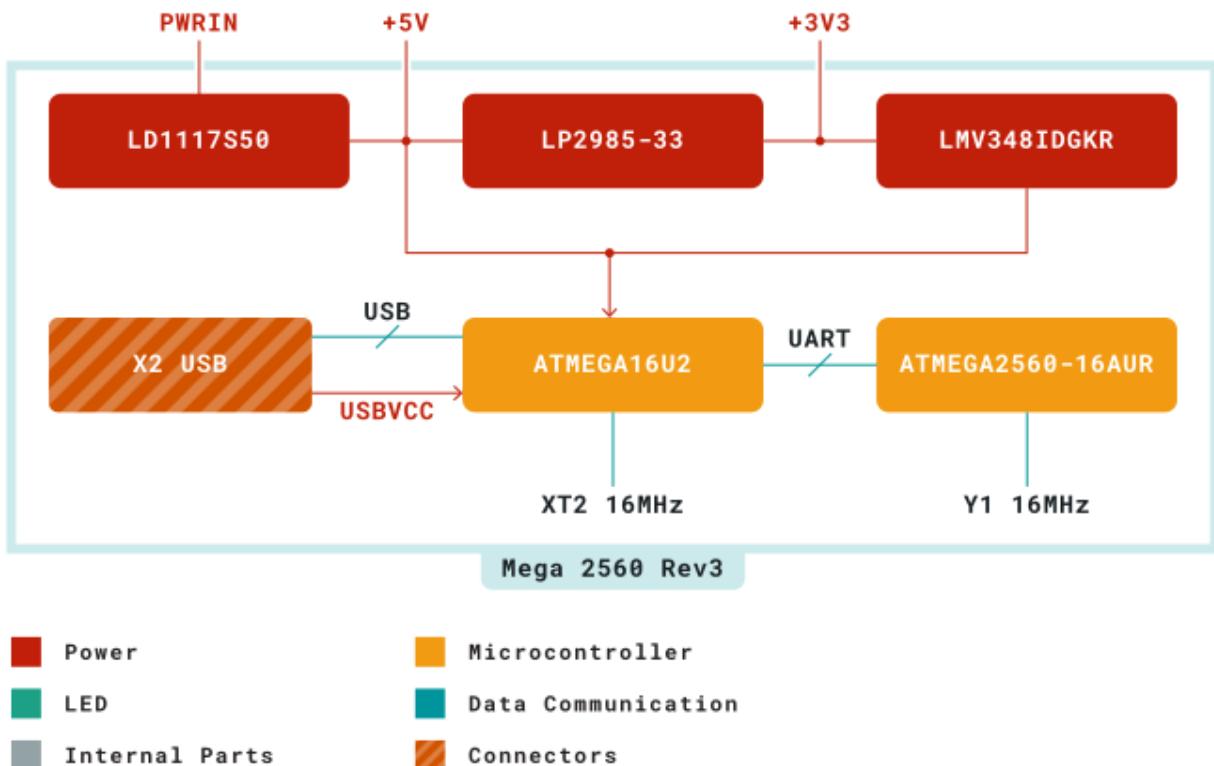
Symbol	Description	Min	Max
TOP	Operating temperature:	-40 °C	85 °C

2.2 Power Consumption

Symbol	Description	Min	Typ	Max	Unit
PWRIN	Input supply from power jack		TBC		mW
USB VCC	Input supply from USB		TBC		mW
VIN	Input from VIN pad		TBC		mW

3 Functional Overview

3.1 Block Diagram

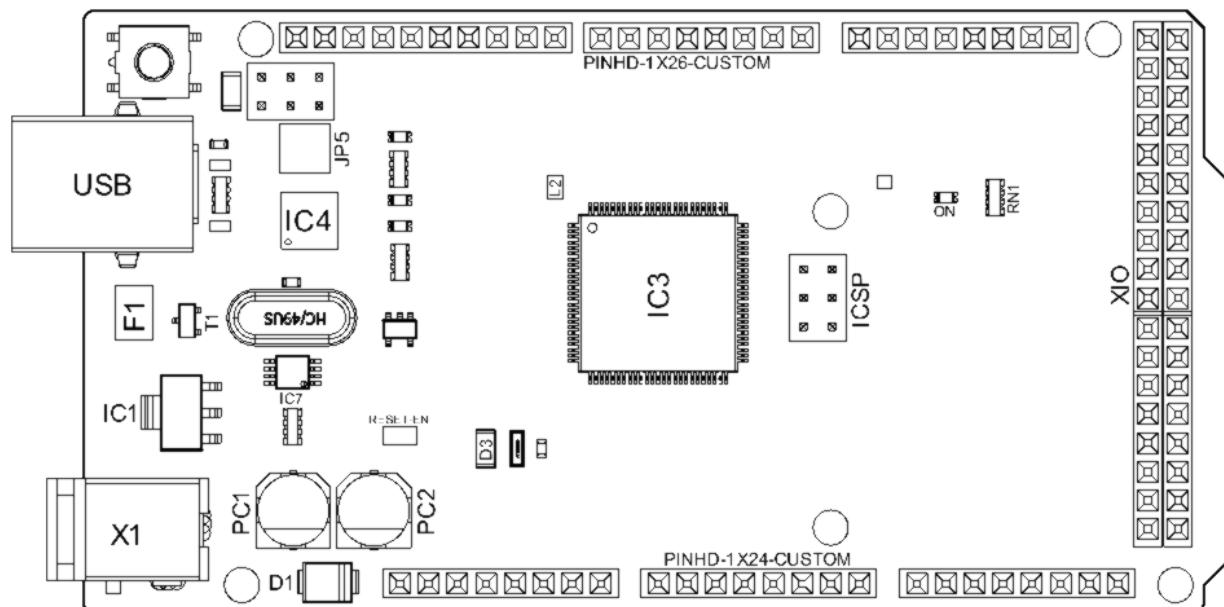


Arduino MEGA Block Diagram



3.2 Board Topology

Front View



Arduino MEGA Top View

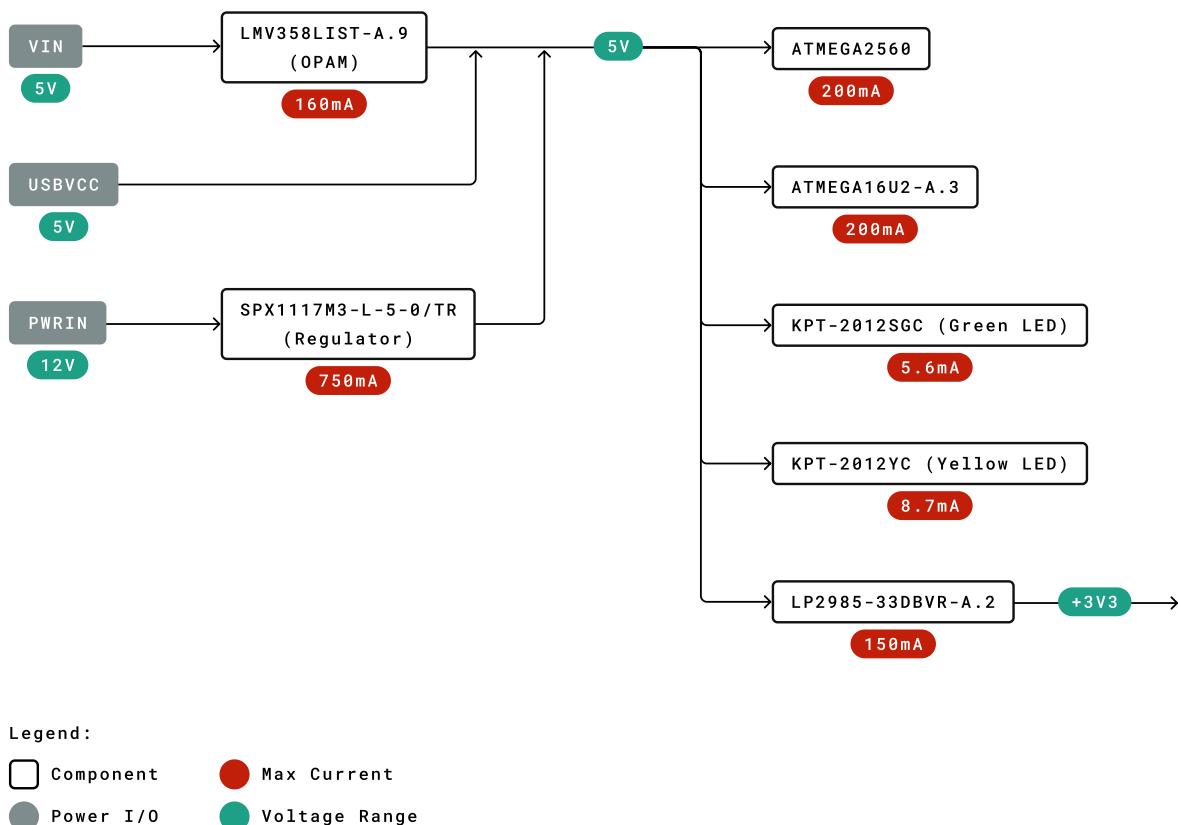
Ref.	Description	Ref.	Description
USB	USB B Connector	F1	Chip Capacitor
IC1	5V Linear Regulator	X1	Power Jack Connector
JP5	Plated Holes	IC4	ATmega16U2 chip
PC1	Electrolytic Aluminum Capacitor	PC2	Electrolytic Aluminum Capacitor
D1	General Purpose Rectifier	D3	General Purpose Diode
L2	Fixed Inductor	IC3	ATmega2560 chip
ICSP	Connector Header	ON	Green LED
RN1	Resistor Array	XIO	Connector



3.3 Processor

Primary processor of Arduino Mega 2560 Rev3 board is ATmega2560 chip which operates at a frequency of 16 MHz. It accommodates a large number of input and output lines which gives the provision of interfacing many external devices. At the same time the operations and processing is not slowed due to its significantly larger RAM than the other processors. The board also features a USB serial processor ATmega16U2 which acts an interface between the USB input signals and the main processor. This increases the flexibility of interfacing and connecting peripherals to the Arduino Mega 2560 Rev 3 board.

3.4 Power Tree



Power Tree



4 Board Operation

4.1 Getting Started - IDE

If you want to program your Arduino® MEGA 2560 while offline you need to install the Arduino® Desktop IDE [1] To connect the Arduino® MEGA 2560 to your computer, you'll need a Type-B USB cable. This also provides power to the board, as indicated by the LED.

4.2 Getting Started - Arduino Web Editor

All Arduino® boards, including this one, work out-of-the-box on the Arduino® Web Editor [2], by just installing a simple plugin.

The Arduino® Web Editor is hosted online, therefore it will always be up-to-date with the latest features and support for all boards. Follow [3] to start coding on the browser and upload your sketches onto your board.

4.3 Sample Sketches

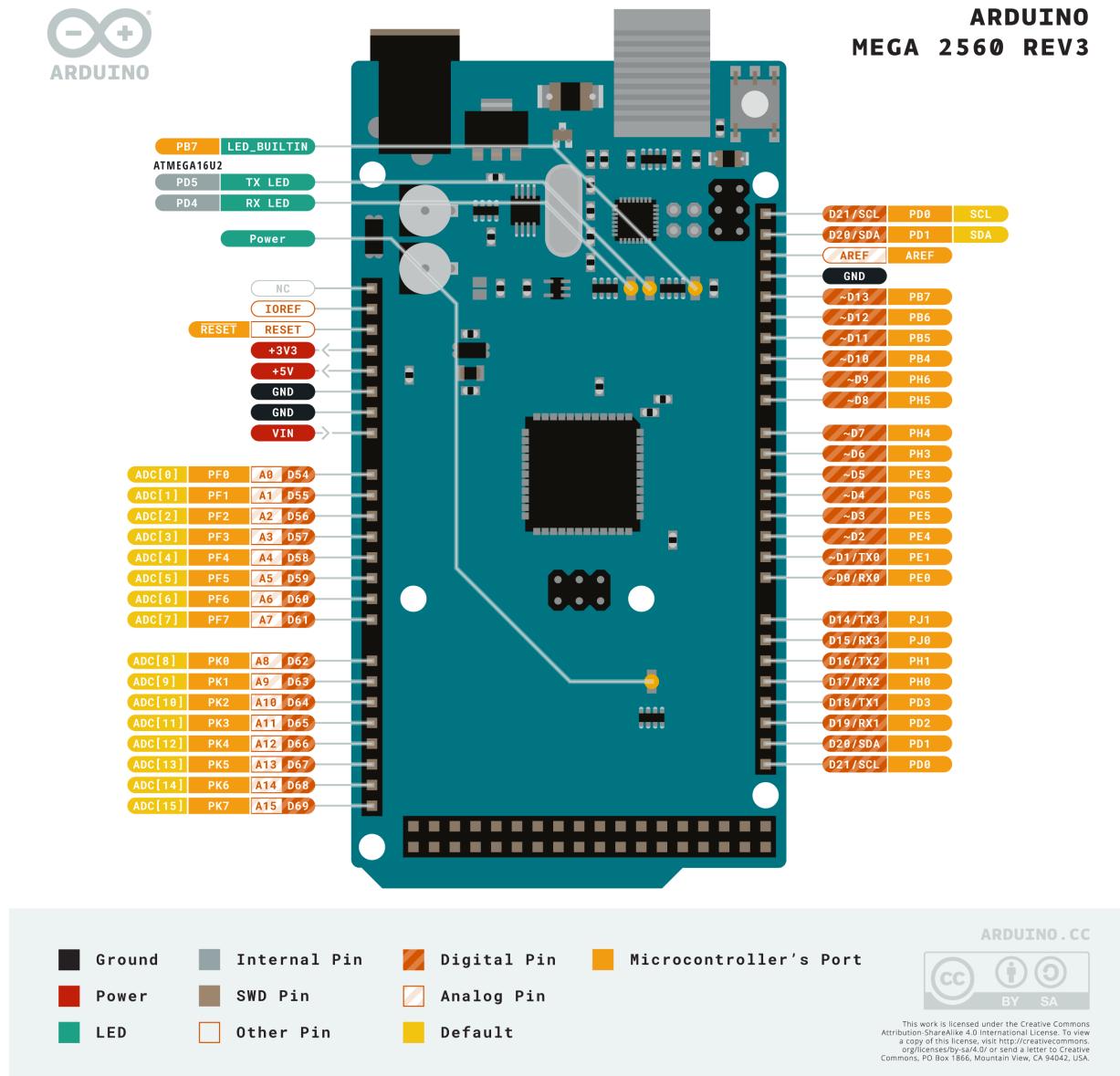
Sample sketches for the Arduino® MEGA 2560 can be found either in the "Examples" menu in the Arduino® IDE

4.4 Online Resources

Now that you have gone through the basics of what you can do with the board you can explore the endless possibilities it provides by checking exciting projects on ProjectHub [5], the Arduino® Library Reference [6] and the online store [7] where you will be able to complement your board with sensors, actuators and more.



5 Connector Pinouts



Arduino Mega Pinout



5.1 Analog

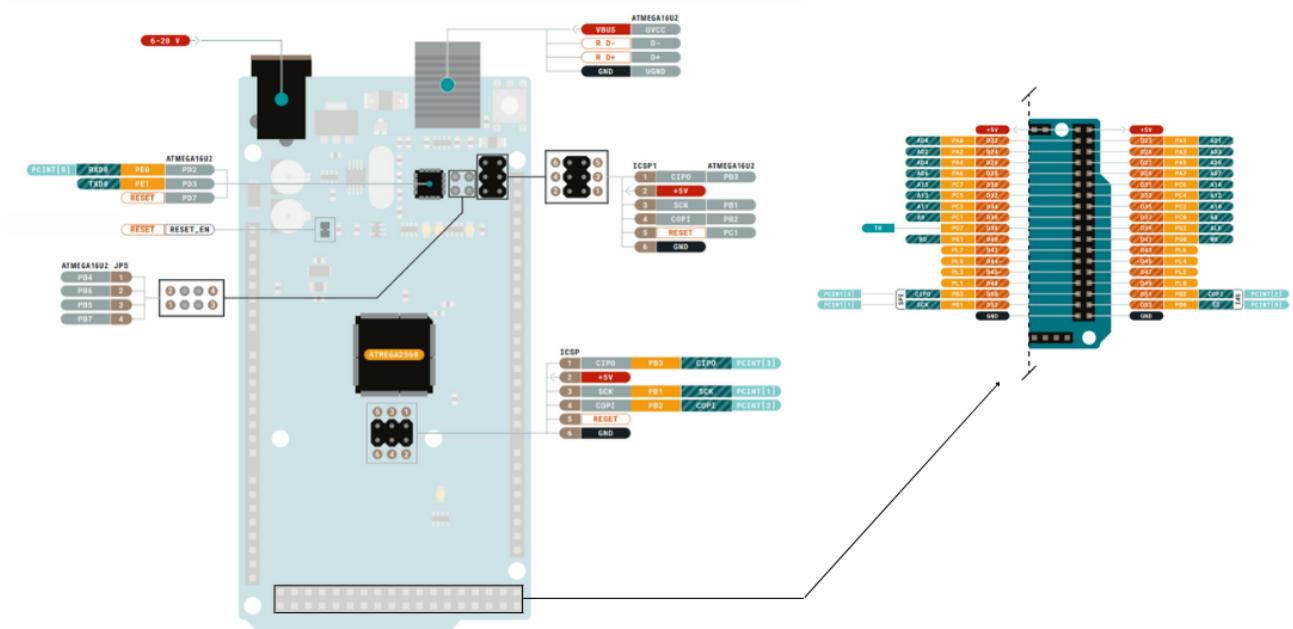
Pin	Function	Type	Description
1	NC	NC	Not Connected
2	IOREF	IOREF	Reference for digital logic V - connected to 5V
3	Reset	Reset	Reset
4	+3V3	Power	+3V3 Power Rail
5	+5V	Power	+5V Power Rail
6	GND	Power	Ground
7	GND	Power	Ground
8	VIN	Power	Voltage Input
9	A0	Analog	Analog input 0 /GPIO
10	A1	Analog	Analog input 1 /GPIO
11	A2	Analog	Analog input 2 /GPIO
12	A3	Analog	Analog input 3 /GPIO
13	A4	Analog	Analog input 4 /GPIO
14	A5	Analog	Analog input 5 /GPIO
15	A6	Analog	Analog input 6 /GPIO
16	A7	Analog	Analog input 7 /GPIO
17	A8	Analog	Analog input 8 /GPIO
18	A9	Analog	Analog input 9 /GPIO
19	A10	Analog	Analog input 10 /GPIO
20	A11	Analog	Analog input 11 /GPIO
21	A12	Analog	Analog input 12 /GPIO
22	A13	Analog	Analog input 13 /GPIO
23	A14	Analog	Analog input 14 /GPIO
24	A15	Analog	Analog input 15 /GPIO

5.2 Digital

Pin	Function	Type	Description
1	D21/SCL	Digital Input/I2C	Digital input 21/I2C Dataline
2	D20/SDA	Digital Input/I2C	Digital input 20/I2C Dataline
3	AREF	Digital	Analog Reference Voltage
4	GND	Power	Ground
5	D13	Digital/GPIO	Digital input 13/GPIO
6	D12	Digital/GPIO	Digital input 12/GPIO
7	D11	Digital/GPIO	Digital input 11/GPIO
8	D10	Digital/GPIO	Digital input 10/GPIO
9	D9	Digital/GPIO	Digital input 9/GPIO
10	D8	Digital/GPIO	Digital input 8/GPIO
11	D7	Digital/GPIO	Digital input 7/GPIO
12	D6	Digital/GPIO	Digital input 6/GPIO
13	D5	Digital/GPIO	Digital input 5/GPIO
14	D4	Digital/GPIO	Digital input 4/GPIO



Pin	Function	Type	Description
15	D3	Digital/GPIO	Digital input 3 /GPIO
16	D2	Digital/GPIO	Digital input 2 /GPIO
17	D1/TX0	Digital/GPIO	Digital input 1 /GPIO
18	D0/Tx1	Digital/GPIO	Digital input 0 /GPIO
19	D14	Digital/GPIO	Digital input 14 /GPIO
20	D15	Digital/GPIO	Digital input 15 /GPIO
21	D16	Digital/GPIO	Digital input 16 /GPIO
22	D17	Digital/GPIO	Digital input 17 /GPIO
23	D18	Digital/GPIO	Digital input 18 /GPIO
24	D19	Digital/GPIO	Digital input 19 /GPIO
25	D20	Digital/GPIO	Digital input 20 /GPIO
26	D21	Digital/GPIO	Digital input 21 /GPIO



Arduino Mega Pinout



5.3 ATMEGA16U2 JP5

Pin	Function	Type	Description
1	PB4	Internal	Serial Wire Debug
2	PB6	Internal	Serial Wire Debug
3	PB5	Internal	Serial Wire Debug
4	PB7	Internal	Serial Wire Debug

5.4 ATMEGA16U2 ICSP1

Pin	Function	Type	Description
1	CIPO	Internal	Controller In Peripheral Out
2	+5V	Internal	Power Supply of 5V
3	SCK	Internal	Serial Clock
4	COPI	Internal	Controller Out Peripheral In
5	RESET	Internal	Reset
6	GND	Internal	Ground

5.5 Digital Pins D22 – D53 LHS

Pin	Function	Type	Description
1	+5V	Power	Power Supply of 5V
2	D22	Digital	Digital input 22/GPIO
3	D24	Digital	Digital input 24/GPIO
4	D26	Digital	Digital input 26/GPIO
5	D28	Digital	Digital input 28/GPIO
6	D30	Digital	Digital input 30/GPIO
7	D32	Digital	Digital input 32/GPIO
8	D34	Digital	Digital input 34/GPIO
9	D36	Digital	Digital input 36/GPIO
10	D38	Digital	Digital input 38/GPIO
11	D40	Digital	Digital input 40/GPIO
12	D42	Digital	Digital input 42/GPIO
13	D44	Digital	Digital input 44/GPIO
14	D46	Digital	Digital input 46/GPIO
15	D48	Digital	Digital input 48/GPIO
16	D50	Digital	Digital input 50/GPIO
17	D52	Digital	Digital input 52/GPIO
18	GND	Power	Ground

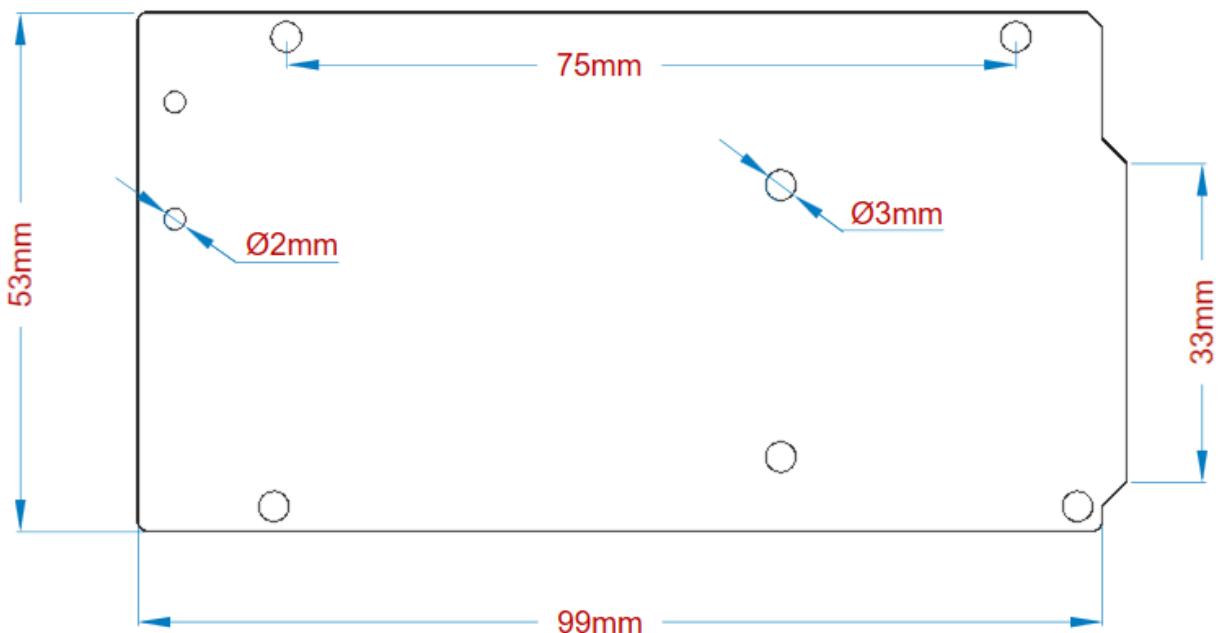


5.6 Digital Pins D22 - D53 RHS

Pin	Function	Type	Description
1	+5V	Power	Power Supply of 5V
2	D23	Digital	Digital input 23/GPIO
3	D25	Digital	Digital input 25/GPIO
4	D27	Digital	Digital input 27/GPIO
5	D29	Digital	Digital input 29/GPIO
6	D31	Digital	Digital input 31/GPIO
7	D33	Digital	Digital input 33/GPIO
8	D35	Digital	Digital input 35/GPIO
9	D37	Digital	Digital input 37/GPIO
10	D39	Digital	Digital input 39/GPIO
11	D41	Digital	Digital input 41/GPIO
12	D43	Digital	Digital input 43/GPIO
13	D45	Digital	Digital input 45/GPIO
14	D47	Digital	Digital input 47/GPIO
15	D49	Digital	Digital input 49/GPIO
16	D51	Digital	Digital input 51/GPIO
17	D53	Digital	Digital input 53/GPIO
18	GND	Power	Ground

6 Mechanical Information

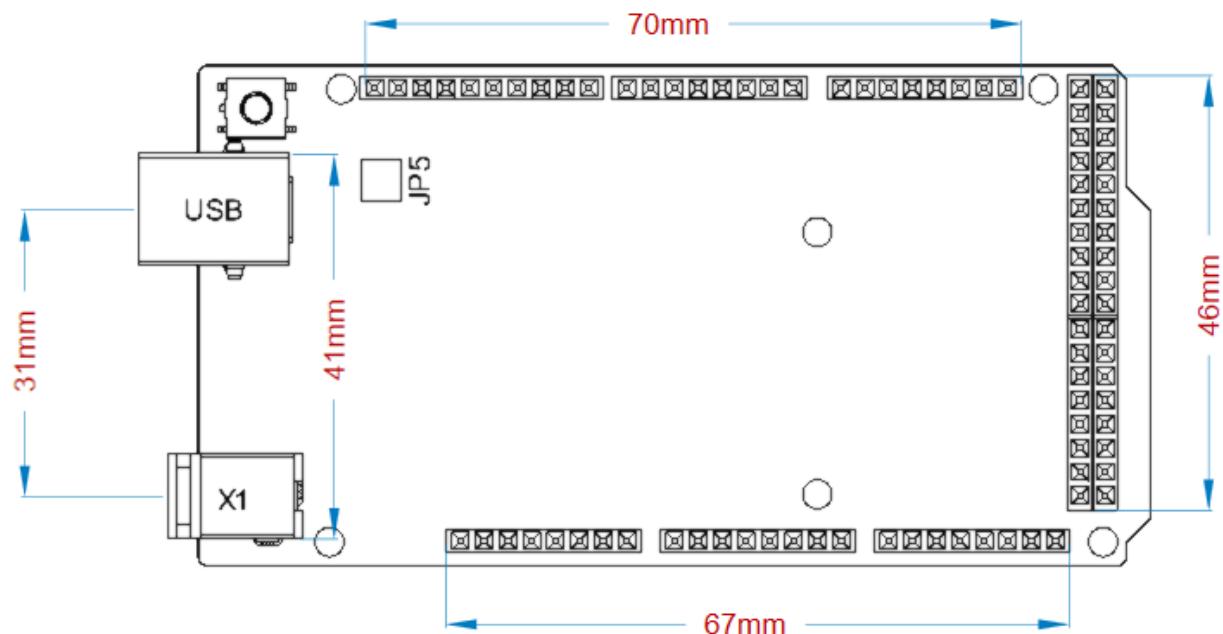
6.1 Board Outline





Arduino Mega Outline

6.2 Board Mount Holes



Arduino Mega Mount Holes

Certifications

7 Declaration of Conformity CE DoC (EU)

We declare under our sole responsibility that the products above are in conformity with the essential requirements of the following EU Directives and therefore qualify for free movement within markets comprising the European Union (EU) and European Economic Area (EEA).



8 Declaration of Conformity to EU RoHS & REACH 211

01/19/2021

Arduino boards are in compliance with RoHS 2 Directive 2011/65/EU of the European Parliament and RoHS 3 Directive 2015/863/EU of the Council of 4 June 2015 on the restriction of the use of certain hazardous substances in electrical and electronic equipment.

Substance	Maximum Limit (ppm)
Lead (Pb)	1000
Cadmium (Cd)	100
Mercury (Hg)	1000
Hexavalent Chromium (Cr6+)	1000
Poly Brominated Biphenyls (PBB)	1000
Poly Brominated Diphenyl ethers (PBDE)	1000
Bis(2-Ethylhexyl) phthalate (DEHP)	1000
Benzyl butyl phthalate (BBP)	1000
Dibutyl phthalate (DBP)	1000
Diisobutyl phthalate (DIBP)	1000

Exemptions : No exemptions are claimed.

Arduino Boards are fully compliant with the related requirements of European Union Regulation (EC) 1907 /2006 concerning the Registration, Evaluation, Authorization and Restriction of Chemicals (REACH). We declare none of the SVHCs (<https://echa.europa.eu/web/guest/candidate-list-table>), the Candidate List of Substances of Very High Concern for authorization currently released by ECHA, is present in all products (and also package) in quantities totaling in a concentration equal or above 0.1%. To the best of our knowledge, we also declare that our products do not contain any of the substances listed on the "Authorization List" (Annex XIV of the REACH regulations) and Substances of Very High Concern (SVHC) in any significant amounts as specified by the Annex XVII of Candidate list published by ECHA (European Chemical Agency) 1907 /2006/EC.



9 Conflict Minerals Declaration

As a global supplier of electronic and electrical components, Arduino is aware of our obligations with regards to laws and regulations regarding Conflict Minerals, specifically the Dodd-Frank Wall Street Reform and Consumer Protection Act, Section 1502. Arduino does not directly source or process conflict minerals such as Tin, Tantalum, Tungsten, or Gold. Conflict minerals are contained in our products in the form of solder, or as a component in metal alloys. As part of our reasonable due diligence Arduino has contacted component suppliers within our supply chain to verify their continued compliance with the regulations. Based on the information received thus far we declare that our products contain Conflict Minerals sourced from conflict-free areas.

10 FCC Caution

Any Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions:

- (1) This device may not cause harmful interference
- (2) this device must accept any interference received, including interference that may cause undesired operation.

FCC RF Radiation Exposure Statement:

1. This Transmitter must not be co-located or operating in conjunction with any other antenna or transmitter.
2. This equipment complies with RF radiation exposure limits set forth for an uncontrolled environment.
3. This equipment should be installed and operated with minimum distance 20cm between the radiator & your body.

English: User manuals for licence-exempt radio apparatus shall contain the following or equivalent notice in a conspicuous location in the user manual or alternatively on the device or both. This device complies with Industry Canada licence-exempt RSS standard(s). Operation is subject to the following two conditions:

- (1) this device may not cause interference
- (2) this device must accept any interference, including interference that may cause undesired operation of the device.

French: Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes :

- (1) l'appareil ne doit pas produire de brouillage
- (2) l'utilisateur de l'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

IC SAR Warning:

English This equipment should be installed and operated with minimum distance 20 cm between the radiator and your body.



French: Lors de l' installation et de l' exploitation de ce dispositif, la distance entre le radiateur et le corps est d 'au moins 20 cm.

Important: The operating temperature of the EUT can't exceed 85°C and shouldn't be lower than -40°C.

Hereby, Arduino S.r.l. declares that this product is in compliance with essential requirements and other relevant provisions of Directive 2014/53/EU. This product is allowed to be used in all EU member states.

11 Company Information

Company name	Arduino S.r.l.
Company Address	Arduino SRL, Via Andrea Appiani 25, 20900 Monza MB, Italy

12 Reference Documentation

Ref	Link
Arduino IDE (Desktop)	https://www.arduino.cc/en/Main/Software
Arduino IDE (Cloud)	https://create.arduino.cc/editor
Cloud IDE Getting Started	https://create.arduino.cc/projecthub/Arduino_Genuino/getting-started-with-arduino-web-editor-4b3e4a
Arduino Pro Website	https://www.arduino.cc/pro
Project Hub	https://create.arduino.cc/projecthub?by=part&part_id=11332&sort=trending
Library Reference	https://www.arduino.cc/reference/en/libraries/
Online Store	https://store.arduino.cc/

13 Revision History

Date	Revision	Changes
29/09/2020	1	First Release

3 TB6600 DATA-SHEET

TB6600 Stepper Motor Driver



< BULK >

3D

Safety Statement

The author of this document is not liable or responsible for any accidents, injuries, equipment damage, property damage, loss of money or loss of time resulting from improper use of electrical or mechanical or software products.

Assembling electrical CNC machine components like power supplies, motors, drivers or other electrical components involves dealing with high voltage AC (alternating current) or DC (direct current) which can be extremely dangerous and needs high attention to detail, experience, knowledge of software, electricity and electro-mechanics or mechanics.

BEFORE MAKING ANY CONNECTIONS OR DISCONNECTIONS POWER MUST BE REMOVED FROM THE DEVICE AND THE CONTROLLER. FAILURE TO DO SO WILL VOID ANY AND ALL WARRANTIES.

Introduction

The TB6600 Stepper Motor Driver is a professional two-phase stepper motor driver. It supports speed and direction control. You can set its micro step and output current with 6 DIP switches. There are 7 kinds of micro steps (1, 2 / A, 2 / B, 4, 8, 16, 32) and 8 kinds of current control (0.5A, 1A, 1.5A, 2A, 2.5A, 2.8A, 3.0A, 3.5A) in all. All signal terminals adopt high-speed optocoupler isolation, enhancing its anti-high-frequency interference ability.

Features

- Supports 8 levels of current control
- Supports 7 levels of micro step adjustment
- Interfaces adopt high-speed optocoupler isolation
- Automatic semi-flow to reduce heat
- Large area heat sink
- Anti-high-frequency interference ability
- Input anti-reverse protection
- Overheat, over current and short circuit protection

Electrical Specification:

- Input Current: 0-5.0A
- Output Current: 0.5-4.0A
- Power (MAX): 160W
- Micro Step: 1, 2/A, 2/B, 4, 8, 16, 32
- Temperature: -10~45°C
- Humidity: No Condensation
- Weight: 0.2 kg
- Dimensions: 96x56x33 mm

Application

Suitable for a variety of small and medium sized automation equipment and instruments, such as: engraving machine, marking machine, cutting machine, laser typesetting, plotters, CNC machine tools, handling the devices.

INPUT & OUTPUT:

Signal Input:

- PUL+ Pulse +
- PUL- Pulse -
- DIR+ Direction +
- DIR- Direction -
- EN+ Off-line Control Enable +
- EN- Off-line Control Enable -

Motor Machine Winding:

- A+ Stepper motor A+
- A- Stepper motor A-
- B+ Stepper motor B+
- B- Stepper motor B-

Power Supply:

- VCC VCC (DC9-42V)
- GND GND

Wiring instructions

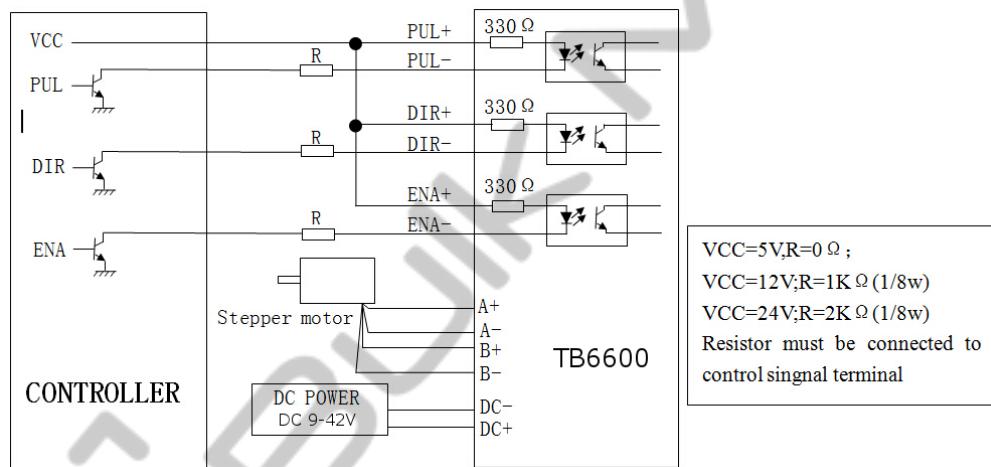
There are three input signals:

1. Step pulse signal PUL +, PUL-;
2. Direction signal DIR +, DIR-;
3. Enable signal EN +, EN-.

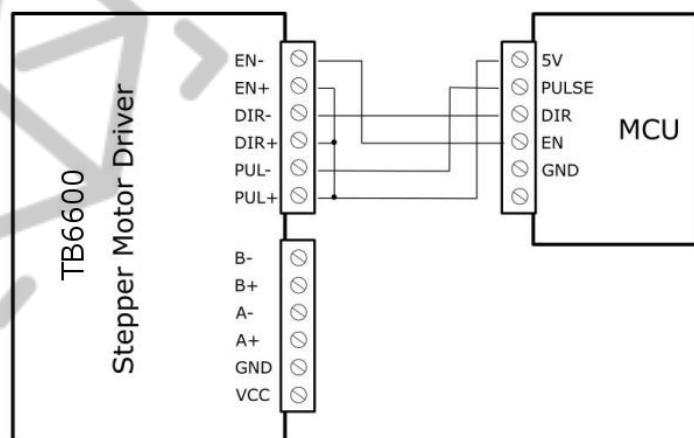
The driver supports common-cathode and common-anode circuit, you can select one according to your requirements.

Common-Anode Connection:

Connect PUL +, DIR + and EN + to the power supply of the control system. If the power supply is +5V, it can be directly connected. If the power supply is more than +5V, a current limiting resistor R must be added externally to ensure that the controller pin can output 8 - 15mA current to drive the internal optocoupler.

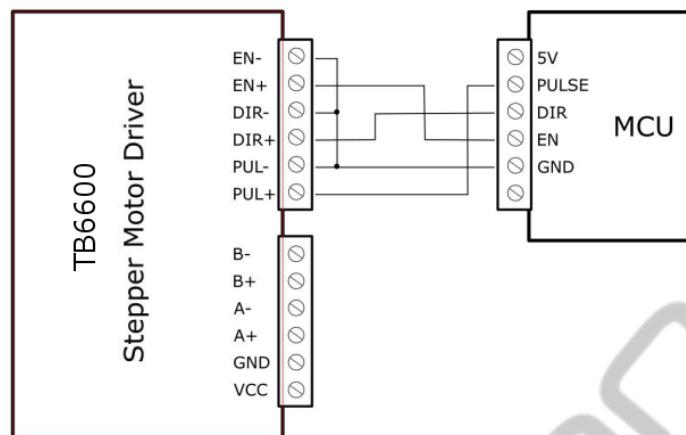


Pulse signal connects to PUL-; direction signal connects to Dir- ; Enable signal connects to EN-. As shown below:



Common-Cathode Connection:

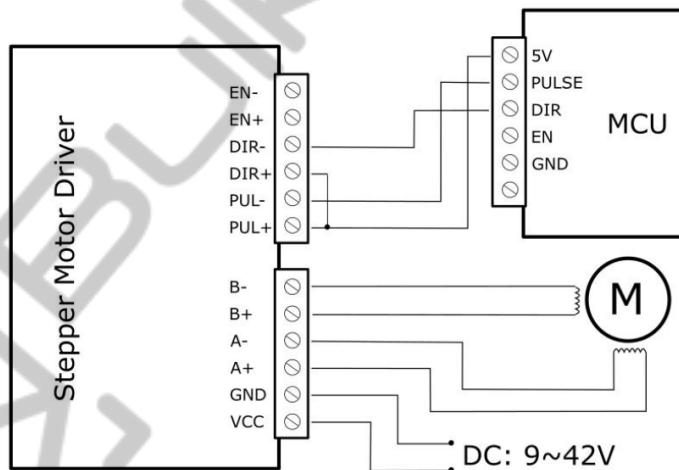
Connect PUL-, DIR- and EN- to the ground terminal of the control system. Pulse signal connects to PUL+; direction signal connects to Dir+ ; Enable signal connects to EN+. As shown below:



Note: When “EN” is in the valid state, the motor is in a free state (Off-line mode). In this mode, you can adjust the motor shaft position manually. When “EN” is in the invalid state, the motor will be in an automatic control mode.

Microcontroller Connection

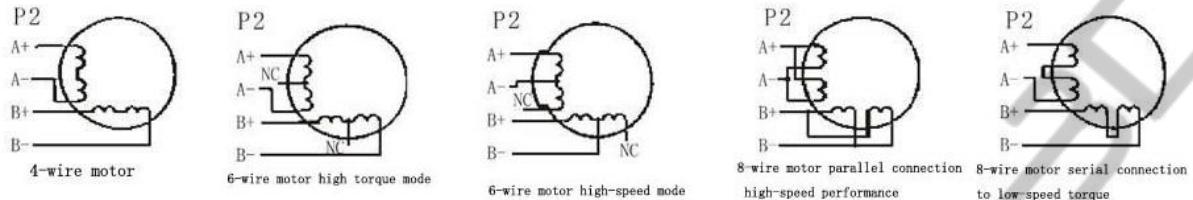
Below is a common-anode connection (“EN” not connected).



Note: Always disconnect the power when making connections and ensure the power polar is correct. Or it will damage the controller.

Stepper Motor Wiring

Two-phase 4-wire, 6-wire, 8-wire motor wiring, as shown below:



DIP Switch Settings

Micro-Step Setting

The following table shows the TB6600 Driver Micro step settings. The first 3 DIP switches are used to set the micro steps.

Step Angle = Motor Step Angle / Micro Step

E.g. A stepper motor with a 1.8° step angle, the final step angle under "Micro step 4" will be $1.8^\circ/4=0.45^\circ$

Micro Step	Pulse/Rev	S1	S2	S3
NC	NC	ON	ON	ON
1	200	ON	ON	OFF
2/A	400	ON	OFF	ON
2/B	400	OFF	ON	ON
4	800	ON	OFF	OFF
8	1600	OFF	ON	OFF
16	3200	OFF	OFF	ON
32	6400	OFF	OFF	OFF

Current Control Setting

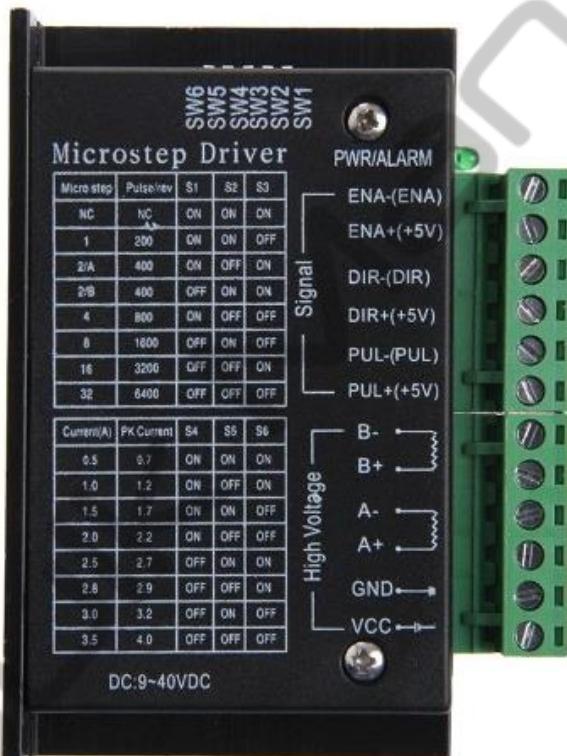
Current (A)	S4	S5	S6
0.5	ON	ON	ON
1.0	ON	OFF	ON
1.5	ON	ON	OFF
2.0	ON	OFF	OFF
2.5	OFF	ON	ON
2.8	OFF	OFF	ON
3.0	OFF	ON	OFF
3.5	OFF	OFF	OFF

Enable / Off-Line Function

If you turn on the Enable / Off-line function, the motor will enter a free state, you will be able to turn the motor shaft freely and any pulse signal will be ignored. If you turn it off, the driver will be in automatic control mode.

Note: Generally, EN terminal is not connected.

Images



Credits

https://www.dfrobot.com/wiki/index.php/TB6600_Stepper_Motor_Driver_SKU:_DRI0043

4 LM7805 IC DATA-SHEET

MC78XX/LM78XX/MC78XXA

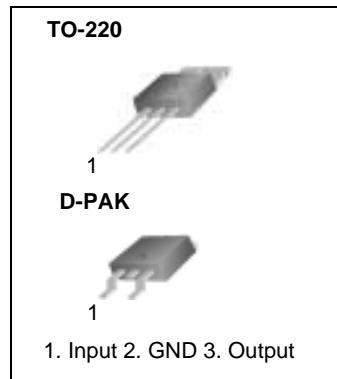
3-Terminal 1A Positive Voltage Regulator

Features

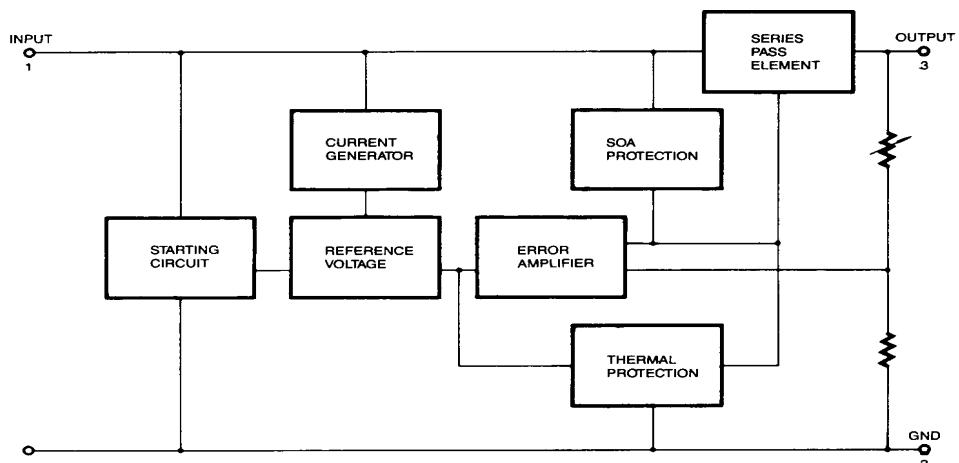
- Output Current up to 1A
- Output Voltages of 5, 6, 8, 9, 10, 12, 15, 18, 24V
- Thermal Overload Protection
- Short Circuit Protection
- Output Transistor Safe Operating Area Protection

Description

The MC78XX/LM78XX/MC78XXA series of three terminal positive regulators are available in the TO-220/D-PAK package and with several fixed output voltages, making them useful in a wide range of applications. Each type employs internal current limiting, thermal shut down and safe operating area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.



Internal Block Diagram



Absolute Maximum Ratings

Parameter	Symbol	Value	Unit
Input Voltage (for $V_O = 5V$ to $18V$) (for $V_O = 24V$)	V_I	35	V
	V_I	40	V
Thermal Resistance Junction-Cases (TO-220)	$R_{\theta JC}$	5	$^{\circ}C/W$
Thermal Resistance Junction-Air (TO-220)	$R_{\theta JA}$	65	$^{\circ}C/W$
Operating Temperature Range	T_{OPR}	0 ~ +125	$^{\circ}C$
Storage Temperature Range	T_{STG}	-65 ~ +150	$^{\circ}C$

Electrical Characteristics (MC7805/LM7805)

(Refer to test circuit, $0^{\circ}C < T_J < 125^{\circ}C$, $I_O = 500mA$, $V_I = 10V$, $C_L = 0.33\mu F$, $C_O = 0.1\mu F$, unless otherwise specified)

Parameter	Symbol	Conditions	MC7805/LM7805			Unit	
			Min.	Typ.	Max.		
Output Voltage	V_O	$T_J = +25^{\circ}C$	4.8	5.0	5.2	V	
		$5.0mA \leq I_O \leq 1.0A$, $P_O \leq 15W$ $V_I = 7V$ to $20V$	4.75	5.0	5.25		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}C$	$V_O = 7V$ to $25V$	-	4.0	100	mV
			$V_I = 8V$ to $12V$	-	1.6	50	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}C$	$I_O = 5.0mA$ to $1.5A$	-	9	100	mV
			$I_O = 250mA$ to $750mA$	-	4	50	
Quiescent Current	I_Q	$T_J = +25^{\circ}C$	-	5.0	8.0	mA	
Quiescent Current Change	ΔI_Q	$I_O = 5mA$ to $1.0A$	-	0.03	0.5	mA	
		$V_I = 7V$ to $25V$	-	0.3	1.3		
Output Voltage Drift	$\Delta V_O/\Delta T$	$I_O = 5mA$	-	-0.8	-	$mV/{}^{\circ}C$	
Output Noise Voltage	V_N	$f = 10Hz$ to $100KHz$, $T_A = +25^{\circ}C$	-	42	-	$\mu V/V_o$	
Ripple Rejection	RR	$f = 120Hz$ $V_O = 8V$ to $18V$	62	73	-	dB	
Dropout Voltage	V_{Drop}	$I_O = 1A$, $T_J = +25^{\circ}C$	-	2	-	V	
Output Resistance	r_O	$f = 1KHz$	-	15	-	$m\Omega$	
Short Circuit Current	I_{SC}	$V_I = 35V$, $T_A = +25^{\circ}C$	-	230	-	mA	
Peak Current	I_{PK}	$T_J = +25^{\circ}C$	-	2.2	-	A	

Note:

1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7806)

(Refer to test circuit , $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$, $I_O = 500\text{mA}$, $V_I = 11\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	MC7806			Unit	
			Min.	Typ.	Max.		
Output Voltage	V_O	$T_J = +25^{\circ}\text{C}$	5.75	6.0	6.25	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$, $P_O \leq 15\text{W}$ $V_I = 8.0\text{V}$ to 21V	5.7	6.0	6.3		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 8\text{V}$ to 25V	-	5	120	mV
			$V_I = 9\text{V}$ to 13V	-	1.5	60	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA}$ to 1.5A	-	9	120	mV
			$I_O = 250\text{mA}$ to 750A	-	3	60	
Quiescent Current	I_Q	$T_J = +25^{\circ}\text{C}$	-	5.0	8.0	mA	
Quiescent Current Change	ΔI_Q	$I_O = 5\text{mA}$ to 1A	-	-	0.5	mA	
		$V_I = 8\text{V}$ to 25V	-	-	1.3		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5\text{mA}$	-	-0.8	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	V_N	$f = 10\text{Hz}$ to 100KHz , $T_A = +25^{\circ}\text{C}$	-	45	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 9\text{V}$ to 19V	59	75	-	dB	
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}$, $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	r_O	$f = 1\text{KHz}$	-	19	-	$\text{m}\Omega$	
Short Circuit Current	I_{SC}	$V_I = 35\text{V}$, $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	I_{PK}	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7808)

(Refer to test circuit , $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$, $I_O = 500\text{mA}$, $V_I = 14\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	MC7808			Unit	
			Min.	Typ.	Max.		
Output Voltage	V_O	$T_J = +25^{\circ}\text{C}$	7.7	8.0	8.3	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$, $P_O \leq 15\text{W}$ $V_I = 10.5\text{V}$ to 23V	7.6	8.0	8.4		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 10.5\text{V}$ to 25V	-	5.0	160	mV
			$V_I = 11.5\text{V}$ to 17V	-	2.0	80	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5.0\text{mA}$ to 1.5A	-	10	160	mV
			$I_O = 250\text{mA}$ to 750mA	-	5.0	80	
Quiescent Current	I_Q	$T_J = +25^{\circ}\text{C}$	-	5.0	8.0	mA	
Quiescent Current Change	ΔI_Q	$I_O = 5\text{mA}$ to 1.0A	-	0.05	0.5	mA	
		$V_I = 10.5\text{A}$ to 25V	-	0.5	1.0		
Output Voltage Drift	$\Delta V_O/\Delta T$	$I_O = 5\text{mA}$	-	-0.8	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	V_N	$f = 10\text{Hz}$ to 100KHz , $T_A = +25^{\circ}\text{C}$	-	52	-	$\mu\text{V}/\text{V}_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$, $V_I = 11.5\text{V}$ to 21.5V	56	73	-	dB	
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}$, $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	r_O	$f = 1\text{KHz}$	-	17	-	$\text{m}\Omega$	
Short Circuit Current	I_{SC}	$V_I = 35\text{V}$, $T_A = +25^{\circ}\text{C}$	-	230	-	mA	
Peak Current	I_{PK}	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7809)

(Refer to test circuit, $0^\circ\text{C} < T_J < 125^\circ\text{C}$, $I_O = 500\text{mA}$, $V_I = 15\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	MC7809			Unit	
			Min.	Typ.	Max.		
Output Voltage	V_O	$T_J = +25^\circ\text{C}$	8.65	9	9.35	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$, $P_O \leq 15\text{W}$ $V_I = 11.5\text{V}$ to 24V	8.6	9	9.4		
Line Regulation (Note1)	Regline	$T_J = +25^\circ\text{C}$	$V_I = 11.5\text{V}$ to 25V	-	6	180	mV
			$V_I = 12\text{V}$ to 17V	-	2	90	
Load Regulation (Note1)	Regload	$T_J = +25^\circ\text{C}$	$I_O = 5\text{mA}$ to 1.5A	-	12	180	mV
			$I_O = 250\text{mA}$ to 750mA	-	4	90	
Quiescent Current	I_Q	$T_J = +25^\circ\text{C}$	-	5.0	8.0	mA	
Quiescent Current Change	ΔI_Q	$I_O = 5\text{mA}$ to 1.0A	-	-	0.5	mA	
		$V_I = 11.5\text{V}$ to 26V	-	-	1.3		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5\text{mA}$	-	-1	-	mV/°C	
Output Noise Voltage	V_N	$f = 10\text{Hz}$ to 100KHz , $T_A = +25^\circ\text{C}$	-	58	-	$\mu\text{V}/\text{V}_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 13\text{V}$ to 23V	56	71	-	dB	
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}$, $T_J = +25^\circ\text{C}$	-	2	-	V	
Output Resistance	r_O	$f = 1\text{KHz}$	-	17	-	$\text{m}\Omega$	
Short Circuit Current	I_{SC}	$V_I = 35\text{V}$, $T_A = +25^\circ\text{C}$	-	250	-	mA	
Peak Current	I_{PK}	$T_J = +25^\circ\text{C}$	-	2.2	-	A	

Note:

1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7810)

(Refer to test circuit , $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$, $I_O = 500\text{mA}$, $V_I = 16\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	MC7810			Unit	
			Min.	Typ.	Max.		
Output Voltage	V_O	$T_J = +25^{\circ}\text{C}$	9.6	10	10.4	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$, $P_O \leq 15\text{W}$ $V_I = 12.5\text{V}$ to 25V	9.5	10	10.5		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 12.5\text{V}$ to 25V	-	10	200	mV
			$V_I = 13\text{V}$ to 25V	-	3	100	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA}$ to 1.5A	-	12	200	mV
			$I_O = 250\text{mA}$ to 750mA	-	4	400	
Quiescent Current	I_Q	$T_J = +25^{\circ}\text{C}$	-	5.1	8.0	mA	
Quiescent Current Change	ΔI_Q	$I_O = 5\text{mA}$ to 1.0A	-	-	0.5	mA	
		$V_I = 12.5\text{V}$ to 29V	-	-	1.0		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5\text{mA}$	-	-1	-	$\text{mV}/^{\circ}\text{C}$	
Output Noise Voltage	V_N	$f = 10\text{Hz}$ to 100KHz , $T_A = +25^{\circ}\text{C}$	-	58	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 13\text{V}$ to 23V	56	71	-	dB	
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}$, $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	r_O	$f = 1\text{KHz}$	-	17	-	$\text{m}\Omega$	
Short Circuit Current	I_{SC}	$V_I = 35\text{V}$, $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	I_{PK}	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7812)

(Refer to test circuit , $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$, $I_O = 500\text{mA}$, $V_I = 19\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	MC7812			Unit	
			Min.	Typ.	Max.		
Output Voltage	V_O	$T_J = +25^{\circ}\text{C}$	11.5	12	12.5	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$, $P_O \leq 15\text{W}$ $V_I = 14.5\text{V}$ to 27V	11.4	12	12.6		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 14.5\text{V}$ to 30V	-	10	240	mV
			$V_I = 16\text{V}$ to 22V	-	3.0	120	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA}$ to 1.5A	-	11	240	mV
			$I_O = 250\text{mA}$ to 750mA	-	5.0	120	
Quiescent Current	I_Q	$T_J = +25^{\circ}\text{C}$	-	5.1	8.0	mA	
Quiescent Current Change	ΔI_Q	$I_Q = 5\text{mA}$ to 1.0A	-	0.1	0.5	mA	
		$V_I = 14.5\text{V}$ to 30V	-	0.5	1.0		
Output Voltage Drift	$\Delta V_O/\Delta T$	$I_O = 5\text{mA}$	-	-1	-	mV/°C	
Output Noise Voltage	V_N	$f = 10\text{Hz}$ to 100KHz , $T_A = +25^{\circ}\text{C}$	-	76	-	µV/V _O	
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 15\text{V}$ to 25V	55	71	-	dB	
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}$, $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	r_O	$f = 1\text{KHz}$	-	18	-	mΩ	
Short Circuit Current	I_{SC}	$V_I = 35\text{V}$, $T_A = +25^{\circ}\text{C}$	-	230	-	mA	
Peak Current	I_{PK}	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7815)

(Refer to test circuit , $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$, $I_O = 500\text{mA}$, $V_I = 23\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	MC7815			Unit	
			Min.	Typ.	Max.		
Output Voltage	V_O	$T_J = +25^{\circ}\text{C}$	14.4	15	15.6	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$, $P_O \leq 15\text{W}$ $V_I = 17.5\text{V}$ to 30V	14.25	15	15.75		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 17.5\text{V}$ to 30V	-	11	300	mV
			$V_I = 20\text{V}$ to 26V	-	3	150	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA}$ to 1.5A	-	12	300	mV
			$I_O = 250\text{mA}$ to 750mA	-	4	150	
Quiescent Current	I_Q	$T_J = +25^{\circ}\text{C}$	-	5.2	8.0	mA	
Quiescent Current Change	ΔI_Q	$I_O = 5\text{mA}$ to 1.0A	-	-	0.5	mA	
		$V_I = 17.5\text{V}$ to 30V	-	-	1.0		
Output Voltage Drift	$\Delta V_O/\Delta T$	$I_O = 5\text{mA}$	-	-1	-	$\text{mV}/^{\circ}\text{C}$	
Output Noise Voltage	V_N	$f = 10\text{Hz}$ to 100KHz , $T_A = +25^{\circ}\text{C}$	-	90	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 18.5\text{V}$ to 28.5V	54	70	-	dB	
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}$, $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	r_O	$f = 1\text{KHz}$	-	19	-	$\text{m}\Omega$	
Short Circuit Current	I_{SC}	$V_I = 35\text{V}$, $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	I_{PK}	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7818)

(Refer to test circuit , $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$, $I_O = 500\text{mA}$, $V_I = 27\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	MC7818			Unit	
			Min.	Typ.	Max.		
Output Voltage	V_O	$T_J = +25^{\circ}\text{C}$	17.3	18	18.7	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$, $P_O \leq 15\text{W}$ $V_I = 21\text{V}$ to 33V	17.1	18	18.9		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 21\text{V}$ to 33V	-	15	360	mV
			$V_I = 24\text{V}$ to 30V	-	5	180	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA}$ to 1.5A	-	15	360	mV
			$I_O = 250\text{mA}$ to 750mA	-	5.0	180	
Quiescent Current	I_Q	$T_J = +25^{\circ}\text{C}$	-	5.2	8.0	mA	
Quiescent Current Change	ΔI_Q	$I_O = 5\text{mA}$ to 1.0A	-	-	0.5	mA	
		$V_I = 21\text{V}$ to 33V	-	-	1		
Output Voltage Drift	$\Delta V_O/\Delta T$	$I_O = 5\text{mA}$	-	-1	-	mV/ $^{\circ}\text{C}$	
Output Noise Voltage	V_N	$f = 10\text{Hz}$ to 100KHz , $T_A = +25^{\circ}\text{C}$	-	110	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 22\text{V}$ to 32V	53	69	-	dB	
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}$, $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	r_O	$f = 1\text{KHz}$	-	22	-	$\text{m}\Omega$	
Short Circuit Current	I_{SC}	$V_I = 35\text{V}$, $T_A = +25^{\circ}\text{C}$	-	250	-	mA	
Peak Current	I_{PK}	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7824)

(Refer to test circuit , $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$, $I_O = 500\text{mA}$, $V_I = 33\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	MC7824			Unit	
			Min.	Typ.	Max.		
Output Voltage	V_O	$T_J = +25^{\circ}\text{C}$	23	24	25	V	
		$5.0\text{mA} \leq I_O \leq 1.0\text{A}$, $P_O \leq 15\text{W}$ $V_I = 27\text{V}$ to 38V	22.8	24	25.25		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}\text{C}$	$V_I = 27\text{V}$ to 38V	-	17	480	mV
			$V_I = 30\text{V}$ to 36V	-	6	240	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$	$I_O = 5\text{mA}$ to 1.5A	-	15	480	mV
			$I_O = 250\text{mA}$ to 750mA	-	5.0	240	
Quiescent Current	I_Q	$T_J = +25^{\circ}\text{C}$	-	5.2	8.0	mA	
Quiescent Current Change	ΔI_Q	$I_O = 5\text{mA}$ to 1.0A	-	0.1	0.5	mA	
		$V_I = 27\text{V}$ to 38V	-	0.5	1		
Output Voltage Drift	$\Delta V_O/\Delta T$	$I_O = 5\text{mA}$	-	-1.5	-	mV/°C	
Output Noise Voltage	V_N	$f = 10\text{Hz}$ to 100KHz , $T_A = +25^{\circ}\text{C}$	-	60	-	$\mu\text{V}/V_O$	
Ripple Rejection	RR	$f = 120\text{Hz}$ $V_I = 28\text{V}$ to 38V	50	67	-	dB	
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}$, $T_J = +25^{\circ}\text{C}$	-	2	-	V	
Output Resistance	r_O	$f = 1\text{KHz}$	-	28	-	$\text{m}\Omega$	
Short Circuit Current	I_{SC}	$V_I = 35\text{V}$, $T_A = +25^{\circ}\text{C}$	-	230	-	mA	
Peak Current	I_{PK}	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A	

Note:

1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7805A)

(Refer to the test circuits. $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$, $I_O = 1\text{A}$, $V_I = 10\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Output Voltage	V_O	$T_J = +25^{\circ}\text{C}$	4.9	5	5.1	V
		$I_O = 5\text{mA to } 1\text{A}, P_O \leq 15\text{W}$ $V_I = 7.5\text{V to } 20\text{V}$	4.8	5	5.2	
Line Regulation (Note1)	Regline	$V_I = 7.5\text{V to } 25\text{V}$ $I_O = 500\text{mA}$	-	5	50	mV
		$V_I = 8\text{V to } 12\text{V}$	-	3	50	
		$T_J = +25^{\circ}\text{C}$ $V_I = 7.3\text{V to } 20\text{V}$	-	5	50	
Load Regulation (Note1)	Regload	$V_I = 8\text{V to } 12\text{V}$	-	1.5	25	mV
		$T_J = +25^{\circ}\text{C}$ $I_O = 5\text{mA to } 1.5\text{A}$	-	9	100	
		$I_O = 5\text{mA to } 1\text{A}$	-	9	100	
Quiescent Current	I_Q	$I_O = 250\text{mA to } 750\text{mA}$	-	4	50	mA
		$T_J = +25^{\circ}\text{C}$	-	5.0	6	
		$I_O = 5\text{mA to } 1\text{A}$	-	-	0.5	
Quiescent Current Change	ΔI_Q	$V_I = 8\text{ V to } 25\text{V}, I_O = 500\text{mA}$	-	-	0.8	mA
		$V_I = 7.5\text{V to } 20\text{V}, T_J = +25^{\circ}\text{C}$	-	-	0.8	
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-0.8	-	mV/°C
Output Noise Voltage	V_N	$f = 10\text{Hz to } 100\text{KHz}$ $T_A = +25^{\circ}\text{C}$	-	10	-	µV/V _O
Ripple Rejection	RR	$f = 120\text{Hz}, I_O = 500\text{mA}$ $V_I = 8\text{V to } 18\text{V}$	-	68	-	dB
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}, T_J = +25^{\circ}\text{C}$	-	2	-	V
Output Resistance	r_O	$f = 1\text{KHz}$	-	17	-	mΩ
Short Circuit Current	I_{SC}	$V_I = 35\text{V}, T_A = +25^{\circ}\text{C}$	-	250	-	mA
Peak Current	I_{PK}	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A

Note:

1. Load and line regulation are specified at constant junction temperature. Change in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7806A)

(Refer to the test circuits. $0^\circ\text{C} < T_J < 125^\circ\text{C}$, $I_O = 1\text{A}$, $V_I = 11\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Output Voltage	V_O	$T_J = +25^\circ\text{C}$	5.58	6	6.12	V
		$I_O = 5\text{mA to } 1\text{A}$, $P_O \leq 15\text{W}$ $V_I = 8.6\text{V to } 21\text{V}$	5.76	6	6.24	
Line Regulation (Note1)	Regline	$V_I = 8.6\text{V to } 25\text{V}$ $I_O = 500\text{mA}$	-	5	60	mV
		$V_I = 9\text{V to } 13\text{V}$	-	3	60	
		$T_J = +25^\circ\text{C}$	$V_I = 8.3\text{V to } 21\text{V}$	-	5	
			$V_I = 9\text{V to } 13\text{V}$	-	1.5	
Load Regulation (Note1)	Regload	$T_J = +25^\circ\text{C}$ $I_O = 5\text{mA to } 1.5\text{A}$	-	9	100	mV
		$I_O = 5\text{mA to } 1\text{A}$	-	4	100	
		$I_O = 250\text{mA to } 750\text{mA}$	-	5.0	50	
Quiescent Current	I_Q	$T_J = +25^\circ\text{C}$	-	4.3	6	mA
Quiescent Current Change	ΔI_Q	$I_O = 5\text{mA to } 1\text{A}$	-	-	0.5	mA
		$V_I = 9\text{V to } 25\text{V}$, $I_O = 500\text{mA}$	-	-	0.8	
		$V_I = 8.5\text{V to } 21\text{V}$, $T_J = +25^\circ\text{C}$	-	-	0.8	
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-0.8	-	mV/°C
Output Noise Voltage	V_N	$f = 10\text{Hz to } 100\text{KHz}$ $T_A = +25^\circ\text{C}$	-	10	-	µV/V _O
Ripple Rejection	RR	$f = 120\text{Hz}$, $I_O = 500\text{mA}$ $V_I = 9\text{V to } 19\text{V}$	-	65	-	dB
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}$, $T_J = +25^\circ\text{C}$	-	2	-	V
Output Resistance	r_O	$f = 1\text{KHz}$	-	17	-	mΩ
Short Circuit Current	I_{SC}	$V_I = 35\text{V}$, $T_A = +25^\circ\text{C}$	-	250	-	mA
Peak Current	I_{PK}	$T_J = +25^\circ\text{C}$	-	2.2	-	A

Note:

1. Load and line regulation are specified at constant junction temperature. Change in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7808A)

(Refer to the test circuits. $0^\circ\text{C} < T_J < 125^\circ\text{C}$, $I_O = 1\text{A}$, $V_I = 14\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Output Voltage	V_O	$T_J = +25^\circ\text{C}$	7.84	8	8.16	V
		$I_O = 5\text{mA to } 1\text{A}, P_O \leq 15\text{W}$ $V_I = 10.6\text{V to } 23\text{V}$	7.7	8	8.3	
Line Regulation (Note1)	Regline	$V_I = 10.6\text{V to } 25\text{V}$ $I_O = 500\text{mA}$	-	6	80	mV
		$V_I = 11\text{V to } 17\text{V}$	-	3	80	
		$T_J = +25^\circ\text{C}$ $V_I = 10.4\text{V to } 23\text{V}$	-	6	80	
Load Regulation (Note1)	Regload	$V_I = 11\text{V to } 17\text{V}$	-	2	40	mV
		$T_J = +25^\circ\text{C}$ $I_O = 5\text{mA to } 1.5\text{A}$	-	12	100	
		$I_O = 5\text{mA to } 1\text{A}$	-	12	100	
Quiescent Current	I_Q	$I_O = 250\text{mA to } 750\text{mA}$	-	5	50	mA
		$T_J = +25^\circ\text{C}$	-	5.0	6	
		$I_O = 5\text{mA to } 1\text{A}$	-	-	0.5	
Quiescent Current Change	ΔI_Q	$V_I = 11\text{V to } 25\text{V}, I_O = 500\text{mA}$	-	-	0.8	mA
		$V_I = 10.6\text{V to } 23\text{V}, T_J = +25^\circ\text{C}$	-	-	0.8	
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-0.8	-	mV/°C
Output Noise Voltage	V_N	$f = 10\text{Hz to } 100\text{KHz}$ $T_A = +25^\circ\text{C}$	-	10	-	µV/V _O
Ripple Rejection	RR	$f = 120\text{Hz}, I_O = 500\text{mA}$ $V_I = 11.5\text{V to } 21.5\text{V}$	-	62	-	dB
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}, T_J = +25^\circ\text{C}$	-	2	-	V
Output Resistance	r_O	$f = 1\text{KHz}$	-	18	-	mΩ
Short Circuit Current	I_{SC}	$V_I = 35\text{V}, T_A = +25^\circ\text{C}$	-	250	-	mA
Peak Current	I_{PK}	$T_J = +25^\circ\text{C}$	-	2.2	-	A

Note:

1. Load and line regulation are specified at constant junction temperature. Change in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7809A)

(Refer to the test circuits. $0^\circ\text{C} < T_J < 125^\circ\text{C}$, $I_O = 1\text{A}$, $V_I = 15\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Output Voltage	V_O	$T_J = +25^\circ\text{C}$	8.82	9.0	9.18	V
		$I_O = 5\text{mA to } 1\text{A}, P_O \leq 15\text{W}$ $V_I = 11.2\text{V to } 24\text{V}$	8.65	9.0	9.35	
Line Regulation (Note1)	Regline	$V_I = 11.7\text{V to } 25\text{V}$ $I_O = 500\text{mA}$	-	6	90	mV
		$V_I = 12.5\text{V to } 19\text{V}$	-	4	45	
		$T_J = +25^\circ\text{C}$	$V_I = 11.5\text{V to } 24\text{V}$	-	6	90
			$V_I = 12.5\text{V to } 19\text{V}$	-	2	45
Load Regulation (Note1)	Regload	$T_J = +25^\circ\text{C}$ $I_O = 5\text{mA to } 1.0\text{A}$	-	12	100	mV
		$I_O = 5\text{mA to } 1.0\text{A}$	-	12	100	
		$I_O = 250\text{mA to } 750\text{mA}$	-	5	50	
Quiescent Current	I_Q	$T_J = +25^\circ\text{C}$	-	5.0	6.0	mA
Quiescent Current Change	ΔI_Q	$V_I = 11.7\text{V to } 25\text{V}, T_J = +25^\circ\text{C}$	-	-	0.8	mA
		$V_I = 12\text{V to } 25\text{V}, I_O = 500\text{mA}$	-	-	0.8	
		$I_O = 5\text{mA to } 1.0\text{A}$	-	-	0.5	
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-1.0	-	$\text{mV}/^\circ\text{C}$
Output Noise Voltage	V_N	$f = 10\text{Hz to } 100\text{KHz}$ $T_A = +25^\circ\text{C}$	-	10	-	$\mu\text{V}/V_O$
Ripple Rejection	RR	$f = 120\text{Hz}, I_O = 500\text{mA}$ $V_I = 12\text{V to } 22\text{V}$	-	62	-	dB
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}, T_J = +25^\circ\text{C}$	-	2.0	-	V
Output Resistance	r_O	$f = 1\text{KHz}$	-	17	-	$\text{m}\Omega$
Short Circuit Current	I_{SC}	$V_I = 35\text{V}, T_A = +25^\circ\text{C}$	-	250	-	mA
Peak Current	I_{PK}	$T_J = +25^\circ\text{C}$	-	2.2	-	A

Note:

1. Load and line regulation are specified at constant junction temperature. Change in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7810A)

(Refer to the test circuits. $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$, $I_O = 1\text{A}$, $V_I = 16\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Output Voltage	V_O	$T_J = +25^{\circ}\text{C}$	9.8	10	10.2	V
		$I_O = 5\text{mA to } 1\text{A}, P_O \leq 15\text{W}$ $V_I = 12.8\text{V to } 25\text{V}$	9.6	10	10.4	
Line Regulation (Note1)	Regline	$V_I = 12.8\text{V to } 26\text{V}$ $I_O = 500\text{mA}$	-	8	100	mV
		$V_I = 13\text{V to } 20\text{V}$	-	4	50	
		$T_J = +25^{\circ}\text{C}$	$V_I = 12.5\text{V to } 25\text{V}$	-	8	100
			$V_I = 13\text{V to } 20\text{V}$	-	3	50
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$ $I_O = 5\text{mA to } 1.5\text{A}$	-	12	100	mV
		$I_O = 5\text{mA to } 1.0\text{A}$	-	12	100	
		$I_O = 250\text{mA to } 750\text{mA}$	-	5	50	
Quiescent Current	I_Q	$T_J = +25^{\circ}\text{C}$	-	5.0	6.0	mA
Quiescent Current Change	ΔI_Q	$V_I = 13\text{V to } 26\text{V}, T_J = +25^{\circ}\text{C}$	-	-	0.5	mA
		$V_I = 12.8\text{V to } 25\text{V}, I_O = 500\text{mA}$	-	-	0.8	
		$I_O = 5\text{mA to } 1.0\text{A}$	-	-	0.5	
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-1.0	-	$\text{mV}/^{\circ}\text{C}$
Output Noise Voltage	V_N	$f = 10\text{Hz to } 100\text{KHz}$ $T_A = +25^{\circ}\text{C}$	-	10	-	$\mu\text{V}/\text{Vo}$
Ripple Rejection	RR	$f = 120\text{Hz}, I_O = 500\text{mA}$ $V_I = 14\text{V to } 24\text{V}$	-	62	-	dB
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}, T_J = +25^{\circ}\text{C}$	-	2.0	-	V
Output Resistance	r_O	$f = 1\text{KHz}$	-	17	-	$\text{m}\Omega$
Short Circuit Current	I_{SC}	$V_I = 35\text{V}, T_A = +25^{\circ}\text{C}$	-	250	-	mA
Peak Current	I_{PK}	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A

Note:

1. Load and line regulation are specified at constant junction temperature. Change in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7812A)

(Refer to the test circuits. $0^\circ\text{C} < T_J < 125^\circ\text{C}$, $I_O = 1\text{A}$, $V_I = 19\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Output Voltage	V_O	$T_J = +25^\circ\text{C}$	11.75	12	12.25	V
		$I_O = 5\text{mA to } 1\text{A}, P_O \leq 15\text{W}$ $V_I = 14.8\text{V to } 27\text{V}$	11.5	12	12.5	
Line Regulation (Note1)	Regline	$V_I = 14.8\text{V to } 30\text{V}$ $I_O = 500\text{mA}$	-	10	120	mV
		$V_I = 16\text{V to } 22\text{V}$	-	4	120	
		$T_J = +25^\circ\text{C}$	$V_I = 14.5\text{V to } 27\text{V}$	-	10	120
			$V_I = 16\text{V to } 22\text{V}$	-	3	60
Load Regulation (Note1)	Regload	$T_J = +25^\circ\text{C}$ $I_O = 5\text{mA to } 1.5\text{A}$	-	12	100	mV
		$I_O = 5\text{mA to } 1.0\text{A}$	-	12	100	
		$I_O = 250\text{mA to } 750\text{mA}$	-	5	50	
Quiescent Current	I_Q	$T_J = +25^\circ\text{C}$	-	5.1	6.0	mA
Quiescent Current Change	ΔI_Q	$V_I = 15\text{V to } 30\text{V}, T_J = +25^\circ\text{C}$	-		0.8	mA
		$V_I = 14\text{V to } 27\text{V}, I_O = 500\text{mA}$	-		0.8	
		$I_O = 5\text{mA to } 1.0\text{A}$	-		0.5	
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-1.0	-	mV/°C
Output Noise Voltage	V_N	$f = 10\text{Hz to } 100\text{KHz}$ $T_A = +25^\circ\text{C}$	-	10	-	µV/V _O
Ripple Rejection	RR	$f = 120\text{Hz}, I_O = 500\text{mA}$ $V_I = 14\text{V to } 24\text{V}$	-	60	-	dB
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}, T_J = +25^\circ\text{C}$	-	2.0	-	V
Output Resistance	r_O	$f = 1\text{KHz}$	-	18	-	mΩ
Short Circuit Current	I_{SC}	$V_I = 35\text{V}, T_A = +25^\circ\text{C}$	-	250	-	mA
Peak Current	I_{PK}	$T_J = +25^\circ\text{C}$	-	2.2	-	A

Note:

1. Load and line regulation are specified at constant junction temperature. Change in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7815A)

(Refer to the test circuits. $0^\circ\text{C} < T_J < 125^\circ\text{C}$, $I_O = 1\text{A}$, $V_I = 23\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Output Voltage	V_O	$T_J = +25^\circ\text{C}$	14.7	15	15.3	V
		$I_O = 5\text{mA}$ to 1A , $P_O \leq 15\text{W}$ $V_I = 17.7\text{V}$ to 30V	14.4	15	15.6	
Line Regulation (Note1)	Regline	$V_I = 17.9\text{V}$ to 30V $I_O = 500\text{mA}$	-	10	150	mV
		$V_I = 20\text{V}$ to 26V	-	5	150	
		$T_J = +25^\circ\text{C}$	$V_I = 17.5\text{V}$ to 30V	-	11	150
			$V_I = 20\text{V}$ to 26V	-	3	75
Load Regulation (Note1)	Regload	$T_J = +25^\circ\text{C}$ $I_O = 5\text{mA}$ to 1.5A	-	12	100	mV
		$I_O = 5\text{mA}$ to 1.0A	-	12	100	
		$I_O = 250\text{mA}$ to 750mA	-	5	50	
Quiescent Current	I_Q	$T_J = +25^\circ\text{C}$	-	5.2	6.0	mA
Quiescent Current Change	ΔI_Q	$V_I = 17.5\text{V}$ to 30V , $T_J = +25^\circ\text{C}$	-	-	0.8	mA
		$V_I = 17.5\text{V}$ to 30V , $I_O = 500\text{mA}$	-	-	0.8	
		$I_O = 5\text{mA}$ to 1.0A	-	-	0.5	
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-1.0	-	mV/°C
Output Noise Voltage	V_N	$f = 10\text{Hz}$ to 100KHz $T_A = +25^\circ\text{C}$	-	10	-	µV/V _O
Ripple Rejection	RR	$f = 120\text{Hz}$, $I_O = 500\text{mA}$ $V_I = 18.5\text{V}$ to 28.5V	-	58	-	dB
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}$, $T_J = +25^\circ\text{C}$	-	2.0	-	V
Output Resistance	r_O	$f = 1\text{KHz}$	-	19	-	mΩ
Short Circuit Current	I_{SC}	$V_I = 35\text{V}$, $T_A = +25^\circ\text{C}$	-	250	-	mA
Peak Current	I_{PK}	$T_J = +25^\circ\text{C}$	-	2.2	-	A

Note:

1. Load and line regulation are specified at constant junction temperature. Change in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7818A)

(Refer to the test circuits. $0^\circ\text{C} < T_J < 125^\circ\text{C}$, $I_O = 1\text{A}$, $V_I = 27\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Output Voltage	V_O	$T_J = +25^\circ\text{C}$	17.64	18	18.36	V
		$I_O = 5\text{mA to } 1\text{A}, P_O \leq 15\text{W}$ $V_I = 21\text{V to } 33\text{V}$	17.3	18	18.7	
Line Regulation (Note1)	Regline	$V_I = 21\text{V to } 33\text{V}$ $I_O = 500\text{mA}$	-	15	180	mV
		$V_I = 21\text{V to } 33\text{V}$	-	5	180	
		$T_J = +25^\circ\text{C}$	$V_I = 20.6\text{V to } 33\text{V}$	-	15	180
			$V_I = 24\text{V to } 30\text{V}$	-	5	90
Load Regulation (Note1)	Regload	$T_J = +25^\circ\text{C}$ $I_O = 5\text{mA to } 1.5\text{A}$	-	15	100	mV
		$I_O = 5\text{mA to } 1.0\text{A}$	-	15	100	
		$I_O = 250\text{mA to } 750\text{mA}$	-	7	50	
Quiescent Current	I_Q	$T_J = +25^\circ\text{C}$	-	5.2	6.0	mA
Quiescent Current Change	ΔI_Q	$V_I = 21\text{V to } 33\text{V}, T_J = +25^\circ\text{C}$	-	-	0.8	mA
		$V_I = 21\text{V to } 33\text{V}, I_O = 500\text{mA}$	-	-	0.8	
		$I_O = 5\text{mA to } 1.0\text{A}$	-	-	0.5	
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-1.0	-	$\text{mV}/^\circ\text{C}$
Output Noise Voltage	V_N	$f = 10\text{Hz to } 100\text{KHz}$ $T_A = +25^\circ\text{C}$	-	10	-	$\mu\text{V}/\text{Vo}$
Ripple Rejection	RR	$f = 120\text{Hz}, I_O = 500\text{mA}$ $V_I = 22\text{V to } 32\text{V}$	-	57	-	dB
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}, T_J = +25^\circ\text{C}$	-	2.0	-	V
Output Resistance	r_O	$f = 1\text{KHz}$	-	19	-	$\text{m}\Omega$
Short Circuit Current	I_{SC}	$V_I = 35\text{V}, T_A = +25^\circ\text{C}$	-	250	-	mA
Peak Current	I_{PK}	$T_J = +25^\circ\text{C}$	-	2.2	-	A

Note:

1. Load and line regulation are specified at constant junction temperature. Change in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Electrical Characteristics (MC7824A)

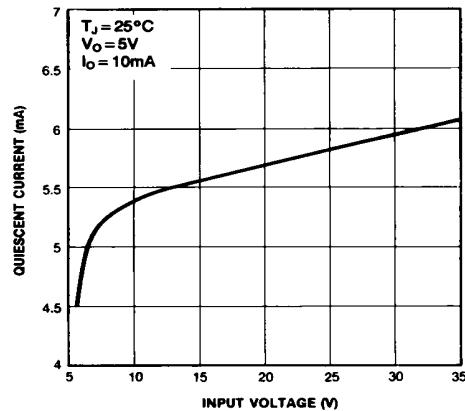
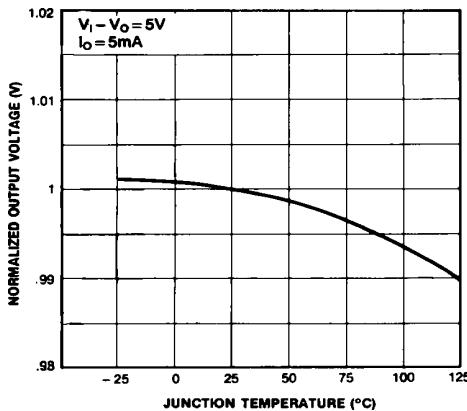
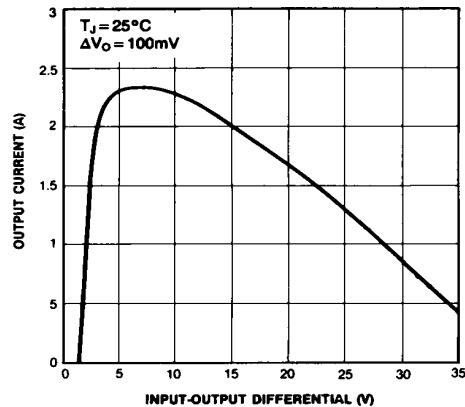
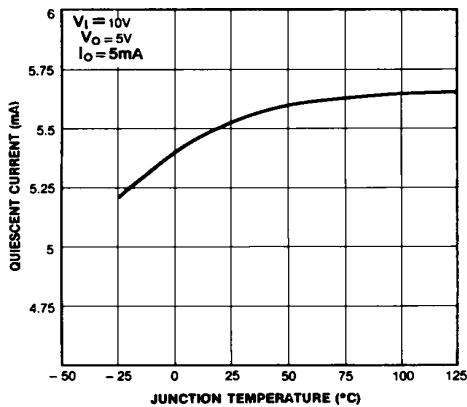
(Refer to the test circuits. $0^{\circ}\text{C} < T_J < 125^{\circ}\text{C}$, $I_O = 1\text{A}$, $V_I = 33\text{V}$, $C_I = 0.33\mu\text{F}$, $C_O = 0.1\mu\text{F}$, unless otherwise specified)

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Output Voltage	V_O	$T_J = +25^{\circ}\text{C}$	23.5	24	24.5	V
		$I_O = 5\text{mA to } 1\text{A}, P_O \leq 15\text{W}$ $V_I = 27.3\text{V to } 38\text{V}$	23	24	25	
Line Regulation (Note1)	Regline	$V_I = 27\text{V to } 38\text{V}$ $I_O = 500\text{mA}$	-	18	240	mV
		$V_I = 21\text{V to } 33\text{V}$	-	6	240	
		$T_J = +25^{\circ}\text{C}$	$V_I = 26.7\text{V to } 38\text{V}$	-	18	240
			$V_I = 30\text{V to } 36\text{V}$	-	6	120
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}\text{C}$ $I_O = 5\text{mA to } 1.5\text{A}$	-	15	100	mV
		$I_O = 5\text{mA to } 1.0\text{A}$	-	15	100	
		$I_O = 250\text{mA to } 750\text{mA}$	-	7	50	
Quiescent Current	I_Q	$T_J = +25^{\circ}\text{C}$	-	5.2	6.0	mA
Quiescent Current Change	ΔI_Q	$V_I = 27.3\text{V to } 38\text{V}, T_J = +25^{\circ}\text{C}$	-	-	0.8	mA
		$V_I = 27.3\text{V to } 38\text{V}, I_O = 500\text{mA}$	-	-	0.8	
		$I_O = 5\text{mA to } 1.0\text{A}$	-	-	0.5	
Output Voltage Drift	$\Delta V/\Delta T$	$I_O = 5\text{mA}$	-	-1.5	-	$\text{mV}/^{\circ}\text{C}$
Output Noise Voltage	V_N	$f = 10\text{Hz to } 100\text{KHz}$ $T_A = 25^{\circ}\text{C}$	-	10	-	$\mu\text{V}/V_O$
Ripple Rejection	RR	$f = 120\text{Hz}, I_O = 500\text{mA}$ $V_I = 28\text{V to } 38\text{V}$	-	54	-	dB
Dropout Voltage	V_{Drop}	$I_O = 1\text{A}, T_J = +25^{\circ}\text{C}$	-	2.0	-	V
Output Resistance	r_O	$f = 1\text{KHz}$	-	20	-	$\text{m}\Omega$
Short Circuit Current	I_{SC}	$V_I = 35\text{V}, T_A = +25^{\circ}\text{C}$	-	250	-	mA
Peak Current	I_{PK}	$T_J = +25^{\circ}\text{C}$	-	2.2	-	A

Note:

1. Load and line regulation are specified at constant junction temperature. Change in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Typical Performance Characteristics



Typical Applications

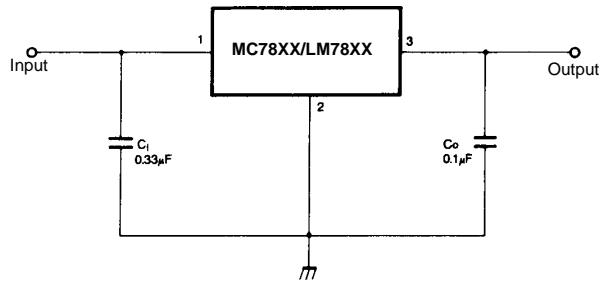


Figure 5. DC Parameters

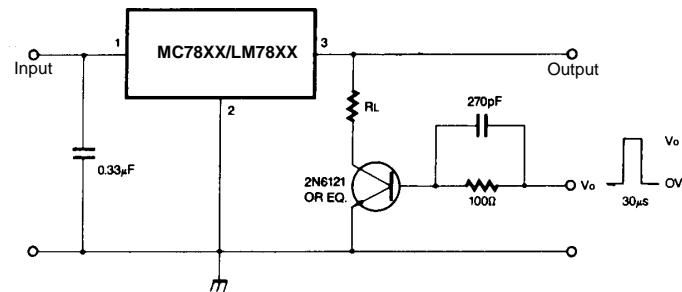


Figure 6. Load Regulation

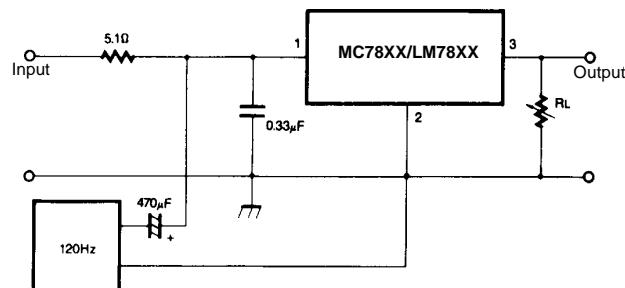


Figure 7. Ripple Rejection

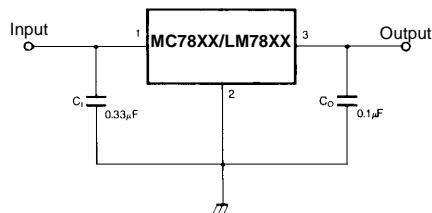
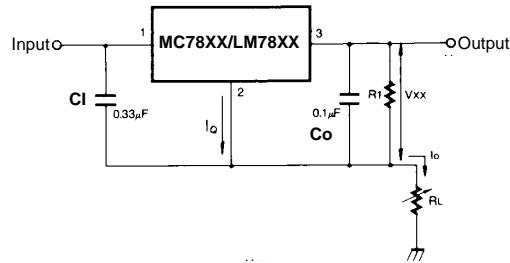


Figure 8. Fixed Output Regulator

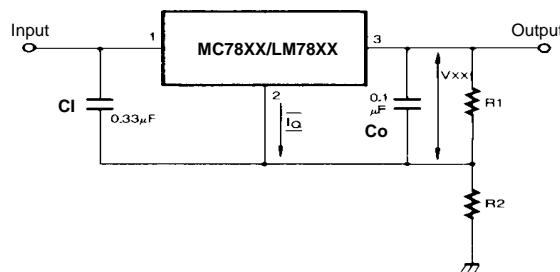


$$I_O = \frac{V_{XX}}{R_1} + I_Q$$

Figure 9. Constant Current Regulator

Notes:

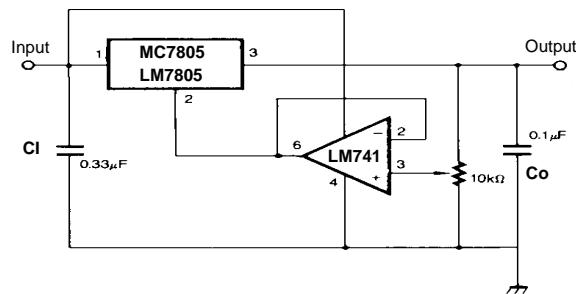
- (1) To specify an output voltage, substitute voltage value for "XX." A common ground is required between the input and the Output voltage. The input voltage must remain typically 2.0V above the output voltage even during the low point on the input ripple voltage.
- (2) Cl is required if regulator is located an appreciable distance from power Supply filter.
- (3) Co improves stability and transient response.



$$I_{R1} \geq 5I_Q$$

$$V_O = V_{XX}(1+R_2/R_1)+I_Q R_2$$

Figure 10. Circuit for Increasing Output Voltage



$$I_{R1} \geq 5I_Q$$

$$V_O = V_{XX}(1+R_2/R_1)+I_Q R_2$$

Figure 11. Adjustable Output Regulator (7 to 30V)

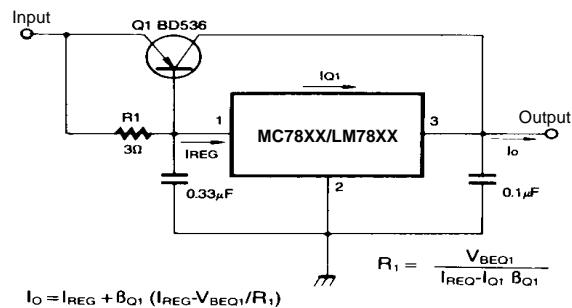


Figure 12. High Current Voltage Regulator

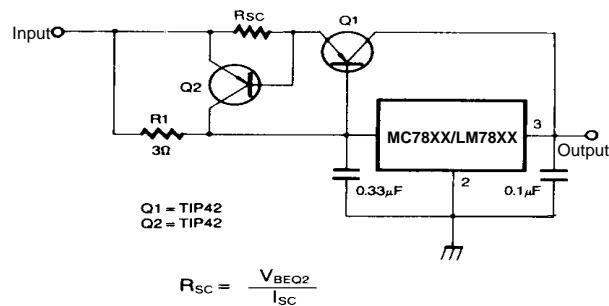


Figure 13. High Output Current with Short Circuit Protection

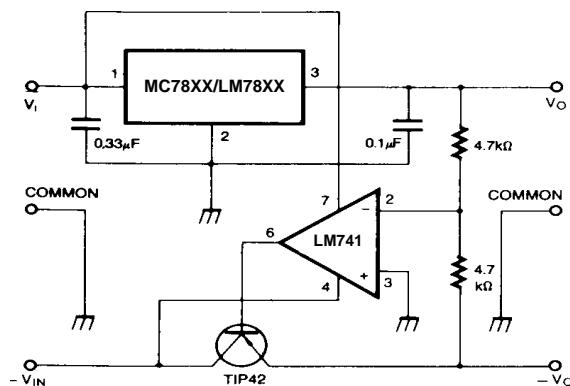


Figure 14. Tracking Voltage Regulator

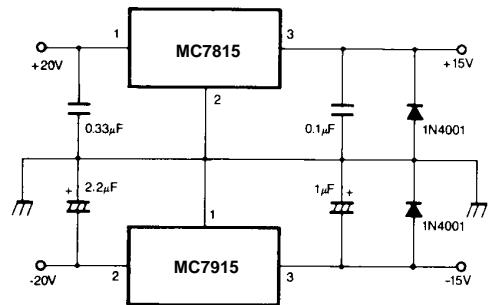


Figure 15. Split Power Supply (±15V-1A)

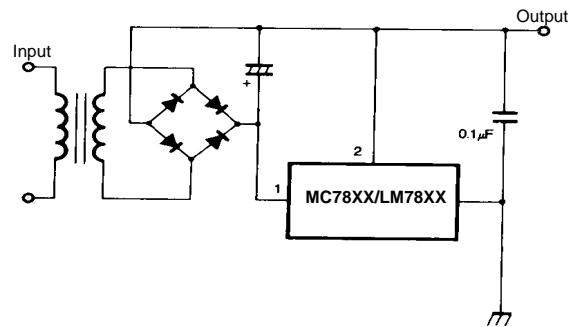


Figure 16. Negative Output Voltage Circuit

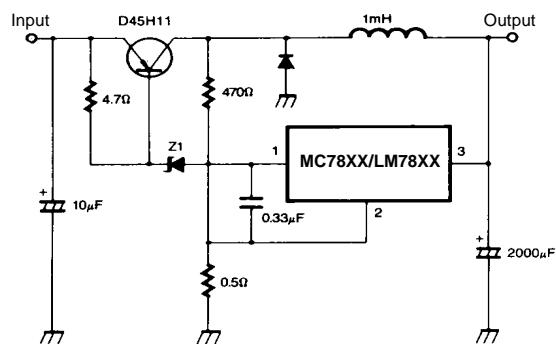
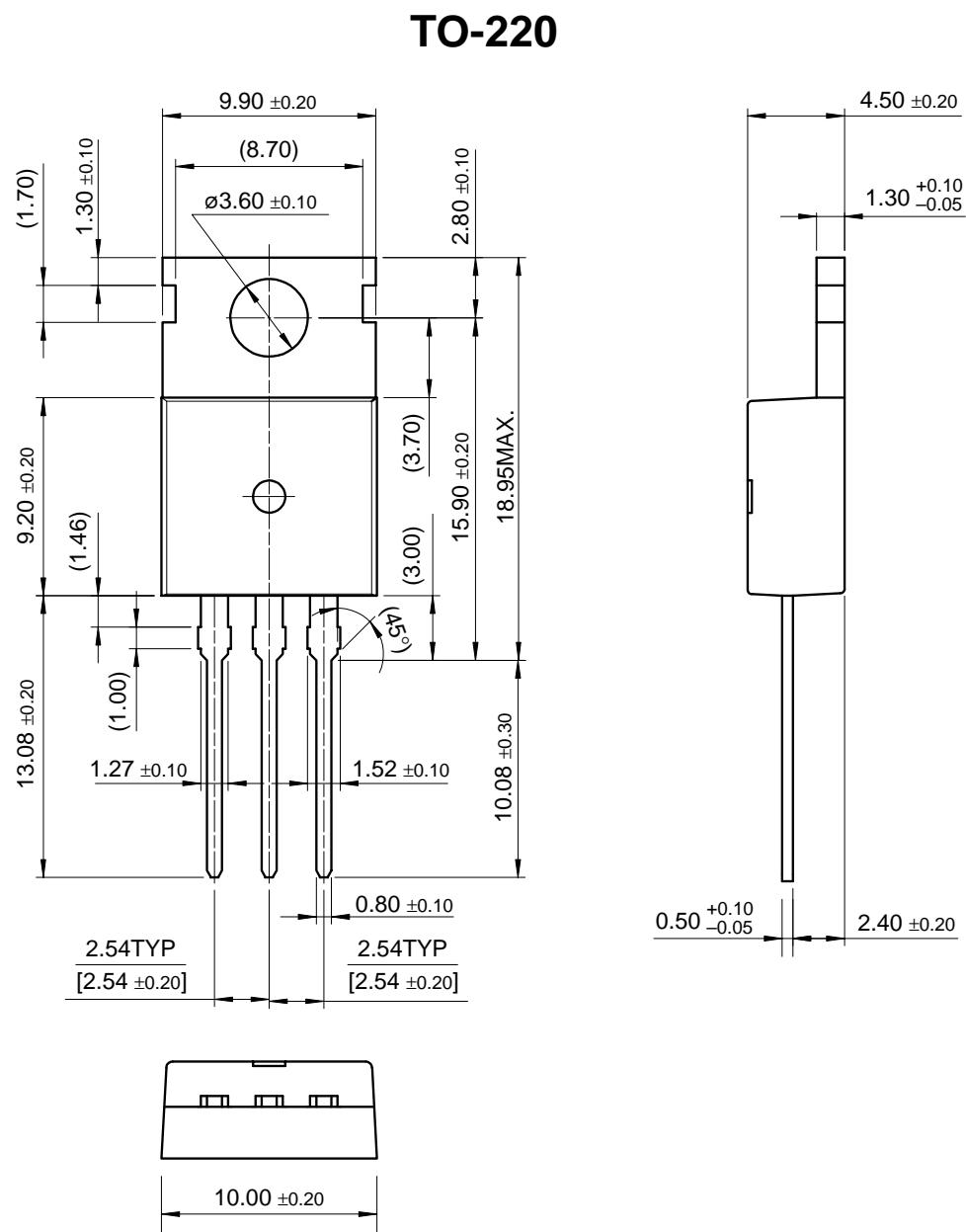


Figure 17. Switching Regulator

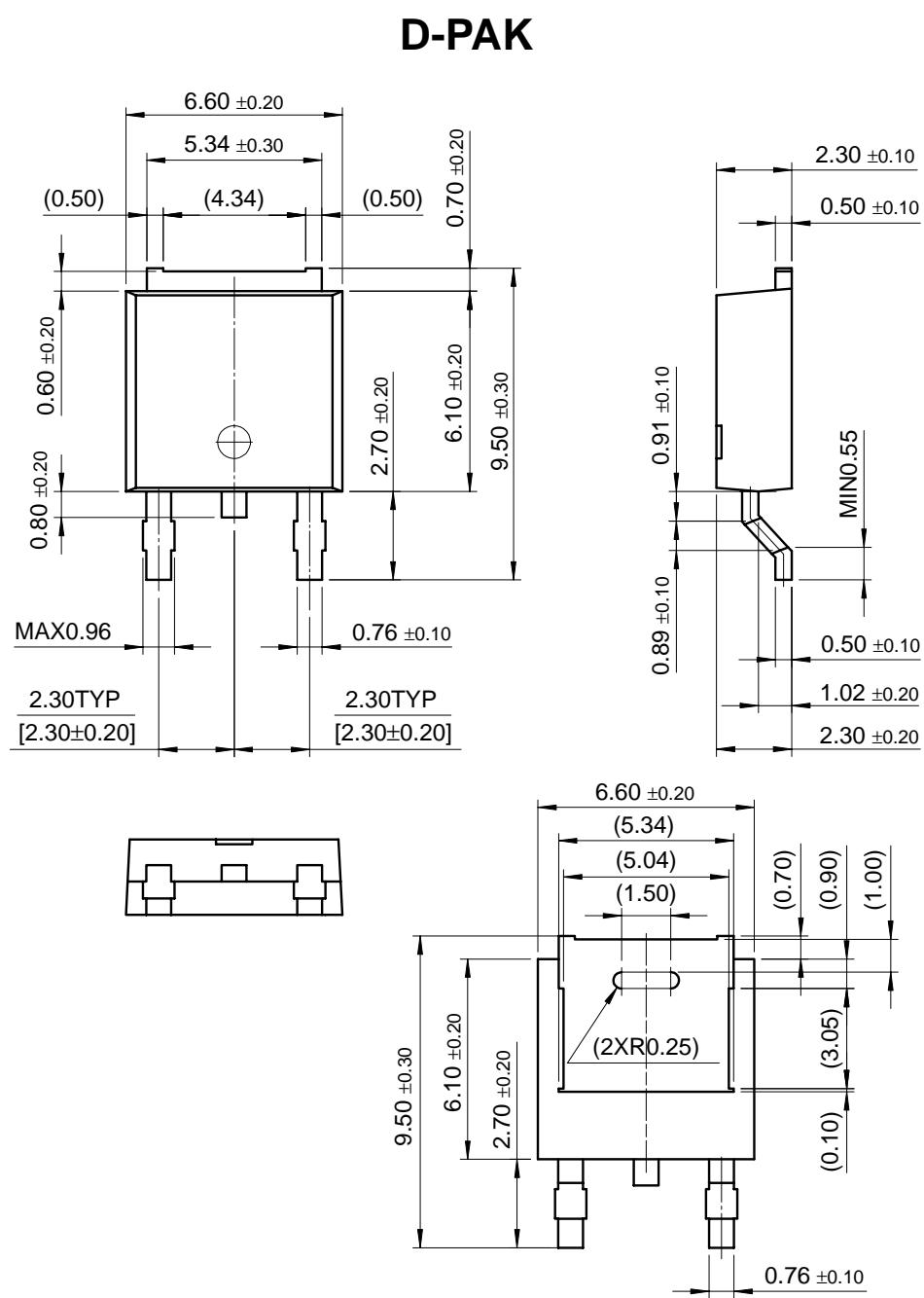
Mechanical Dimensions

Package



Mechanical Dimensions (Continued)

Package



Ordering Information

Product Number	Output Voltage Tolerance	Package	Operating Temperature
LM7805CT	$\pm 4\%$	TO-220	0 ~ + 125°C

Product Number	Output Voltage Tolerance	Package	Operating Temperature	
MC7805CT	$\pm 4\%$	TO-220	0 ~ + 125°C	
MC7806CT				
MC7808CT				
MC7809CT				
MC7810CT				
MC7812CT				
MC7815CT				
MC7818CT				
MC7824CT				
MC7805CDT		D-PAK		
MC7806CDT				
MC7808CDT				
MC7809CDT				
MC7810CDT				
MC7812CDT				
MC7805ACT	$\pm 2\%$	TO-220	0 ~ + 125°C	
MC7806ACT				
MC7808ACT				
MC7809ACT				
MC7810ACT				
MC7812ACT				
MC7815ACT				
MC7818ACT				
MC7824ACT				

DISCLAIMER

FAIRCHILD SEMICONDUCTOR RESERVES THE RIGHT TO MAKE CHANGES WITHOUT FURTHER NOTICE TO ANY PRODUCTS HEREIN TO IMPROVE RELIABILITY, FUNCTION OR DESIGN. FAIRCHILD DOES NOT ASSUME ANY LIABILITY ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT OR CIRCUIT DESCRIBED HEREIN; NEITHER DOES IT CONVEY ANY LICENSE UNDER ITS PATENT RIGHTS, NOR THE RIGHTS OF OTHERS.

LIFE SUPPORT POLICY

FAIRCHILD'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF FAIRCHILD SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and (c) whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury of the user.
2. A critical component in any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

REFERENCES

- [1] Amrit Kumar Das Saptarsi Gowswami Sruti Das Abriti Pual Sourav Ghosh and Soumya Sen. “A review on Agricultural Advancement Based on Computer Vision and Machine Learning”. In: *springer*. 2020. DOI: 10.1007/978-981-13-7403-6_50.
- [2] D. Wang Y.Zhao and D. Qian. “Machine Vision based image analysis for the estimation of Pear external quality”. In: *Second International Conference on Intelligent Computation Technology and Automation* (2009).
- [3] J. K .Archibald D. J. Lee and Guangming Xiong. “Rapid color grading for fruit quality evaluation using direct color mapping,” in: *IEEE Trans. Autom. Sci. Eng., vol. 8(2)* (2011).
- [4] P. H. Heinemann H. J. Sommer Y. Tao Varghese C. T. Morrow and R. W. Crassweller. “Automated inspection of golden delicious apples using color computer vision”. In: *American Society of Agricultural Engineers* 16 (1991(7002)).
- [5] C. T. Morrow Varghese. “Computer vision based date fruit grading system: Design and implementation”. In: *American Society of Agricultural Engineers* 16 (1991(7002)).
- [6] Leo Pauly and Deepa Sankar. “A New Method for Sorting And Grading Of Mangos Based On Computer Vision System”. In: *2015 3rd International Conference on Signal Processing, Communication and Networking (ICSCN)*. 2015. DOI: 10.1109/ICSCN.2015.7219819.
- [7] Yanhua Cui, Yizhong Wang, Shaohui Chen, Ping Zhang, and Huafang Huang. “Study on HSI Color Model-Based Fruit Quality Evaluation”. In: *2013 3rd International Congress on Image and Signal Processing (CISP2010)*. 2013. DOI: 10.1109/CISP.2010.5646706.
- [8] Di Wu and Da-Wen Sun. “Color measurements by Computer vision for food quality control: a review”. In: *Trends in Food Science Technology* 29 (2013), pp. 5–20.
- [9] L. Pauly and D. Sankar. “A New Method for Sorting And Grading Of Mangos Based On Computer Vision System”. In: *IEEE*. 2015. DOI: 10.1109/icsenst.2012.6461669.

- [10] D. Wu and D. W. Sun. “Color measurements by computer vision for food quality control: A review”. In: *Trends in Food Science & Technology* 29 (), pp. 5–20.
- [11] C. S. Nandi, B. Tudu, and C. Koley. “An automated machine vision based system for fruit sorting and grading”. In: *IEEE*. 2012. DOI: 10.1109/icsenst.2012.6461669.
- [12] Yousef Al Ohali. “Computer vision based date fruit Grading system: Design and implementation”. In: *Journal of King Saud University – Computer and Information Sciences* 23 (2011), pp. 29–36.
- [13] Di Wu and Da-Wen Sun. “Color measurements by Computer vision for food quality control: a review”. In: *Trends in Food Science & Technology* 29 (2013), pp. 5–20.
- [14] W. Huang, B. Zhang, L. Gong, and J. Li. “Computer vision detection of defective apples using automatic lightness correction and weighted RVM classifier”. In: *Journal of Food Engineering* 146 (2015), pp. 143–151.
- [15] Y. Cui, Y. Wang, S. Chen, P. Zhang, and H. Huang. “Study on HSI color model-based fruit quality evaluation”. In: *3rd International Congress on Image and Signal Processing (CISP2010)*. 2013.
- [16] E. Delage, H. Lee, and A. Y. Ng. “A dynamic Bayesian network model for autonomous 3D reconstruction from a single indoor image”. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. Vol. 2. 2006.
- [17] P. Coulibaly and N. D. Evora. “Comparison of neural network methods for infilling missing daily weather records”. In: *Journal of Hydrology* 341.1-2 (2007), pp. 27–41.
- [18] Raspberry Pi Foundation. *Raspberry Pi 4 Model B*. <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/>.
- [19] E. Upton and G. Halcree. *Raspberry Pi User Guide*. 4th. Wiley, 2014.
- [20] Raspberry Pi Foundation. *Raspberry Pi GPIO Pinout*. <https://pinout.xyz/>.
- [21] NVIDIA. *NVIDIA Jetson Nano Developer Kit*. <https://developer.nvidia.com/embedded/jetson-nano-developer-kit>.
- [22] Xilinx. *PYNQ-Z2: Python Productivity for Zynq*. <https://www.xilinx.com/products/boards-and-kits/1-dk-pony.html>.
- [23] Arduino. *Arduino Nano*. <https://www.arduino.cc/en/Guide/ArduinoNano>.

- [24] Adafruit. *Stepper Motor NEMA23 (KL23H2100-50-4B)*. <https://www.adafruit.com/product/324>.
- [25] Lisa Olsen. *Stepper Motor Basics*. <https://www.motioncontroltips.com/stepper-motor-basics/>.