CmpE242
Spring2023

1/

Jan 25th(Wed). 2023
Organizational meeting.
1° Class Syllabus, "Greensheet".

Organizational meeting. Note: 1° In-person Class, M.W. 3:00-4:15pm.; 2° Office Hours, Please make a good use of the O.H. But it will expire by the end of the Semester, By the end of the class. 3°.

San José State University
Computer Engineering Department
CMPE 242 Embedded Hardware Systems, Section 1, S2023 Lab Room/Facility: Rm 268. Access form.

Course and Contact Information

Instructor:	Hua Harry Li, Ph.D.
Office Location:	Engineering Building, Rm 267A
Telephone:	(650) 400-1116 Text Message Only
Email:	hua.li@sjsu.edu
Office Hours:	Mondays and Wednesdays 4:30 – 5:30 pm Zoom link for the Office Hours Join Zoom Meeting https://us04web.zoom.us/j/9841607683?pwd=UlA3aEk1TnV4bjNLQk5CQkw0dDk4UT09 Meeting ID: 984 160 7683 Passcode: 121092
Class Days/Time:	Monday and Wednesday 3:00 – 4:15 pm
Classroom:	Engineering Build Room 325
Prerequisites:	CMPE 180A and 180D, classified standing, or instructor consent

Note: 4°. Nature of the Class

Target platform selection:
- Hardware: NVDA Jetson NANO 2gb/4gb. Broadcom R.pie 3b+, Pie 4.
- Software: 1° Kernel Source Dist. JetPack; 2° Device Development SPI, I2C, PWM, 3° Python, C/C++

Course Format

Technology Intensive, Hybrid, and Online Courses (Required if applicable)

This course requires use of computer/laptop, special microprocessor/ARM hardware for system prototyping, Python and/or C/C++ compiler for software programming. Students must have to participate in classroom activities and after class homework and projects assignment.

Faculty Web Page and MYSJSU Messaging (Optional)

Copies of the course reference materials such as datasheets, project references etc. can be found on line at https://github.com/hualili/CMPE242-Embedded-Systems- and/or SJSU CANVAS. Office hours Zoom link (during the Pandemic): Join Zoom Meeting https://us04web.zoom.us/j/9841607683?pwd=UlA3aEk1TnV4bjNLQk5CQkw0dDk4UT09 Meeting ID: 984 160 7683 Passcode: 121092

5° Github

6°. Homework/projects submission on CANVAS.

Course Description (Required)

Advanced topics dealing with microprocessor and microcontroller hardware and firmware including processor architecture, advanced memory and I/O systems design, multilevel bus architecture, interrupt systems. Design project. Prerequisites: CMPE 180A and 180D, classified standing, or instructor consent.

Course Learning Outcomes (CLO) (Required)

hualili / CMPE242-Embedded-Systems- Publi

Course Learning Objectives (CLO):

CMPE242
Spring2023

2/

Course Description/Nature: Hands-on, Sound Theoretical Background, Coverage of Theory. Note: Sensors. LSM303

3D Printer, CNC machines.

Motors { Stepper motors. NEMA17
3phase BLDC motor.
Automobile window Wipper motor

Robotics.

Required Texts/Readings (Required) Note: Datasheets.

Textbook

1. S3C6410 RISC Processor datasheets, Samsung Electronics https://github.com/hualili/CMPE244/blob/main/2021F-105-%230-cpu-arm11-2018S-29-CPU_S3C6410X.pdf and Development Board schematics https://github.com/hualili/CMPE244/blob/main/2021F-105b-%232018S-29-SCH-Tiny6410SDK-1111-PCB.pdf
2. Nvidia Jetson NANO datasheets.
   (a) Jetson Nano development kit document https://github.com/hualili/CMPE244/blob/main/2021F-108-%231NVIDIA_Jetson_Nano_Developer_Kit_User_Guide.pdf
   (b) Jetson NANO System-on-Module https://github.com/hualili/CMPE244/blob/main/2021F-108b-%23JetsonNano_DataSheet.pdf
   (c) Optional (not used) SoC Park CPU reference https://github.com/hualili/CMPE244/blob/main/2021F-106-tx2-%23Parker_TRM_DP07821001p.pdf
3. Broadcom Raspberry Pie CPU datasheets, BCM2835 CPU https://github.com/hualili/CMPE244/blob/main/2021F-104-%230-cpu-pie-BCM2835-ARM-Peripherals.pdf and https://github.com/hualili/CMPE244/blob/main/2021F-104d-simplifiedCPU-datasheet-%23rpi_DATA_CM_1p0.pdf

Other Readings

1. Professor Li's PPT, handout materials, lecture notes on line https://github.com/hualili/CMPE242-Embedded-Systems-

Ref: on github, Lecture Notes.

2022S-101-notes-cmpe242-3-14.pdf

CmpE242
Spring2023

Grading Information (Required)

Midterm Examination	30%
Homework and Projects	30%
Final Examination	40%

The examination grades are given based on the written answer in
grades are given based on the work submitted, prototype system
programming source code. The detailed rubrics for each homew
assignment is given, check online both CANVAS and https://git
project will be given to students for each submission with multip
learning. Rubrics examples for project 1 submission, for exampl
software implementation counts 40%, report counts 20%, so the

Determination of Grades

Exams: In-Person, In-Class.
Submission To CANVAS.
Need Laptop & Prototype System in the exam.

Alternative1: Broadcom. Raspberry Pie3, 3B+, 4.
Quad ARM.

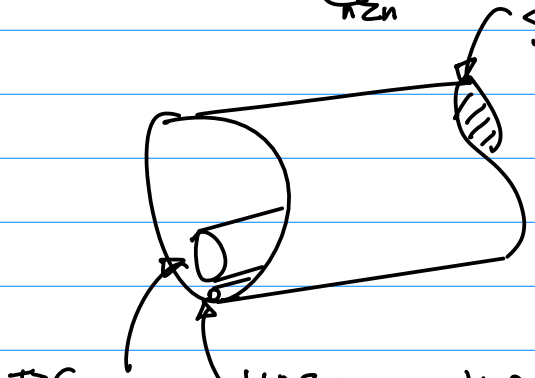
Jan29 (Monday).
1. Homework. 0 pt. Honesty pledge.
Due this Wednesday, ON CANVAS.

Discussion: I/O I/F for An Embedded System:
1° UART (Serial Communication) Tx/Rx/GND
MiniCom/Putty ~1200kbps Slow!
2° SPI. ~100mbps
3° I2C {SDA SCK
4° PWM
5° CAN
6° ADC

"3+1" {MOSI MISO SCK nEn

Ref from the github
2022S-101-notes-cmpe242-3-14.pdf

Example: Selection of Target platform.
Build Selection matrix Below.
1° Architectural Aspects.
X86; ARM, MIPS
CISC : RISC MIPS 1982
for Server
2. User Basis, Marketshare.
3. OS kernel Aspect: Linux/Unix.
4. Forward Looking → GPU →
GPGPU (general purpose) →
AI/ML.
for Example: Jetson NANO
Quad CPU: ARM.
GPU (128 GPUs)

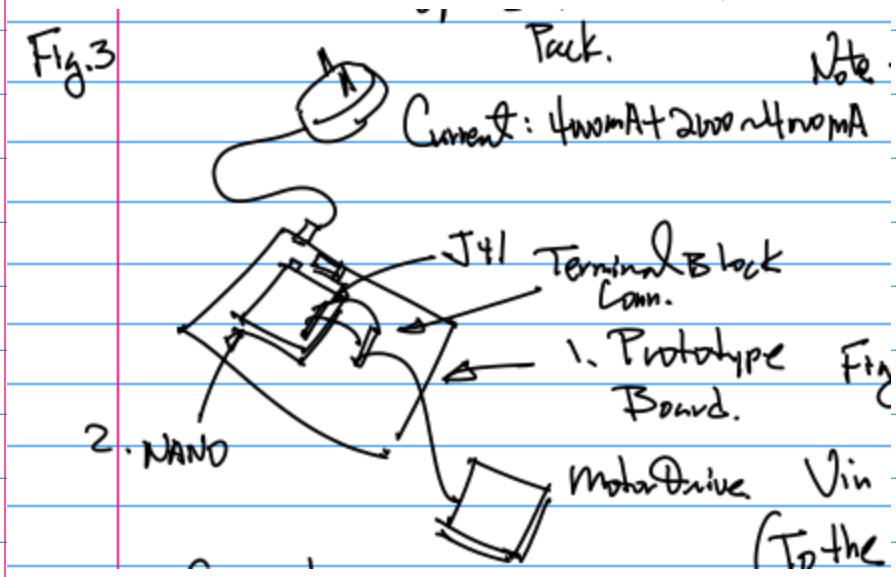


NANO, ~$140
4gb.
Check 2gb Towards the end of Life.

Homework Preparation.

1° Build A prototype Board. Ref. pp.3. Fig.3

Fig.3



Note: 1° prototype Board. Dimension:

Feb 1. (Wednesday).

Note: 1. Target Board Selection By today. Bring your Target Together with the Prototype Board to the Class a week from today.

Example: To prepare the first Homework.

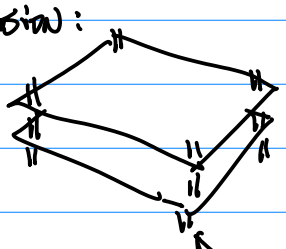
Note: 1° First Homework: "Hello, the World" prototype System.

Python (Pycharm IDE) program to flip LED. Turn ON/OFF LED.

2° Prototype Board.

Order on Line (Amazon) or Local store, Anchor Electronics. (Santa Clara.)

Dimension:



a. Stand-offs Required

b. Connectors Encouraged/Required;

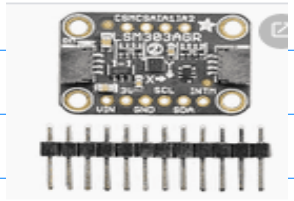
10pcs Upgraded Tiny Whoop JST-PH 2.0 Male and Female Connector Cable for Battery JJRC H36 H67 Blade Inductrix E010 E013

C. Bread Board for Quick Prototyping.

d. Right gauge of the prototyping wires, #28 or higher (etc. 1 or 2 steps. Ref: 10mA ~ upto 40~50mA)

3. LED Assted, Red, Yellow. etc.
Resistors.
Capacitors. 4.7nF (1~2PCS)

4° LSM303 (I2C, SPI)



Adafruit Industries
LLC 4413

This board/chip uses I2C 7-bit addresses 0x19 & 0x1E

5° I2C Approximity Sensor.

Note: I2C Mux for Multi_I2C Devices.

6° Design for the 1st Homework.

- Hardware Design
- Software Design
  - I/P Testing.
    - I.P. High
    - I.P. Low
  - O/P Testing
    - Op. High → LED ON
    - Op. Low → OFF

Note: Toggle SW to generate "High" or "Low" Input to the GPIO of the NANO

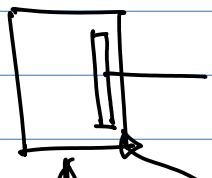
Pin Assignment Table

Pin	Description	Note	
J41-1	3V3		
J41-2	5V		
J41-39	GND		
J41-12	GPIO	gpio79	Output
J41-40	GPIO	gpio78	Input

element14 Community

Jetson Nano Dev-Board Expansion Header

Alt Function	Linux(BCM)	Board Label		Board Label	Linux(BCM)
DAP4_DOUT	78(21)	D21	40 39	GND	
DAP4_DIN	77(20)	D20	38 37	D26	12(26)
UART2_CTS	51(16)	D16	36 35	D19	76(19)
		GND	34 33	D13	38(13)
LCD_BL_PWM	168(12)	D12	32 31	D6	200(6)
		GND	30 29	D5	149(5)
		D1/ID_SC	28 27	D0/ID_SD	
SPI1_CS1	20(7)	D7	26 25	GND	
SPI1_CS0	19(8)	D8	24 23	D11	18(11)
SPI2_MISO	13(25)	D25	22 21	D9	17(9)
		GND	20 19	D10	16(10)
SPI2_CS0	15(24)	D24	18 17	3.3V	
SPI2_CS1	232(23)	D23	16 15	D22	194(22)
		GND	14 13	D27	14(27)
DAP4_SCLK	79(18)	D18	12 11	D17	50(17)
		RXD/D15	10 9	GND	
		TXD/D14	8 7	D4	216(4)



Target GPU NANO

J41, 40pins connector/2 Rows. (Longest Connector)

Identify 3 pins
- GND
- Vcc
- GPIO

Good Ref. Source

https://jetsonhacks.com › nvidia-jetso

NVIDIA Jetson Nano J41 H

Feb 6 (Monday).
Todays Topics: Design of Prototype Board to Bring up the target platform (NAND).

Ref: 1° Github

CMPE242-Embedded-Systems- / 2022S / 2022S-103-SDcard-source-distribution-tool-chain-menuconfigu-2021-10-8.pdf

2° Github, Lecture Notes

/stems- / 2022S / 2022S-101-notes-cmpe242-3-14.pdf

Note: Bring your target platform to the Class for inspection on Wednesday.

a. Target platform.
b. Work-in-progress.

Note: Try to use ON-Line tool.

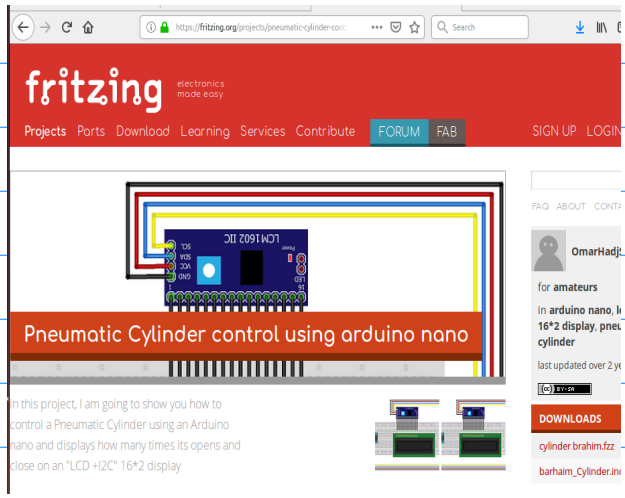
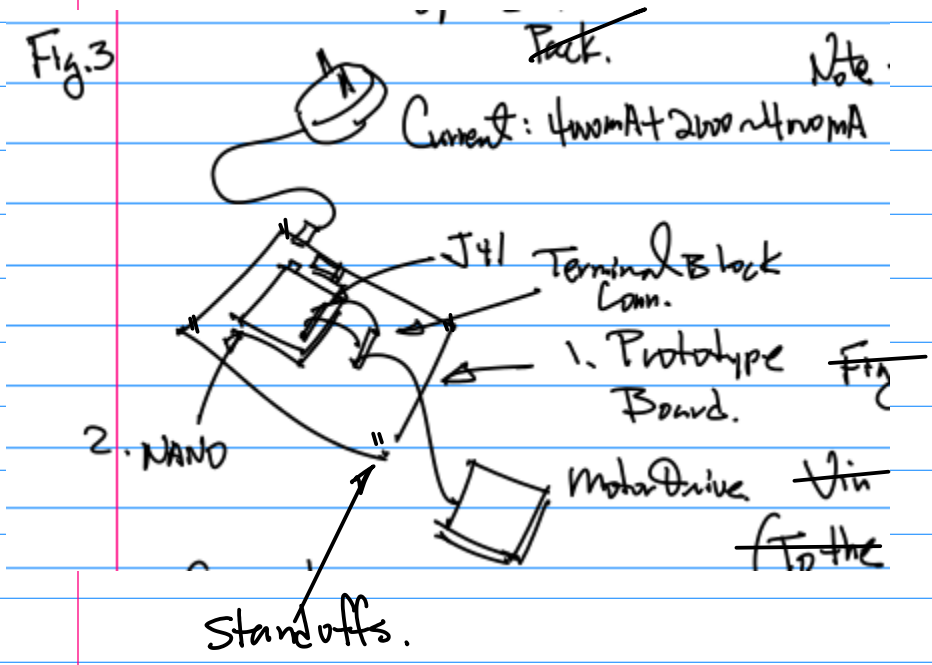
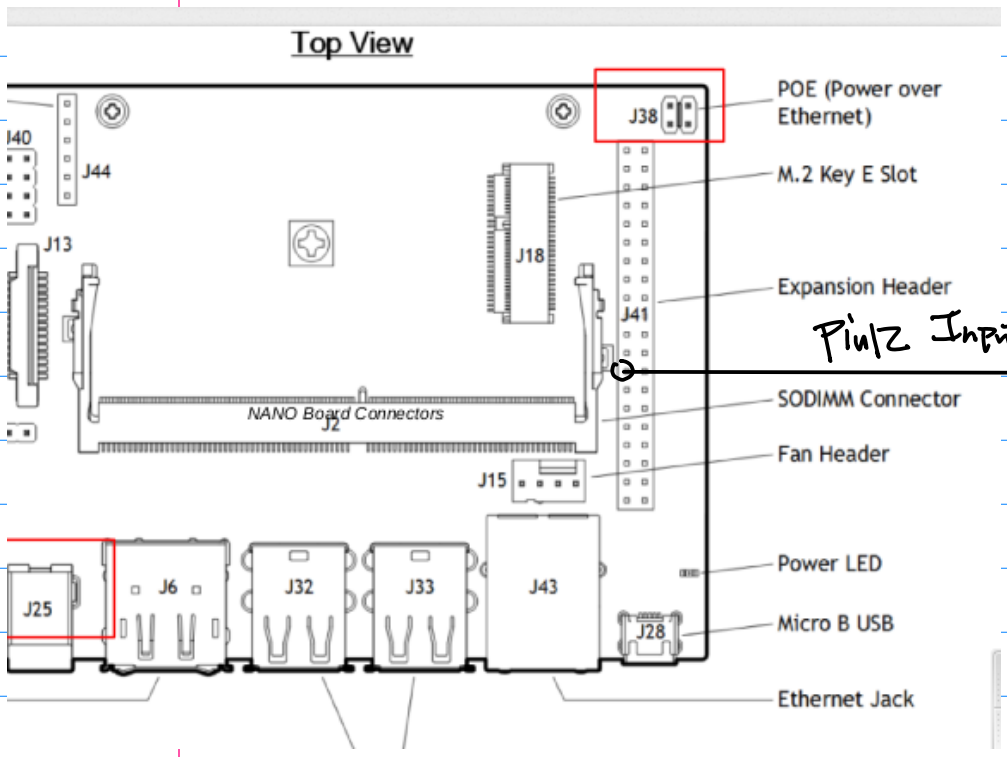
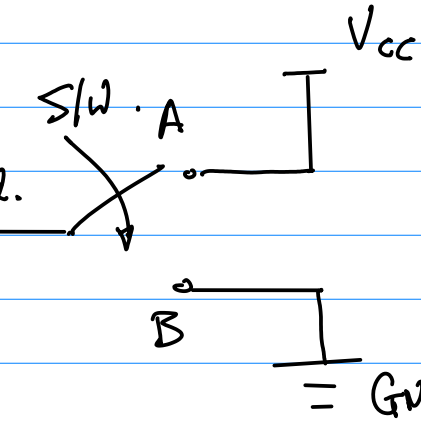


Fig.3



Example: Continuation ON GPIO I/F Design.
Design for the Input Testing.





↓ modify the Design to Add R_1

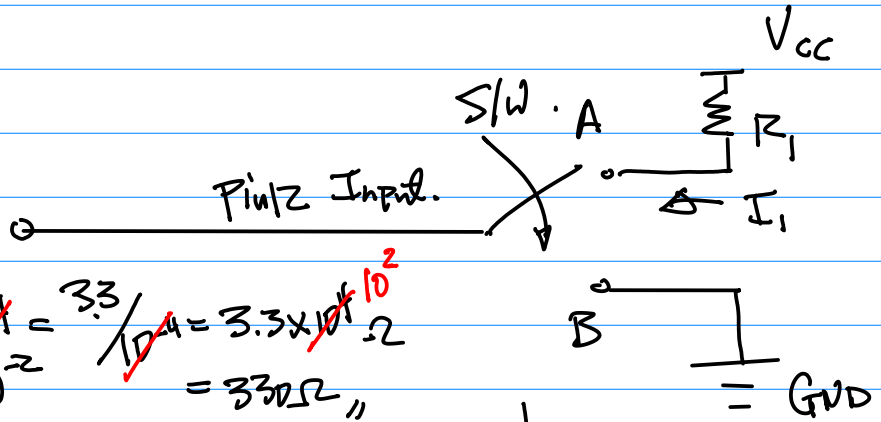
Let $I_1 = 10\,mA$, find R_1.

$V_{CC} - R_1 I_1 = V_{in}$

where $I_1 = 10\,mA$, $V_{in} = 0V$.

$V_{CC} - R_1 \times 10 \times 10^{-3} = 0$, $R_1 = V_{CC}/10^{-2} = 3.3/10^{-2} = 3.3 \times 10^{2}\,\Omega$

$= 330\Omega$ //



↓ modify the Design By Adding R_2

Next, update the Design to add R_2 to Regulate I_2.

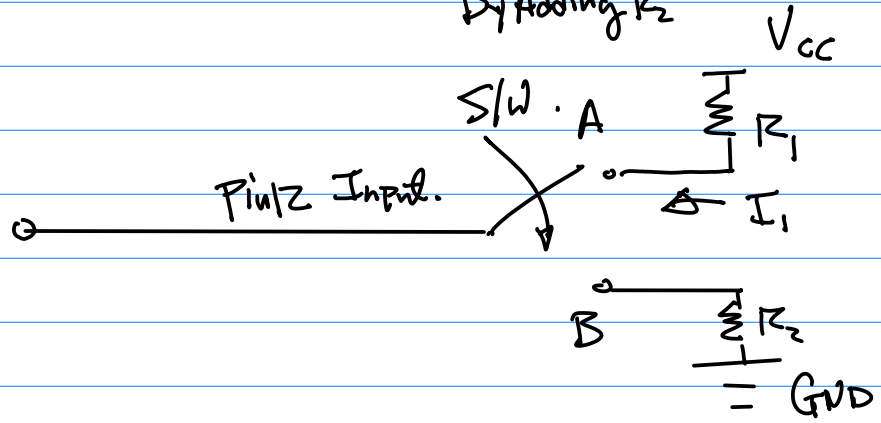
Make $I_2 = 10 \times 10^{-3}$ A

Assum $V_{12} = V_{CC} = 3.3V$.

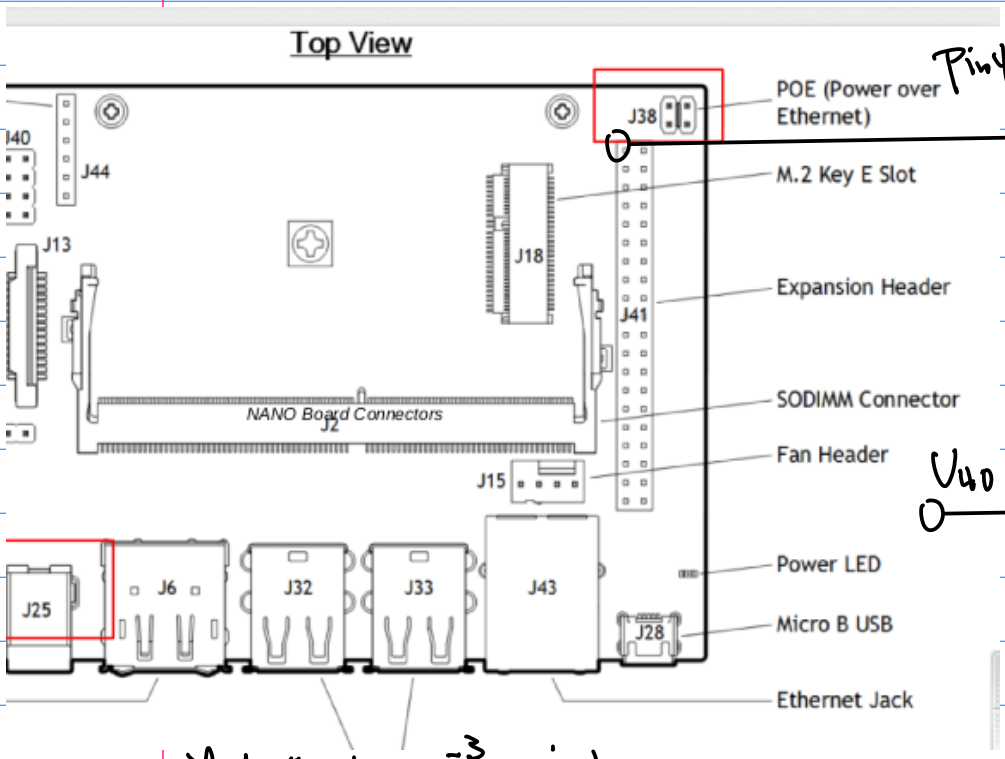
$V_{12} - I_2 R_2 = 0$, hence

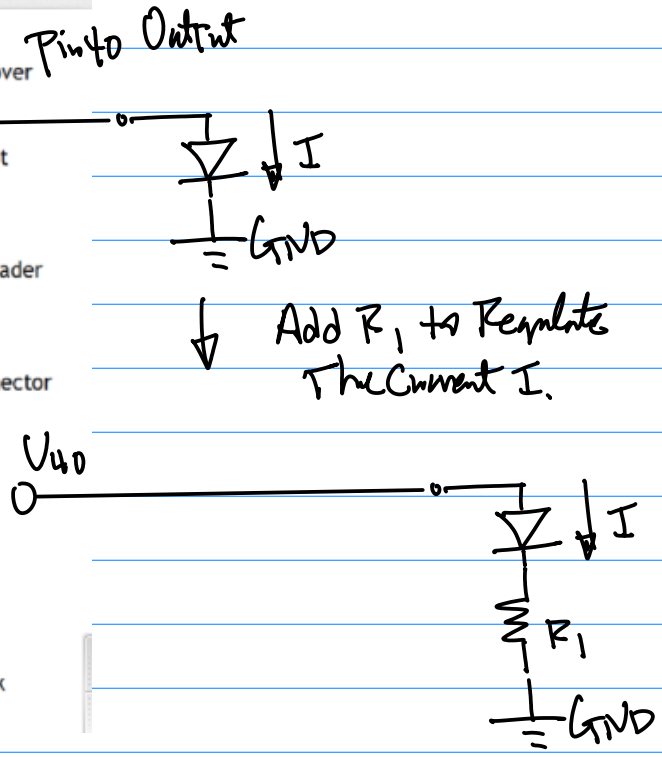
$V_{CC} = I_2 R_2 \Big|_{I_2 = 10 \times 10^{-3}}$

$\therefore R_2 = V_{CC}/I_2 = 3.3/10^{-2} = 330\Omega$.



Output Testing Circuit.





Let $I \approx 10 \times 10^{-3}$. find R_1.

$V_{40} - I R_1 = 0$, $R_1 = V_{40}/I \Big|_{V_{40}=V_{CC}=3.3V,\ I=10\times10^{-3}} = 3.3/10^{-2} = 330\Omega$.

2022S-103-SDcard-source-distribution-tool-chain-menuconfigu-2021-10-8.pdf

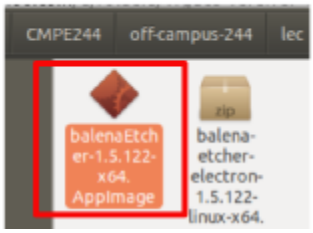
Software Design:

Step 1. Prepare microSD Card, 16GB or 32GB (for Additional App.). Down Load Pre-Compiled/Built Kernel Image from the Nvidia Site

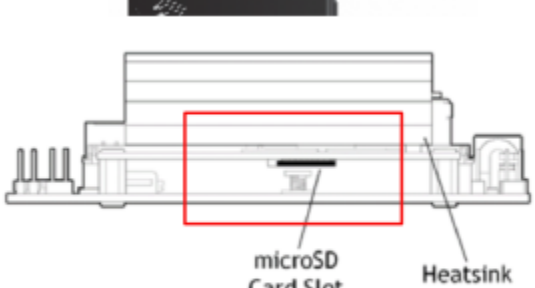
https://developer.nvidia.com/embedded/learn/get-started-jetson-nano-devkit#write

Step2. Down Load the software "Etcher" to your host machine in order to write the Kernel image to your micro SD Card.

(2.1) for Linux host, Download, install, and launch Etcher.
https://www.balena.io/etcher/



Step3. Take the MicroSD, insert it to your target platform.



the power up the NANO Board, Config the Board By following the prompt.

Note: It is recommended to use 4 Amps Power Adaptor.

Step4. Init & Config. for GPIO Driver.

Ref.

2022S-104-gpio-systemLevel-and-c-#2021F-114-gpio-nano-v3-hl-2021-10-20.pdf

P.2. Note: a. Website, Ref. Sources.

Pi and NANO are pin to pin compatible
Jetson Nano GPIO - JetsonHackshttps://www.jetsonhacks.com› ... › GPIO/I2C
Jun 7, 2019 — As you may have heard, the GPIO pin layout on the Jetson Nano is compatible with the 40 pin layout of a Raspberry Pi (RPi).

b. 40 pin Connector is Compatible With Raspberry Pie.

c. By Default, the Kernel Image (O.S.) has already configured GPIO Driver, so, use Command line to Turn ON/OFF LED As follows.

```
$echo 79 > /sys/class/gpio/export
$ echo out > /sys/class/gpio/gpio79/direction
$echo 1 >  /sys/class/gpio/gpio79/value
$echo 0 >  /sys/class/gpio/gpio79/value
$echo 79 /sys/class/gpio/unexport
$cat /sys /kernel /debug/gpio
```

Connect to the GPIO Driver.
Config the GPIO As An Output
Set output="1"
Set Output="0"
Release GPIO

Homework: GPIO Testing. Due A week from Today.
Bring the Board to Class for Demo.