Display-Audio Processor Interface Validation

Block Owner: Abdulla Al-Ansari
Date: Dec-07-2020

Design Details

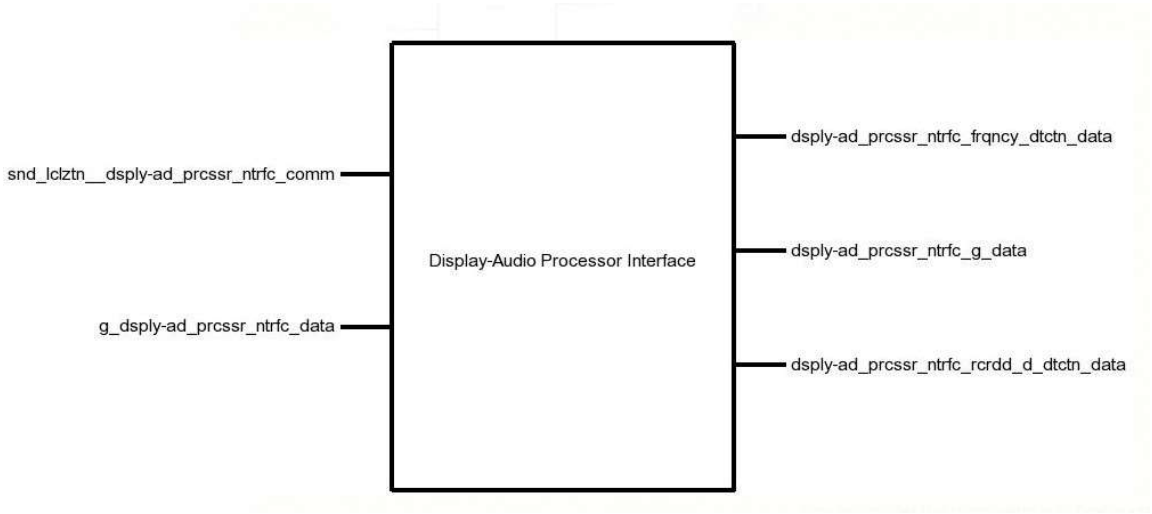

Figure 1: Black Box for Display-Audio Processor Interface

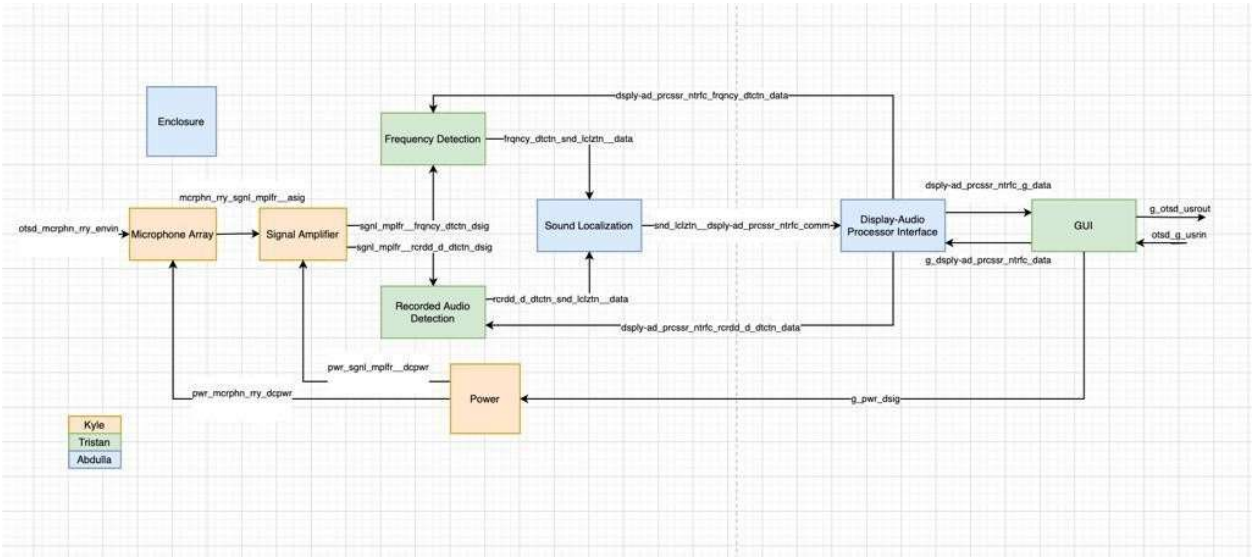

Figure 2: Top Level Block Diagram of Project

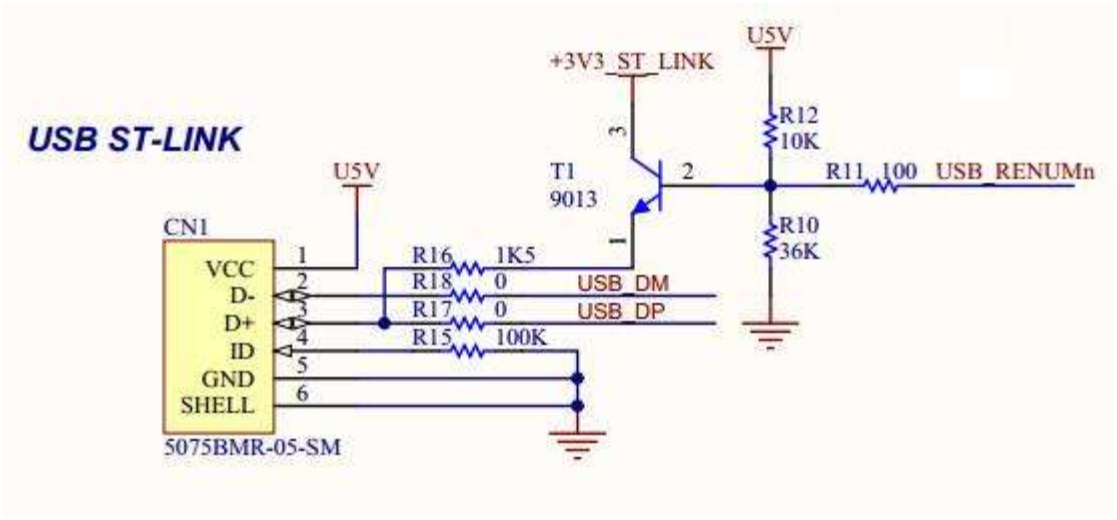

Figure 3: USB ST-LINK for STM32 Microcontroller (ref)

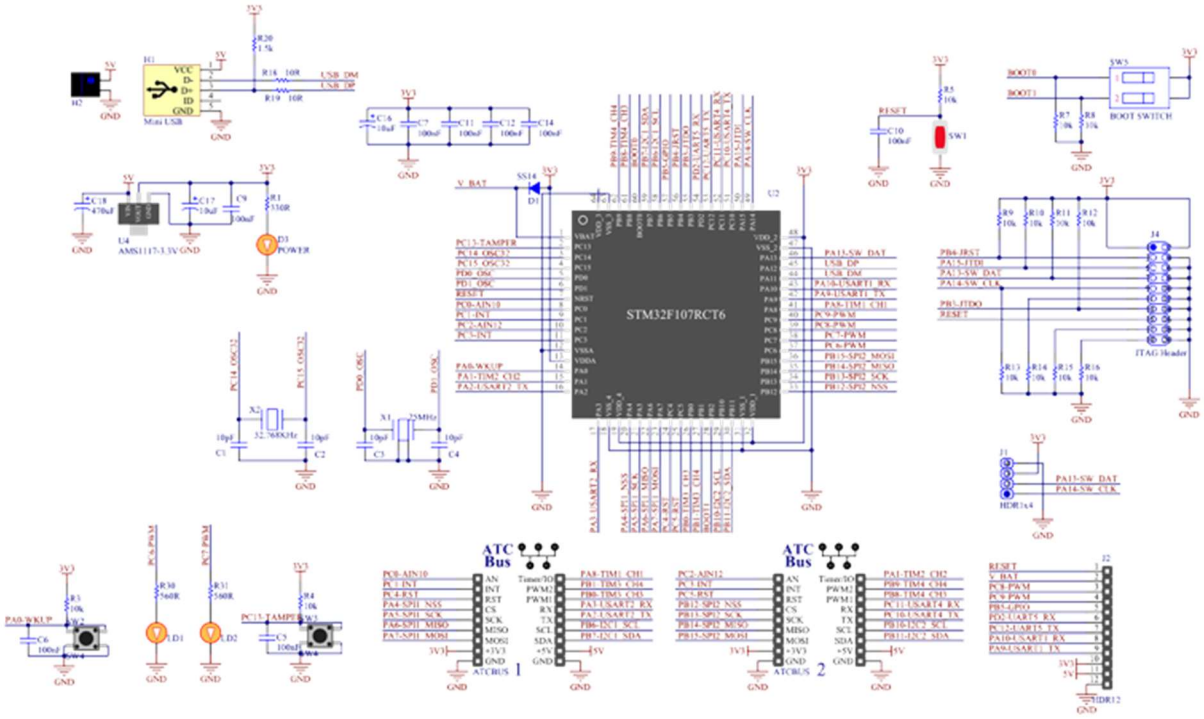

Figure 4: STM32 overall Schematic Diagram (ref)

Notes:

Data from processor, to be sent to GUI device, **snd_lctztn_dsply-ad_prcssr_ntrfc_comm**
USB Serial Protocol Send/Receive Commands, **g_dsply-ad_prcssr_ntrfc_data**
Output towards GUI for communication protocol
data, **dsply_ad_prcssr_ntrfc_frqncy_dtctn_data**
Data for GUI to display results, **dsply-ad_prcssr_ntrfc_g_data**
Recorded command for data transmission authorization, **dsply-ad_prcssr_ntrfc_rcrdd_d_dtctn_data**

Design Validation Overview

The block consists of a STM32 Microcontroller whose overall schematic is mentioned in Figure 4. This block will serve as the overall control block of the project as well as the main communication link between the controller and the android device. The block will make the communication channel through usb ST-LINK over hardwired usb cable from stm32 to the android. The android will serve as Host and will poll for data from the STM 32.

The communication design is based on a clone based on a paper , USB-Phone and other sources. The design consists of usb protocol commands from stm32 microcontroller in a low frequency kept on purpose which may be improved in future research on project.

The data will come from sound localization device which is also through code of STM32 in form of 2d plane coordinates. The data rate is kept on 576000 bauds per second transmission following Rules of Serial Communication. This may be increased further by testing higher data rates.

The data will be sent to a GUI device using usb serial communication protocol with specs defined in validation interface table.

Design Validation Interface Table

snd_lclztn__dsply-ad_prcssr_ntrfc_comm

Datarate: 57600 baud	The baud rates for usb serial communication range from 9600 and 115200. 57600 was taken as a suitable option by judgement call meeting the engineering requirement to allow fast but noise free data transmission.(Ref Rules of Serial)
Messages: (x,y) coordinates	The coordinates from the sound locating device to where the sound is coming from in 2d plane. Ref Engineering requirements and project scope. The data may be improved to 3d in future implementations following idea from paper.
Protocol: USB Serial	Bluetooth and USB serial were suitable and USB serial was preferred for the need of this project for fast secure communication.

	USB performance can be extremely low or very high, depending on the available USB version, which in turn, combined with the SDcard reading/writing speed on each device allows us to accurately measure and ascertain the transfer speed. Ref & Ref

dsply-ad_prcssr_ntrfc_frqncy_dtctn_data

Datarate: 10 Hz	Data rate for communication commands was lowered down on purpose to avoid false command transmission. STM32 support a clock speed of a hefty 120Khz (ref stm32datasheet) but the command send/receive frequency is kept at a minimum by judgment call.
Messages: Frequency range (20-20,000Hz), begin location command	The sounds frequencies to be analyzed by the device would be within human hearing range which is 20-20000Khz. Ref
Protocol: Function call	Function call within the code for data transmission protocol commands. This function generates the requisite commands for send/receive authorization between both hosts. Ref commands

dsply-ad_prcssr_ntrfc_g_data

Datarate: 10 Hz	Data rate for communication commands was lowered down on purpose to avoid false command transmission STM32 support a clock speed of a hefty 120Khz (ref stm32datasheet) but the command send/receive frequency is kept at a minimum by judgment call.
Messages: (x, y) coordinates	The coordinates from the communication processing device to GUI sending coordinates data in 2d plane for sound incoming.

dsply-ad_prcssr_ntrfc_rcrdd_d_dtctn_data

Datarate: 10Hz	Data rate for communication commands was lowered down on purpose to avoid false command transmission
Messages: Begin recording command, Stop recording command, Begin location command	Function call within the code to send/receive commands for sound recording. Once the sound snippet is stored, the stop recording command will be generated and then begin location command will issue authorization for sound localization on recorded snippet. Ref Engineering requirements
Protocol: Function call	Function call within the code for data transmission protocol commands. This function generates the requisite commands for send/receive authorization between both hosts. Ref commands

g_dsply-ad_prcssr_ntrfc_data

Datarate: 10 Hz	Data rate for communication commands was lowered down on purpose to avoid false command transmission. STM32 support a clock speed of a hefty 120Khz (ref stm32datasheet) but the command send/receive frequency is kept at a minimum by judgment call.
Messages: Record command (true/false), frequency selection (20-2000Hz), mode selection (recorded audio/frequency)	The sounds frequencies to be analyzed by the device would be within human hearing range which is 20-20000Khz. Ref
Protocol: Function call	Function call within the code for data transmission protocol commands. This function generates the requisite commands for send/receive authorization between both hosts. Ref commands

References

USB Serial Encoding schemes: https://www.engineersgarage.com/tutorials/signal-and- encoding-of-usb-system-part-5-6/
USB vs Bluetooth: https://android.stackexchange.com/questions/28686/what-is-the-fastest- way-to-transfer-huge-files-beween-two-android-powered-device
USB vs Bluetooth: https://smallbusiness.chron.com/usb-20-vs-bluetooth-47408.html Selection for frequency range to analyze: https://www.ncbi.nlm.nih.gov/books/NBK10924/#:~:text=Humans%20can%20detect%20sound s%20in,to%2015%E2%80%9317%20kHz.)
Send receive commands: https://learn.sparkfun.com/tutorials/terminal-basics/all
Sound localization: https://www.sciencedirect.com/science/article/pii/S1877705810010441 USB communication Info: https://guni91.wordpress.com/2017/08/13/usb-device-to-device-communication-via-stm32f407/
Engineering Requirements: https://nam01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fdocs.google.com%2Fdocument%2Fd%2F1rtj7cQx9B18AxznZRHtdkGpBiZvSbkBrMU44rtdmdoc%2Fedit%3Fusp%3Dsharing&data=04%7C01%7C%7C36bf91a35d7743e909bb08d89b4503a7%7C84df9e7fe9f640afb435aaaaaaaaaaaa%7C1%7C0%7C637430067995360876%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C1000&sdata=I6SEPmY6 u6CQVt8pjxoOdiPbS%2FEGyAqGPLXQ44oZHKk%3D&reserved=0
STM32 schematics diagram: https://www.atckey.com.vn/upload//images/Easy%20boards/ATC34_STM32F1%20Easy/SCHEMATIC_v3_800.png
USB ST-LINK Schematic diagram: https://community.st.com/s/question/0D50X00009XkY1uSAF/stm32f103-usb-circuit