

# ระบบส่งและรับสัญญาณเสียงแบบ Real-time ด้วย SRT และ GStreamer บน Raspberry Pi

## โครงสร้างระบบ

[อุปกรณ์ส่ง]

Raspberry Pi (Sender) -> เสียงจากไมโครโฟน/USB Audio -> GStreamer Pipeline -> srt-live-transmit -> เครือข่าย

[อุปกรณ์รับ]

## อุปกรณ์ที่ต้องการ

สำหรับทั้งสองเครื่อง

- Raspberry Pi 3B+ หรือใหม่กว่า (แนะนำ Pi 4 สำหรับประสิทธิภาพที่ดีกว่า)
- การ์ดเสียง USB (เช่น Behringer UCA202 หรือ Sabrent USB Audio)
- Raspberry Pi OS (64-bit เวอร์ชันล่าสุด)
- การเชื่อมต่อเครือข่ายแบบมีสาย (แนะนำ)

## ขั้นตอนการติดตั้งและกำหนดค่า

### 1. ติดตั้งพื้นฐานทั้งสองเครื่อง

# อัปเดตระบบ

```
sudo apt update && sudo apt upgrade -y
```

# ติดตั้ง dependencies

```
sudo apt install -y build-essential cmake git pkg-config libssl-dev \  
gstreamer1.0-plugins-base gstreamer1.0-plugins-good \  
gstreamer1.0-plugins-bad gstreamer1.0-plugins-ugly \  
gstreamer1.0-libav gstreamer1.0-tools gstreamer1.0-alsa \  
libgstreamer1.0-dev libgstreamer-plugins-base1.0-dev
```

### 2. ติดตั้ง SRT ทั้งสองเครื่อง

```
git clone https://github.com/Haivision/srt.git  
cd srt  
./configure --prefix=/usr/local  
make -j$(nproc)  
sudo make install  
sudo ldconfig
```

### 3. ตรวจสอบอุปกรณ์เสียงทั้งสองเครื่อง

```
# ตรวจสอบอุปกรณ์เสียง
```

```
arecord -l
```

```
aplay -l
```

```
# ตั้งค่าอุปกรณ์เสียงเริ่มต้น (如果需要)
```

```
# แก้ไขไฟล์ /etc/asound.conf หรือ ~/.asoundrc
```

#### การกำหนดค่าอุปกรณ์ส่ง (Sender)

##### 1. สร้างสคริปต์ส่งสัญญาณ

สร้างไฟล์ **sender.sh**:

```
#!/bin/bash
```

```
# กำหนดพารามิเตอร์
```

```
SRT_TARGET_IP="192.168.1.100" # เปลี่ยนเป็น IP ของเครื่องรับ
```

```
SRT_PORT="12345"
```

```
AUDIO_DEVICE="hw:1" # ตรวจสอบด้วยคำสั่ง arecord -l
```

```
LATENCY="250000" # หน่วย microseconds
```

```
# สร้าง GStreamer pipeline + SRT
```

```
gst-launch-1.0 -v \
```

```
  alsasrc device=$AUDIO_DEVICE ! \
```

```
  audio/x-raw,format=S16LE,channels=2,rate=44100 ! \
```

```
  queue ! \
```

```
  audioconvert ! \
```

```
  audioresample ! \
```

```
  voaacenc bitrate=128000 ! \
```

```
  mpegtsmux name=mux ! \
```

```
  fdsink fd=1 | \
```

```
  srt-live-transmit -v -b 1000000 -l $LATENCY \
```

```
  "file://con" "srt://$SRT_TARGET_IP:$SRT_PORT?mode=caller"
```

##### 2. ทำให้สคริปต์ทำงานได้

```
chmod +x sender.sh
```

##### 3. เริ่มระบบส่งสัญญาณ

```
./sender.sh
```

## การกำหนดค่าอุปกรณ์รับ (Receiver)

### 1. สร้างสคริปต์รับสัญญาณ

สร้างไฟล์ **receiver.sh**:

```
#!/bin/bash

# กำหนดพารามิเตอร์
SRT_PORT="12345"
AUDIO_DEVICE="hw:1"      # ตรวจสอบด้วยคำสั่ง aplay -l
LATENCY="250000"         # ควรตรงกับค่าของเครื่องส่ง

# สร้าง pipeline รับสัญญาณ
srt-live-transmit -v -l $LATENCY \
  "srt://:$SRT_PORT?mode=listener" "file://con" | \
  gst-launch-1.0 -v \
    fdsrc fd=0 ! \
    tsdemux name=demux ! \
    queue ! \
    aacparse ! \
    avdec_aac ! \
    audioconvert ! \
    audioresample ! \
    alsasink device=$AUDIO_DEVICE sync=false
```

### 2. ทำให้สคริปต์ทำงานได้

```
chmod +x receiver
```

### 3. เริ่มระบบรับสัญญาณ

```
./receiver.sh
```

## การปรับแต่งเพิ่มเติม

### 1. ปรับค่า SRT สำหรับเครือข่ายไม่เสถียร

เพิ่มพารามิเตอร์เหล่านี้ใน **srt-live-transmit**:

```
--maxbw 1000000 --pbkeylen 32 --mss 1316 --fc 10000 --rcvbuf 10000000 --sndbuf 10000000
```

## 2. ใช้ระบบบัฟเฟอร์ใน GStreamer

เพิ่ม queue ใน pipeline เพื่อป้องกันการหยุดชะงัก:

```
bash
```

```
Copy
```

```
queue max-size-buffers=0 max-size-bytes=0 max-size-time=2000000000
```

## 3. การตรวจสอบระบบ

ตรวจสอบการทำงานด้วยคำสั่ง:

```
bash
```

```
Copy
```

```
# ตรวจสอบ CPU
```

```
htop
```

```
# ตรวจสอบเครือข่าย
```

```
iftop -i eth0
```

```
# ตรวจสอบกระบวนการ
```

```
pgrep -l gst-launch
```

```
pgrep -l srt-live-transmit
```

การแก้ไขปัญหาเบื้องต้น

### 1. ปัญหาเสียงขาดหาย:

- เพิ่มค่า latency (-l parameter)
- ตรวจสอบการใช้งาน CPU
- ตรวจสอบความเสถียรของเครือข่าย

### 2. ปัญหาไม่พบอุปกรณ์เสียง:

- ตรวจสอบด้วย arecord -l หรือ aplay -l
- แก้ไขไฟล์ asound.conf หากจำเป็น

### 3. SRT connection failed:

- ตรวจสอบ firewall (sudo ufw allow 12345)
- ตรวจสอบว่า IP ถูกต้อง

การเริ่มต้นทำงานอัตโนมัติเมื่อบูตระบบ

สำหรับทั้งสองเครื่อง:

```
bash
```

```
Copy
```

```
# สร้าง systemd service
```

```
sudo nano /etc/systemd/system/srt-audio.service
```

เนื้อหาสำหรับเครื่องส่ง:

```
Copy
```

```
[Unit]
```

```
Description=SRT Audio Sender
```

```
After=network.target
```

```
[Service]
```

```
ExecStart=/home/pi/sender.sh
```

```
WorkingDirectory=/home/pi
```

```
User=pi
```

```
Restart=always
```

```
[Install]
```

WantedBy=multi-user.target

เนื้อหาสำหรับเครื่องรับ:

Copy

[Unit]

Description=SRT Audio Receiver

After=network.target

[Service]

ExecStart=/home/pi/receiver.sh

WorkingDirectory=/home/pi

User=pi

Restart=always

[Install]

WantedBy=multi-user.target

จากนั้น:

bash

Copy

sudo systemctl daemon-reload

sudo systemctl enable srt-audio.service

sudo systemctl start srt-audio.service

ระบบนี้จะให้การส่งสัญญาณเสียงคุณภาพดีด้วย **latency** ต่ำภายในเครือข่ายท้องถิ่น โดยสามารถปรับแต่งพารามิเตอร์ต่างๆ ให้

เหมาะสมกับสภาพแวดล้อมการทำงานจริงของคุณได้