

ภาคผนวก I

การทดลองที่ 9 การศึกษาและปรับแก้อินพุตและเอาต์พุตต่างๆ

การทดลองในภาคผนวกนี้จะช่วยอธิบายเนื้อหาในบทที่ 6 ซึ่งเกี่ยวข้องกับอุปกรณ์อินพุต/เอาต์พุตที่หลากหลายบนเครื่องคอมพิวเตอร์ตั้งโต๊ะ โดยมีวัตถุประสงค์เหล่านี้

- เพื่อให้เข้าใจการปรับแก้อุปกรณ์อินพุตและเอาต์พุตชนิดต่าง ๆ บนระบบปฏิบัติการ Raspberry Pi OS
- เพื่อให้เข้าใจความแตกต่างระหว่างอุปกรณ์อินพุตและเอาต์พุตชนิดต่าง ๆ บนบอร์ด Pi
- เพื่อให้สามารถอ่านข้อความแสดงรายละเอียดของอุปกรณ์อินพุตและเอาต์พุตชนิดต่างๆ

หลักการและพื้นฐานความเข้าใจจะช่วยแนะแนวทางให้ผู้อ่านสามารถศึกษาค้นคว้า อินพุต/เอาต์พุตอื่น ๆ ในชิปและบนบอร์ดได้เพิ่มเติม รวมไปถึงบนโทรศัพท์เคลื่อนที่ แท็บเล็ต คอมพิวเตอร์ และอุปกรณ์อินเทอร์เน็ตสรรพสิ่ง (Internet of Things)

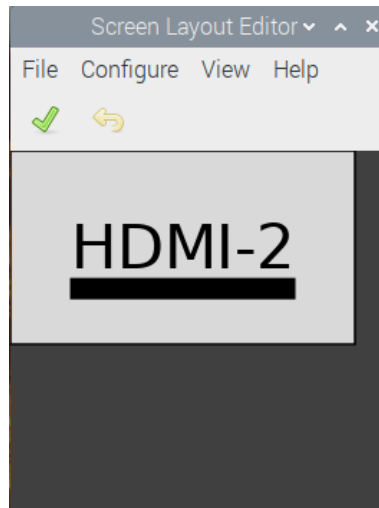
I.1 จอแสดงผลผ่านพอร์ต HDMI

I.1.1 การปรับแก้ความละเอียดของจอแสดงผล

เมื่อขนาดหน่วยความจำสำหรับการใช้งานและแสดงผลของจีพียูมีปริมาณเพียงพอ ซึ่งตั้งอยู่บริเวณชื่อว่า VC SDRAM ในพื้นที่หน่วยความจำกายภาพ (ARM Physical Memory) ของรูปที่ 6.16 ผู้ใช้สามารถปรับเพิ่มหรือลดความละเอียดของจอแสดงผลได้โดยกดปุ่มบนเมนูดังนี้

menu->Preferences->Screen Configuration

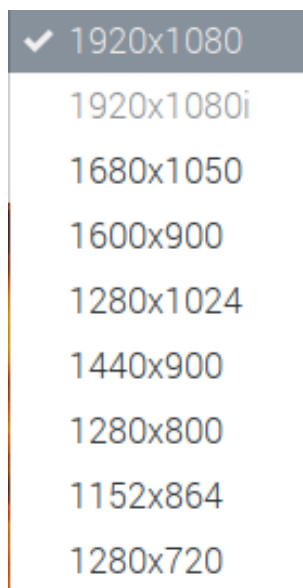
นี่เป็นการเรียกใช้โปรแกรม Screen Layout Editor



รูปที่ I.1: หน้าต่าง Screen Layout Editor สำหรับกำหนดค่าต่าง ๆ กับพอร์ตแสดงผล HDMI

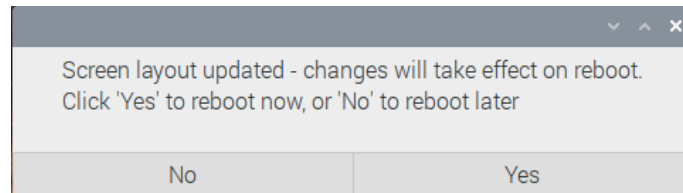
กดปุ่มบนเมนูบนโปรแกรมเพื่อปรับความละเอียดของพอร์ตที่จอแสดงผลเชื่อมต่ออยู่และรองรับ

Configure->Screens->HDMI-1 (HDMI-2) ->Resolution



รูปที่ I.2: หน้าต่าง Set Resolution สำหรับกำหนดความละเอียดหน้าจอแสดงผลที่ต้องการ

กด ปุ่ม เลือก ความ ละเอียด หน้า จอ ที่ เหมาะ สม กับ จอ ที่ เชื่อม ต่อ อยู่ คือ ไม่ เกิน ความ ละเอียด 1920x1080 หลังจากนั้นกดปุ่ม เครื่องหมายถูก ในหน้าต่างหลักของ Screen Layout Editor เพื่อยืนยัน หน้าต่างต่อไปนี้จะปรากฏขึ้น



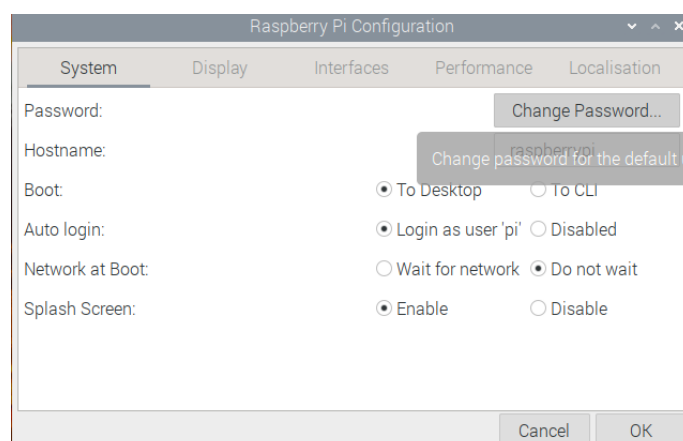
รูปที่ I.3: หน้าต่าง Reboot needed กดปุ่ม Yes เมื่อต้องการรีบูต ณ เวลานั้น

กด Yes เพื่อรีบูตระบบใหม่ ผู้อ่านสามารถค้นคว้าเรื่องสายมาตรฐาน HDMI ในหัวข้อที่ 6.1 เพิ่มเติม ก่อนค้นคว้าเพิ่มเติมในอินเทอร์เน็ต

I.1.2 การปรับแก้ขนาดหน่วยความจำของ GPU

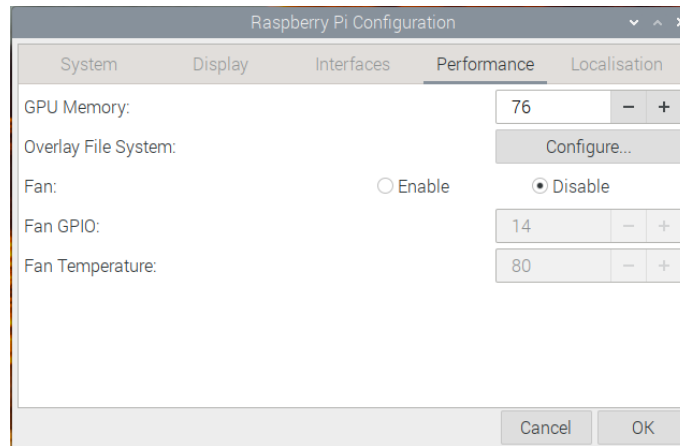
หน่วยความจำสำหรับจอแสดงผลหรือจีพียู (Graphic Processing Unit) ถูกแบ่งพื้นที่ออกจาก หน่วยความจำ SDRAM บนบอร์ด เพื่อใช้งานร่วมกันทำให้ประหยัดต้นทุน แต่มีข้อเสียในด้านประสิทธิภาพจะลดลง เมื่อผู้ใช้งานต้องการภาพที่มีอัตราเฟรมเรต (Frame Rate) สูง เช่น ภาพวิดีโอเคลื่อนไหว เกม 3 มิติ ความละเอียดของจอแสดงผลขึ้นตรงกับขนาดของหน่วยความจำของจีพียู ผู้อ่านสามารถปรับแก้ขนาดหน่วยความจำของจีพียูได้ดังนี้

menu->Preferences->Raspberry Pi Configuration



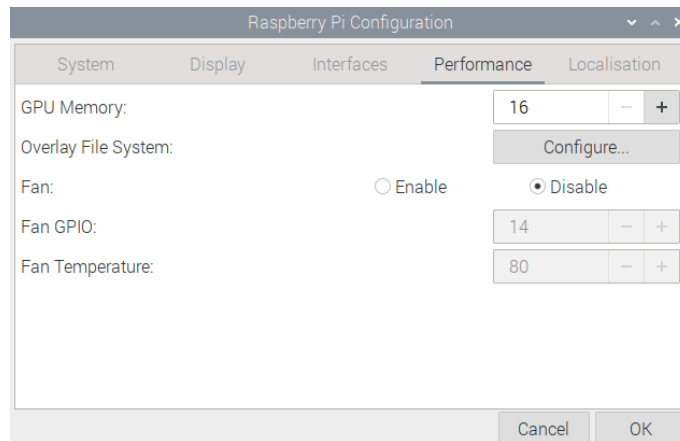
รูปที่ I.4: หน้าต่าง Raspberry Pi Configuration

กดแท็บ Performance เพื่อปรับค่าต่าง ๆ เกี่ยวกับจีพียู ผู้อ่านสามารถปรับลด (-) หรือเพิ่ม (+) ขนาดหน่วยความจำซึ่งเท่ากับ 76 เมกไบต์ในรูป



รูปที่ I.5: แท็บ Performance หน้าต่าง Raspberry Pi Configuration

โดยหน้าต่างที่ปรากฏขึ้นมีลักษณะดังนี้ ผู้ใช้สามารถกำหนดขนาดที่ต้องการโดยขั้นต่ำคือ 16 เมกไบต์ (MiB)



รูปที่ I.6: หน้าต่างกำหนดขนาดหน่วยความจำสำหรับจีพียูที่ 16 MiB

I.2 ระบบเสียงดิจิทัล

อุปกรณ์ระบบเสียงดิจิทัลที่ติดตั้งมาบนบอร์ด Pi จากโรงงาน ผู้ใช้สามารถเพิ่มเติมได้ผ่านพอร์ต USB และปรับแต่งระดับเสียงได้เช่นกัน

I.2.1 การเลือกช่องสัญญาณเสียงเชื่อมต่อกับลำโพง

ผู้อ่านสามารถเชื่อมต่อสัญญาณเสียงกับลำโพงภายนอกผ่านช่องแจ๊ค 3.5 มม. หรือ ลำโพงของจอทีวี LCD ผ่านช่องสัญญาณ HDMI จากการทดลองต่อไปนี้

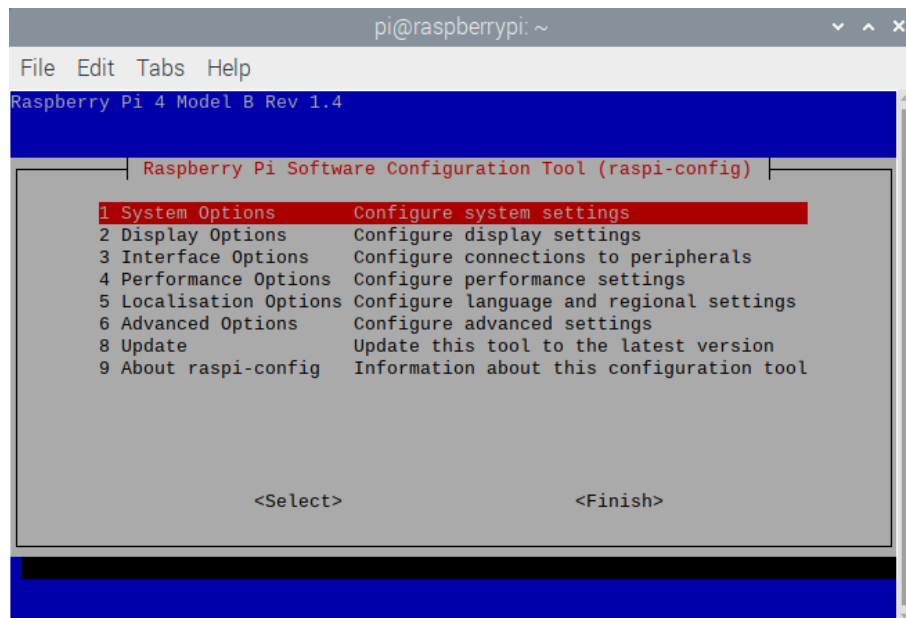
1. ใช้คำสั่งต่อไปนี้ในโปรแกรม Terminal

```
$ sudo raspi-config
```

จงบอกเหตุผลว่าคำสั่ง `sudo` ที่นำหน้ามีความสำคัญอย่างไร **ทำให้ user ที่ไม่มี permission สามารถเข้าถึงคำสั่งนี้ได้**

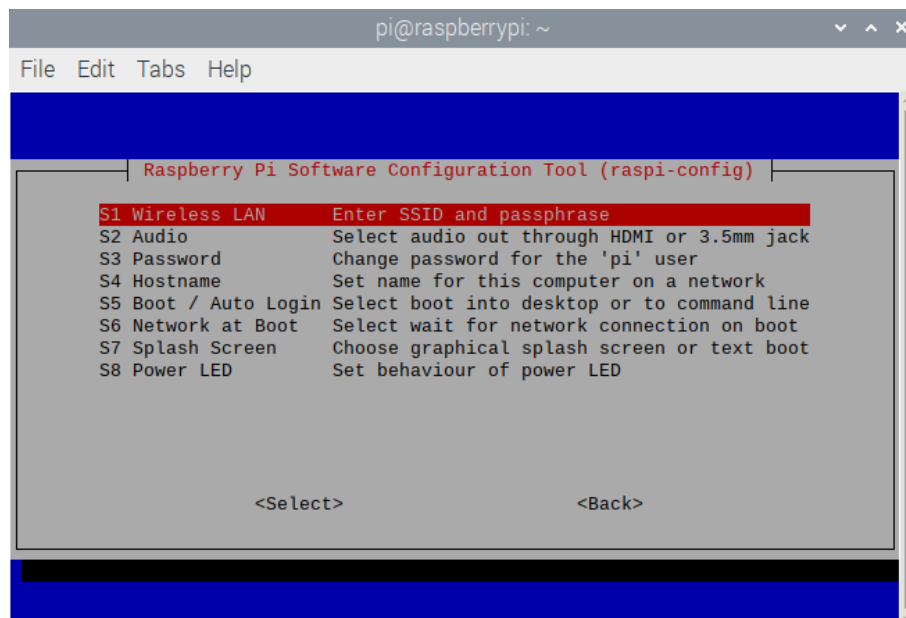
- กดปุ่มลูกศรขึ้นลงเพื่อเลือกเมนู System Options ในรูป

```
$ sudo raspi-config
```



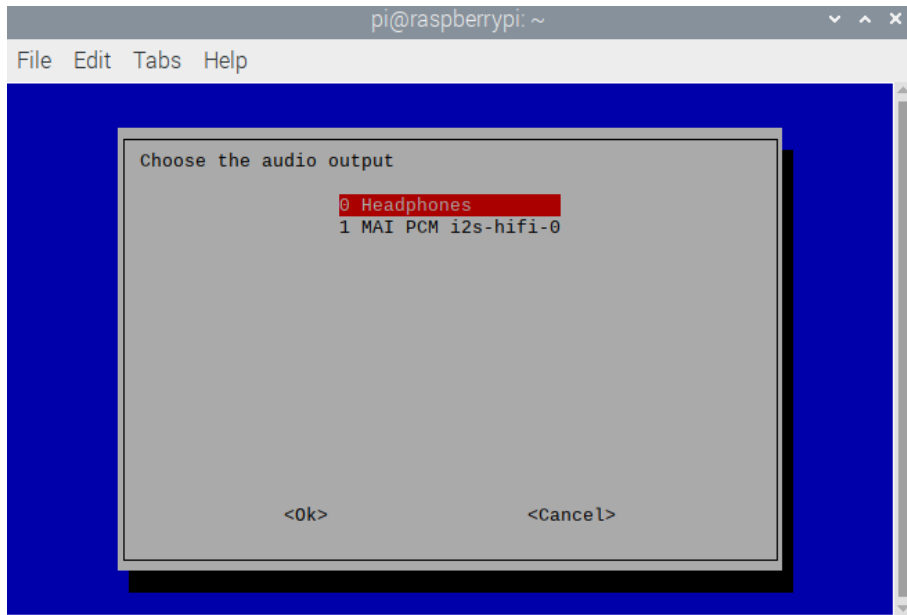
รูปที่ 1.7: หน้าต่างโปรแกรม raspi-config สำหรับบอร์ด Pi

- กดปุ่มลูกศรขึ้นลงเพื่อเลือกเมนู S2 Audio ในรูป



รูปที่ 1.8: เมนู System Options ในหน้าต่างโปรแกรม raspi-config สำหรับบอร์ด Pi

- กดปุ่มลูกศรขึ้นลงในรูปเพื่อเลือกเมนู 0 Headphones สำหรับแจ็ค 3.5 มม. หรือ 1 MAI PCM i2s-hifi-0 สำหรับพอร์ต HDMI

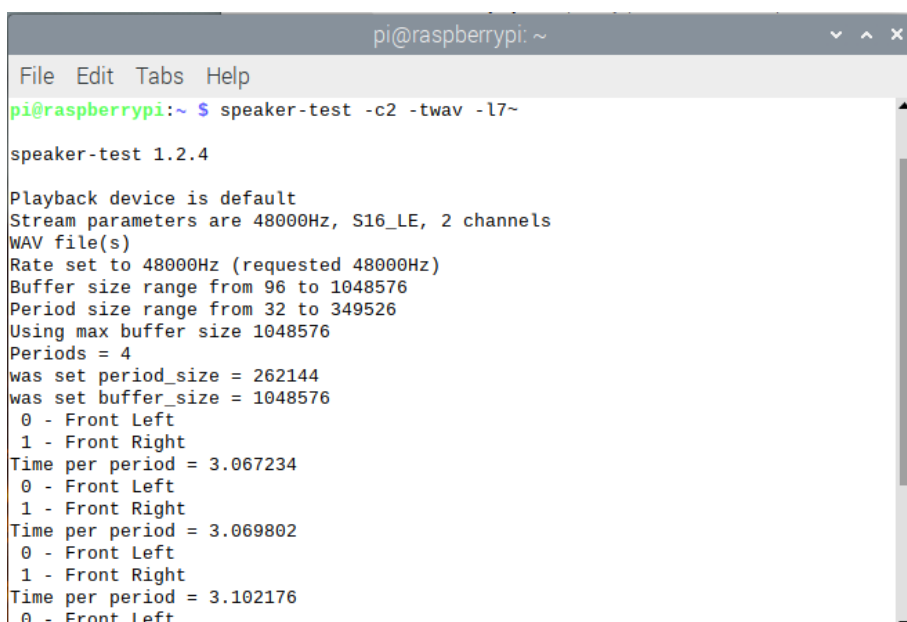


รูปที่ I.9: เมนู Audio ภายในเมนู System Options ในหน้าต่างโปรแกรม raspi-config สำหรับบอร์ด Pi

5. เมื่อเลือกเมนูที่ต้องการแล้ว กดปุ่ม Esc(ape) เพื่อถอยกลับ ออกมาจากเมนู และ กดจนออกจากโปรแกรม
6. ใช้คำสั่งต่อไปนี้ในโปรแกรม Terminal เพื่อทดสอบสัญญาณเสียงกับลำโพงที่เลือกต่อ

```
$ speaker-test -c2 -twav -l7
```

หากสำเร็จ ผู้อ่านจะได้ยินเสียงและผลลัพธ์คล้ายรูปต่อไปนี้



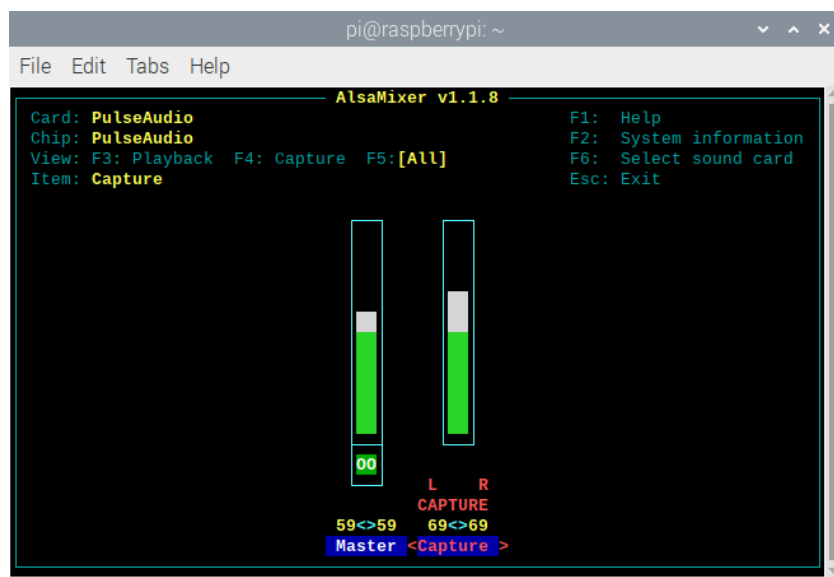
รูปที่ I.10: ผลลัพธ์ในหน้าต่างโปรแกรม speaker-test สำหรับบอร์ด Pi

1.2.2 การควบคุมระดับเสียง

นอกเหนือจากระดับเสียงที่ไอคอนรูปลำโพงด้านขวาบนของจอ ผู้อ่านสามารถควบคุมระดับความดังของเสียงทั้งด้านอินพุต (Capture) และเอาต์พุต (Playback) โดยพิมพ์คำสั่งนี้

```
$ alsamixer
```

หน้าต่างโปรแกรม alsamixer จะปรากฏขึ้น ผู้อ่านสามารถกดปุ่มลูกศรขึ้น/ลง เพื่อเพิ่ม/ลด ระดับความดังของ Playback ด้วยปุ่ม F3 ของ Capture ด้วยปุ่ม F4 และแสดงผลทั้งสอง ด้วยปุ่ม F5



รูปที่ 1.11: โปรแกรม ALSA Mixer สำหรับควบคุมระดับเสียงทั้งด้านอินพุต (Capture: F4) และเอาต์พุต (Playback: F3) บนบอร์ด Pi

1.2.3 รายชื่ออุปกรณ์ในระบบเสียง

ระบบเสียงในระบบปฏิบัติการ Linux ควบคุมการทำงานของเสียงผ่านระบบ ALSA (Advanced Linux Sound Architecture) ซึ่งจัดเตรียมไดรเวอร์ (Device Driver) สำหรับเสียงให้กับเคอร์เนล และอุปกรณ์ที่เกี่ยวข้องกับเสียงผ่านพอร์ต HDMI ช่องเสียบหูฟัง (Headphone) พอร์ต USB เช่น ไมโครโฟน, หูฟังพร้อมไมโครโฟน, เว็บแคม เป็นต้น

สำหรับการควบคุมอุปกรณ์เสียงขั้นสูง ผู้อ่านสามารถแสดงรายชื่อไฟล์หรือไดเรกทอรีที่เกี่ยวข้องกับระบบเสียงดังนี้

```
$ ls -l /proc/asound
```

```
total 0
lrwxrwxrwx 1 root root 5 Jun 21 19:48 b1 -> card0
dr-xr-xr-x 4 root root 0 Jun 20 19:45 card0
dr-xr-xr-x 4 root root 0 Jun 20 19:45 card1
```

```

-r--r--r-- 1 root root 0 Jun 21 19:48 cards
-r--r--r-- 1 root root 0 Jun 21 19:48 devices
lrwxrwxrwx 1 root root 5 Jun 21 19:48 Headphones -> card1
-r--r--r-- 1 root root 0 Jun 21 19:48 modules
dr-xr-xr-x 4 root root 0 Jun 21 19:48 oss
-r--r--r-- 1 root root 0 Jun 21 19:48 pcm
dr-xr-xr-x 2 root root 0 Jun 21 19:48 seq
-r--r--r-- 1 root root 0 Jun 21 19:48 timers
-r--r--r-- 1 root root 0 Jun 21 19:48 version

```

ผลลัพธ์ คือ รายชื่อ อุปกรณ์ ที่เกี่ยวข้องกับ เสียง ซึ่ง ได้ แสดง ไป ก่อน หน้า นี้ ผู้อ่าน จะ สังเกต ได้ ว่า ไดรเวอร์ /proc/asound/pcm จะเชื่อมโยงกับเนื้อหาในหัวข้อที่ 6.4 และเห็นว่ามีไดรเวอร์ชื่อ card0 อยู่สองตำแหน่งคือ ในแถวแรก และแถวที่มีชื่อ b1 -> card0 สัญลักษณ์ -> เรียกว่า **ซิมบอกลิงก์** (Symbolic Link) หมายความว่า ไดรเวอร์ชื่อ b1 คือไดรเวอร์ card0 ส่วนแถวที่มีชื่อ Headphones -> card1 สัญลักษณ์ -> เรียกว่า **ซิมบอกลิงก์** (Symbolic Link) หมายความว่า ไดรเวอร์ชื่อ Headphones คือ ไดรเวอร์ card1

1. ผู้อ่านสามารถค้นเพิ่มเติมโดยพิมพ์คำสั่งต่อไปนี้

```
$ cat /proc/asound/cards
```

บันทึกผลลัพธ์ในพื้นที่ว่างต่อไปนี้

```

pi@raspberrypi:~ $ cat /proc/asound/cards
0 [Headphones      ]: bcm2835_headpho - bcm2835 Headphones
                        bcm2835 Headphones
1 [vc4hdmi         ]: vc4-hdmi - vc4-hdmi
                        vc4-hdmi

```

2. ค้นคว้าว่า b1 และ Headphones คือ อุปกรณ์ใด ทั้งสองอุปกรณ์นี้แตกต่างกันหรือไม่

ต่างกัน b1 - HDMI

Headphone - jack 3.5 mm

3. ค้นคว้าเพิ่มเติมเพื่อหาความหมายของ Symbolic Link และจดบันทึก

shortcut ที่เชื่อมกับ file or folder ที่อยู่อื่น

ทำในกรณีที่การเรียกใช้งานกับไฟล์หรือโฟลเดอร์ที่ถูกเรียกใช้ไปในตัวเดียวกัน จะมีข้อผิดพลาดน้อย

4. พิมพ์คำสั่งนี้ในโปรแกรม Terminal

```
$ cat /proc/asound/cards
```

โดยคำสั่ง cat ซึ่งได้อธิบายแล้วในการทดลองที่ 4 ภาคผนวก D สามารถอ่านไฟล์และแสดงข้อมูลภายในไฟล์ผ่านทางหน้าจอแสดงผล บันทึกในที่ว่างต่อไปนี้

```
pi@raspberrypi:~ $ cat /proc/asound/cards
0 [Headphones      ]: bcm2835_headpho - bcm2835 Headphones
                        bcm2835 Headphones
1 [vc4hdmi          ]: vc4-hdmi - vc4-hdmi
                        vc4-hdmi
```

อธิบายผลที่ได้ ดังนี้ ผลลัพธ์ได้จากบอร์ด Pi4 ใช้ชิป BCM [2711](#) แต่ยังใช้ไดรเวอร์เสียงเดียวกันกับ BCM2835 โดย หมายเลข 0 คือ หมายเลขของระบบเสียงที่ติดตั้งใช้งานเพียงระบบเดียว และตรงกับอุปกรณ์ ชื่อ [card 0](#)

1.3 พอร์ตเชื่อมต่ออุปกรณ์ USB

1.3.1 รายชื่ออุปกรณ์กับพอร์ต USB

1. ในการทดลองนี้ ขอผู้อ่านให้ดึงหัวเชื่อมต่อ USB ของเมาส์ที่ใช้อยู่ ออก แล้วพิมพ์คำสั่งนี้ในโปรแกรม Terminal

```
$ lsusb
```

เพื่อแสดงรายชื่ออุปกรณ์ USB ที่เชื่อมต่ออยู่ทั้งหมดในบอร์ด ดังตัวอย่างต่อไปนี้

```
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 003: ID 046d:c534 Logitech, Inc. Unifying Receiver
Bus 001 Device 002: ID 2109:3431 VIA Labs, Inc. Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

ผู้อ่านจะเห็นรายชื่ออุปกรณ์ที่เชื่อมต่อกับพอร์ต USB เรียงลำดับย้อนกลับ จาก Bus 001 และ Bus 002 แล้วจึงเรียงจาก Device 003 - Device 001 โดย

- Bus 002 Device 001 คือ วงจร Root Hub เป็นวงจรภายในชิป BCM2711 สำหรับเชื่อมต่อพอร์ต USB 3.0 เพิ่มเติม สังเกตได้จากพลาสติกสีฟ้า มีหมายเลข ID = 1d6b:0003
- Bus 001 Device 003 คือ ตัวรับส่งสัญญาณคีย์บอร์ดและเมาส์ไร้สายของผู้เขียน มีหมายเลข ID = 046d:c534 ผลิตโดย บริษัท Logitech
- Bus 001 Device 002 คือ วงจร USB Hub สำหรับเชื่อมต่อพอร์ต USB เพิ่มเติม มีหมายเลข ID = 2109:3431 ผลิตโดย บริษัท VIA Labs
- Bus 001 Device 001 คือ วงจร Root Hub เป็นวงจรภายในชิป BCM2711 สำหรับเชื่อมต่อพอร์ต USB 2.0 เพิ่มเติม สังเกตได้จากพลาสติกสีดำ มีหมายเลข ID = 1d6b:0002

2. บันทึกผลลัพธ์ของผู้อ่าน

Bus 00_ Device 005 : ID = 3 1 5 1: 4 0 1 0

Bus 00_ Device 004 : ID = 0 4 6 d: c 5 4 7

Bus 00_ Device 003 : ID = 0 4 2 4: e c 0 0

Bus 00_ Device 002 : ID = 0 4 2 4: 9 5 1 4

Bus 00_ Device 001 : ID = 1 d 6 b: 0 0 0 2

3. ผู้อ่านย้ายคีย์บอร์ดหรือเมาส์จากพอร์ต USB 2.0 ไปยัง USB 3.0 แล้วแสดงรายชื่ออุปกรณ์ USB ด้วยคำสั่ง

```
$ lsusb
```

เช่นเดิม บันทึกเฉพาะผลที่เปลี่ยนแปลง

Bus 00_ Device 006 : ID = 3 1 5 1: 4 0 1 0

I.3.2 รายละเอียดการเชื่อมต่ออุปกรณ์กับพอร์ต USB

คำสั่งต่อไป คือ **dmesg** สามารถแสดงรายการทำงาน หรือ Log ของระบบปฏิบัติการว่าตั้งแต่เริ่มเปิดเครื่อง โดยคำว่า **dmesg** ย่อมาจากคำสั่ง “display message or display driver” ซึ่งเคอร์เนลได้บันทึกไว้ในบัฟเฟอร์ชนิดวงแหวน (Ring Buffer) ซึ่งข้อความตอนต้นจะถูกเขียนทับเมื่อบัฟเฟอร์เต็ม

1. รันคำสั่งนี้ แล้วเลื่อนหน้าต่างขึ้นไปตำแหน่งวินาทีที่ 0.000000

```
$ dmesg
```

2. จงเปรียบเทียบ ข้อความที่ผู้อ่านได้กับข้อความต่อไปนี้ ระบุตำแหน่งที่แตกต่าง และเขียนข้อความนั้นลงในผลการทดลอง

```
[ 0.000000] Booting Linux on physical CPU 0x0000000000 [0x410fd083]
[ 0.000000] Linux version 5.10.63-v8+ (dom@buildbot)
(aarch64-linux-gnu-gcc-8 (Ubuntu/Linaro 8.4.0-3ubuntu1) 8.4.0,
GNU ld (GNU Binutils for Ubuntu) 2.34) #1459 SMP PREEMPT
Wed Oct 6 16:42:49 BST 2021
[ 0.000000] random: fast init done
[ 0.000000] Machine model: Raspberry Pi 4 Model B Rev 1.4
[ 0.000000] efi: UEFI not found.
[ 0.000000] Reserved memory: created CMA memory pool at
0x000000001ac00000, size 320 MiB
[ 0.000000] OF: reserved mem: initialized node linux,cma,
compatible id shared-dma-pool
...
```

ใช้ผลการทดลองของผู้อ่านเอง เติมรายละเอียดใน ____ ที่เว้นว่างไว้ ซึ่งเรียงลำดับตามเหตุการณ์ที่ได้จากคำสั่ง \$ dmesg สัญลักษณ์ [xxxx.yyyyyy] แสดงลำดับที่เกิดขึ้นตามเวลา โดย xxxx คือเลขวินาทีตั้งแต่เคอร์เนลเริ่มทำงาน และ yyyyyy คือเศษวินาที ข้อความที่แสดงเป็น 0.000000 เนื่องจากเคอร์เนลอยู่ระหว่างการเริ่มต้น

- เริ่มต้นการบูตระบบปฏิบัติการด้วยแกนประมวลผลหมายเลข 0x0
- จัดบันทึกหมายเลขเวอร์ชันของลินุกซ์ของผู้อ่านโดยละเอียด 5. 15.61-v7+
- จัดบันทึกคำสั่งภาษาแอสเซมบลีเวอร์ชัน 7 บิต
- แสดงผลการตรวจจับว่าเป็นบอร์ด Raspberry Pi 3 Model B Rev 1.2
- cma ย่อ มา จาก Contiguous Memory Allocator สำหรับ ขบวนการ DMA เริ่ม ต้น ที่ แอดเดรส 0x 1ec00000 ขนาด 256 เมกไบต์
- ...

ในการทดลองนี้ ระบบสามารถตรวจจับอุปกรณ์ USB และติดตั้งไดรเวอร์ได้อย่างถูกต้องปราศจากข้อผิดพลาด

1. ผู้อ่านสามารถล้างบัฟเฟอร์โดยใช้คำสั่ง ต่อไปนี้

```
$ sudo dmesg -C
```

โดย -C คือ Clear เป็นคำสั่งเพิ่มเติมให้ dmesg ล้างข้อความในบัฟเฟอร์ออก โปรดสังเกต ตัว C พิมพ์ใหญ่ หลังจากนั้น ผู้อ่านทดสอบโดยการถอดเมาส์ออก แล้วเสียบกลับเข้าไปใหม่

2. ผู้อ่านจะต้องถอดและเสียบเมาส์กลับเข้าไปใหม่อีกรอบ
3. ผู้อ่านสามารถแสดงข้อความที่เพิ่มเข้ามาในบัฟเฟอร์ได้อีก โดยเรียกคำสั่ง

```
$ dmesg
```

4. จดบันทึกเฉพาะ 4 บรรทัดแรก

```
pi@raspberrypi:~$ sudo dmesg
[ 1892.992329] usb 1-1.2: USB disconnect, device number 8
[ 1894.341988] usb 1-1.2: new full-speed USB device number 9 using dwc_otg
[ 1894.480212] usb 1-1.2: New USB device found, idVendor=046d, idProduct=c547, bcdDevice= 4.02
[ 1894.480243] usb 1-1.2: New USB device strings: Mfr=1, Product=2, SerialNumber=0
```

5. อภิปรายผลลัพธ์ที่บันทึกได้ในพื้นที่ว่างต่อไปนี้

- USB ยกเลิกการเชื่อมต่อกับ device 8
- มี USB ใหม่ที่ต่อเข้า device 9 ด้วย dwc_otg
- เจอ USB ที่เพิ่งเสียบ
- บอกรหัสผู้ผลิต, รหัสสินค้า, version USB

ในการเชื่อมต่อพอร์ต USB หากระบบแจ้งชื่ออุปกรณ์โดยไม่มีข้อความผิดพลาด แต่อุปกรณ์นั้นยังไม่สามารถทำงานได้ แสดงว่าอุปกรณ์ขาดซอฟต์แวร์ซึ่งทำหน้าที่เป็นไดรเวอร์ ขอให้อ่านค้นหาจากหมายเลขประจำตัวของผู้ผลิต (idVendor) หากผู้ผลิตมิได้เปิดเผยซอฟต์แวร์ ผู้อ่านจำเป็นต้องดาวน์โหลดหรือคอมไพล์เองจากนักพัฒนารายอื่นแทน

I.4 พอร์ตเชื่อมต่ออินเทอร์เน็ตผ่านสัญญาณ WiFi และ Ethernet

I.4.1 รายชื่ออุปกรณ์เครือข่าย

1. ผู้อ่านสามารถ ตรวจสอบ รายชื่อ อุปกรณ์ สำหรับ เชื่อม ต่อ เครือ ข่าย ได้ จาก คำ สั่ง **ifconfig** ทาง โปรแกรม Terminal ตัวอย่างผลลัพธ์เป็นดังนี้

```
$ ifconfig
```

2. เติมข้อมูลหรือตัวเลขในช่องว่าง _ ที่เตรียมไว้ให้จากผลลัพธ์ที่ได้ต่อไปนี้ ซึ่งลำดับรายการอาจแตกต่างกัน

```
eth0: flags=4163<UP, BROADCAST, RUNNING, MULTICAST> mtu 1 5 0 0
      inet 161 . 246 . 5 . 51
      netmask 255 . 255 . 255 . 0
      broadcast 161 . 246 . 5 . 255
      ...
```

```
lo: flags=73<UP, LOOPBACK, RUNNING> mtu 6 5 5 3 6
      inet 127 . 0 . 0 . 1
      netmask 255 . 0 . 0 . 0
      inet6 ::1 prefixlen 128 scopeid 0x10<host>
      loop txqueuelen 1000 (Local Loopback) ...
```

```
wlan0: flags=4163<UP, BROADCAST, MULTICAST> mtu 1 5 0 0
      inet 10 . 66 . 12 . 30
      netmask 255 . 255 . 240 . 0
      broadcast 10 . 66 . 15 . 255
      ...
```

```
% ether b8 : 27 : eb : 15 : 14 : 95
```

3. โปรดสังเกตคำเริ่มต้นในแต่ละรายการ ค้นคว้า และกรอกรายละเอียดเพิ่มเติม ดังนี้

- eth0 หมายถึง ชื่อของ interface เครื่องข่ายบน computer ที่มาจาก ethernet ซึ่งเป็นการเชื่อมต่อจากสาย , 0 คือ ลำดับ interface ในเครื่อง
- lo หมายถึง ชื่อของ interface เครื่องข่าย loopback บน computer ซึ่งเป็นเครื่องข่ายเสมือน

- wlan0 หมายถึง *ชื่อของ interface เครื่องคอมพิวเตอร์ wireless LAN บน computer*

ผู้อ่านสามารถค้นคว้าเพิ่มเติมได้ที่หน้าเว็บต่อไปนี้

<https://www.tecmint.com/ifconfig-command-examples/>

I.4.2 การเปิด/ปิดอุปกรณ์เครือข่าย

1. ผู้อ่านสามารถเปิดอุปกรณ์ eth0 ได้ตามต้องการแล้วทำการตรวจสอบ ดังนี้

```
$ sudo ifconfig eth0 down
$ ifconfig
```

จดว่าข้อความใดที่บ่งบอกว่า eth0 ไม่ทำงานแล้ว *sudo ifconfig eth0 down*

2. ผู้อ่านสามารถเปิดอุปกรณ์ eth0 ได้ตามต้องการแล้วทำการตรวจสอบ ดังนี้

```
$ sudo ifconfig eth0 up
$ ifconfig
```

จดว่าข้อความใดที่บ่งบอกว่า eth0 ทำงานแล้ว *sudo ifconfig eth0 up*

3. ผู้อ่านสามารถใช้คำสั่ง ifconfig สำหรับปิด อุปกรณ์ wlan0 ดังนี้

```
$ sudo ifconfig wlan0 down
$ ifconfig
```

4. ผู้อ่านสามารถใช้คำสั่ง ifconfig สำหรับเปิด อุปกรณ์ wlan0 ดังนี้

```
$ sudo ifconfig wlan0 up
$ ifconfig
```

จดว่าข้อความใดที่บ่งบอกว่า wlan0 ทำงานแล้ว *sudo ifconfig wlan0 up*

5. นอกเหนือจากการเปิดปิดอุปกรณ์เครือข่าย ผู้อ่านสามารถตรวจสอบรายชื่อเครือข่าย WiFi ที่บอร์ดเคยเชื่อมต่อสำเร็จได้จากไฟล์ wpa_supplicant.conf ซึ่งจะบันทึกรายละเอียดต่าง ๆ ของการเชื่อมต่อ นั้น ๆ รวมถึงพาสเวิร์ด (password) โดยพิมพ์คำสั่งต่อไปนี้ในโปรแกรม Terminal

```
$ cat /etc/wpa_supplicant/wpa_supplicant.conf
```

บันทึกผลที่ได้โดยกรอกในช่อง _ เท่านั้น

```
network={
  ssid=" KMITL-WIFI "
  psk="*****"
  key_mgmt= NONE
}
```

- ssid หมายถึง **ชื่อ wifi**
- ssid ย่อมาจาก **service set identifier**
- psk ย่อมาจาก **Dynamic Pre-Shared Key**
- key_mgmt คือ **การเข้ารหัส**

ผู้อ่าน สามารถ ค้นคว้า เพิ่ม เติม ได้ที่ https://wiki.archlinux.org/title/wpa_supplicant

1.4.3 การตรวจสอบการเชื่อมต่อกับอินเทอร์เน็ตเบื้องต้น

เมื่อผู้อ่านเปิดและทำการเชื่อมต่อสำเร็จ แล้วจึงสามารถตรวจสอบการเชื่อมต่อในระดับชั้นเครือข่าย โดยใช้คำสั่ง ping ใน Terminal ดังนี้

```
$ ping <ip address or host name>
```

การตรวจสอบการเชื่อมต่อเบื้องต้น คือ การ ping ไปหาเราเตอร์ฝั่งต้นทางที่บอร์ดเชื่อมต่อ ผู้อ่านสามารถสืบค้นหมายเลขไอพีของเราเตอร์ที่ต้นทาง โดยสังเกตที่ inet ของ eth0 หรือ wlan0 ว่าเริ่มต้นด้วยหมายเลข 192.168.x.y ซึ่งเราเตอร์ต้นทางมักจะมีหมายเลข 192.168.x.1 หรือ 192.168.x.254

นี่เป็นตัวอย่างผลลัพธ์ที่ได้ของ คำสั่ง ping 192.168.1.1 ที่ผู้อ่านจะต้องเติมหมายเลขลงใน __ ที่เตรียมไว้ให้

```
PING 192.168. 1 .1 (192.168. 1 .1) 56(84) bytes of data.
```

```
64 bytes from 192.168. 1 .1: icmp__seq= 1 ttl=252 time= 6 . 33 ms
```

```
64 bytes from 192.168. 1 .1: icmp__seq= 2 ttl=252 time= 2 . 07 ms
```

```
64 bytes from 192.168. 1 .1: icmp__seq= 3 ttl=252 time= 2 . 45 ms
```

โดย 192.168. 1 .1 คือหมายเลขไอพีแอดเดรสของอุปกรณ์ที่คำสั่งจะส่งแพ็กเก็ต ICMP (Internet Control Message Protocol) ความยาว 64 ไบต์ไป แล้วรออุปกรณ์หมายเลขนี้ตอบกลับมายังบอร์ด Pi โดยจับเวลาตั้งแต่ส่งไปและรอตอบกลับมา ของแพ็กเก็ตลำดับที่ 1 (icmp_seq= 1) เป็นระยะเวลา 6 . 33 มิลลิวินาที ส่วน ttl= 252 ย่อมาจากคำว่า time to live หมายถึง เลขจำนวนเต็มที่ผู้ส่งกำหนดค่าอายุของ

แพ็คเก็ตที่สามารถเดินทางผ่านเครือข่าย หากตั้งไว้น้อยจะทำให้แพ็คเก็ตข้อมูลนี้อายุสั้นและอาจเดินทางไปไม่ถึงปลายทางเนื่องจากหมดอายุก่อน โดย `ttl=64` เป็นค่าปกติ

ผู้อ่าน จะสังเกตเห็นว่า ระยะเวลา มีค่า ตั้งแต่ 2.07 - 6.33 มิลลิวินาที ขึ้นอยู่กับคุณภาพ ของสาย Ethernet หรือ ความแรง ของสัญญาณ WiFi คุณภาพดี จะทำให้ ระยะเวลา สั้น กว่า หลังจาก ตรวจสอบ ว่าบอร์คสามารถ เชื่อม ต่อ กับเราเตอร์ ต้นทาง ได้ตาม ตัวอย่าง ก่อน หน้า ผู้อ่านสามารถใช้ ตรวจสอบ การเชื่อมต่อได้ว่า เราเตอร์ต้นทางสามารถ เชื่อม ต่อ กับเครือข่าย อินเทอร์เน็ต ได้สำเร็จหรือไม่ โดย Host name คือ ชื่อเซิร์ฟเวอร์ปลายทางที่จดทะเบียนโดเมนเนม (Domain Name) เรียบร้อยแล้ว เช่น `ping www.google.com`

I.5 กิจกรรมท้ายการทดลอง

1. จงค้นหาว่าความละเอียดของการแสดงผลผ่านพอร์ต HDMI ในหัวข้อที่ [I.1.1](#) เก็บบันทึกลงในไฟล์ชื่ออะไร เก็บ file ที่ `/boot/config.txt`
2. ใช้คำสั่ง `ifconfig` ปิดอุปกรณ์ `lo` แล้วใช้คำสั่ง `ping 127.0.0.1` ว่ามีการตอบสนองกลับมาหรือไม่ ปิดอุปกรณ์ `lo` แล้ว `ping` อีกรอบ จงอธิบายว่า `127.0.0.1` คือ อะไร
3. ใช้ คำ สั่ง `ping` เพื่อ ทดสอบ เราเตอร์ ที่ อยู่ ต้นทาง ของ ผู้อ่าน เช่น `ping 192.168.x.1` หรือ `192.168.x.254` โดย `x` มีค่าเท่ากับ 0, 1, 2, ... จนกว่าจะมีการตอบสนองกลับมา
4. ใช้คำสั่ง `ping` เพื่อตรวจสอบการเชื่อมต่อไปยัง `www.google.com`