MCU Architecture

The Milan architecture involves the Big+Little, with SoC (big) focusing on CPU/GPU heavy activities and MCU (little) focusing on low compute but always on activities. The baseline candidate for MCU is RT600.

Down select process can be found below:

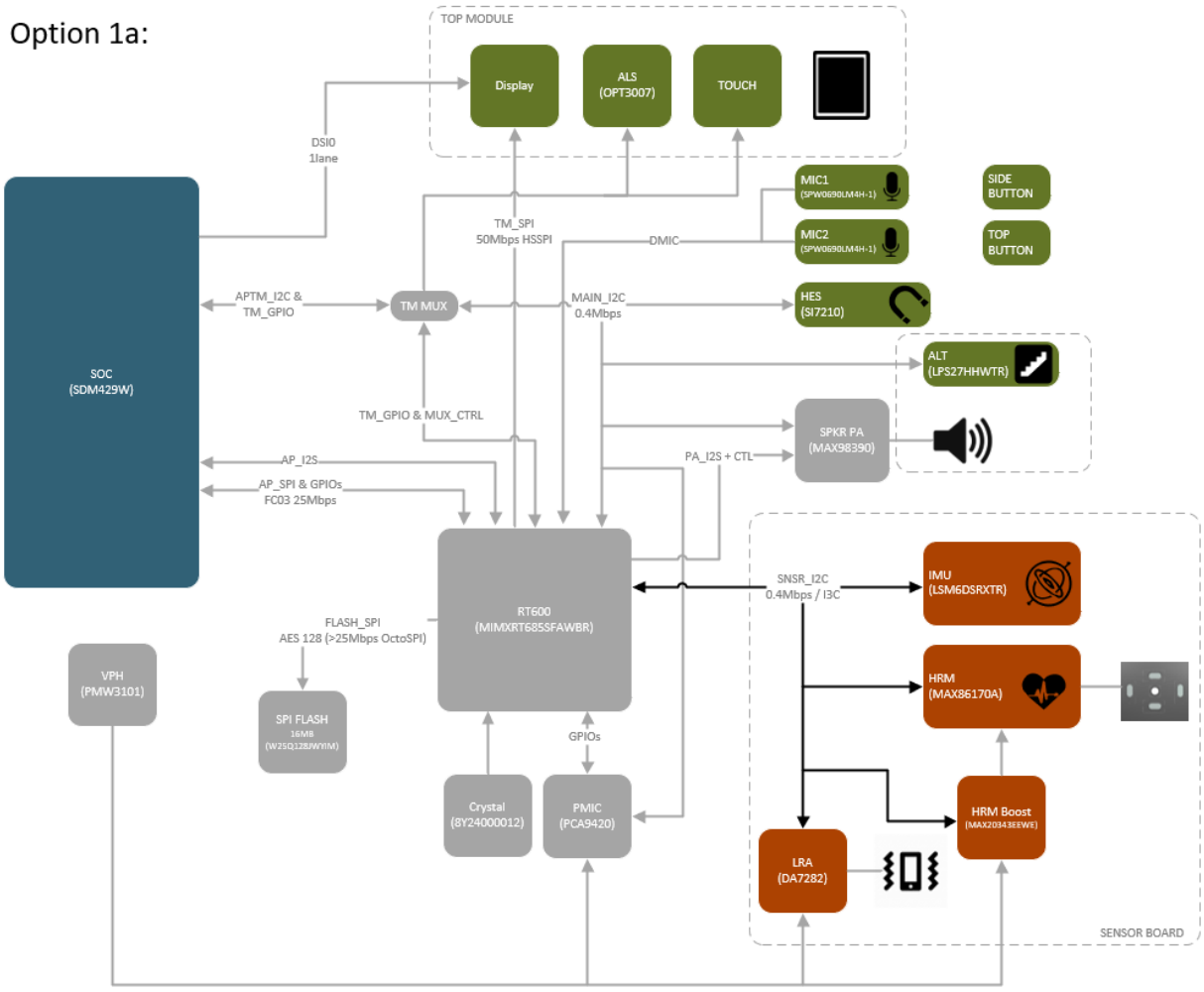
https://docs.google.com/presentation/d/1kLpMDTl2ay-51-weAGai1GSXMeXUQhwwBg6gvUCqHkE/edit#slide=id.g7e657a51c3_0_6

Functions

MCU helps to minimize system power consumption by assuming roles such as the sensor hub, audio processor, as well as the always on display driver.

Block Diagram

The diagram below shows the interfaces between the MCU and other subsystems.



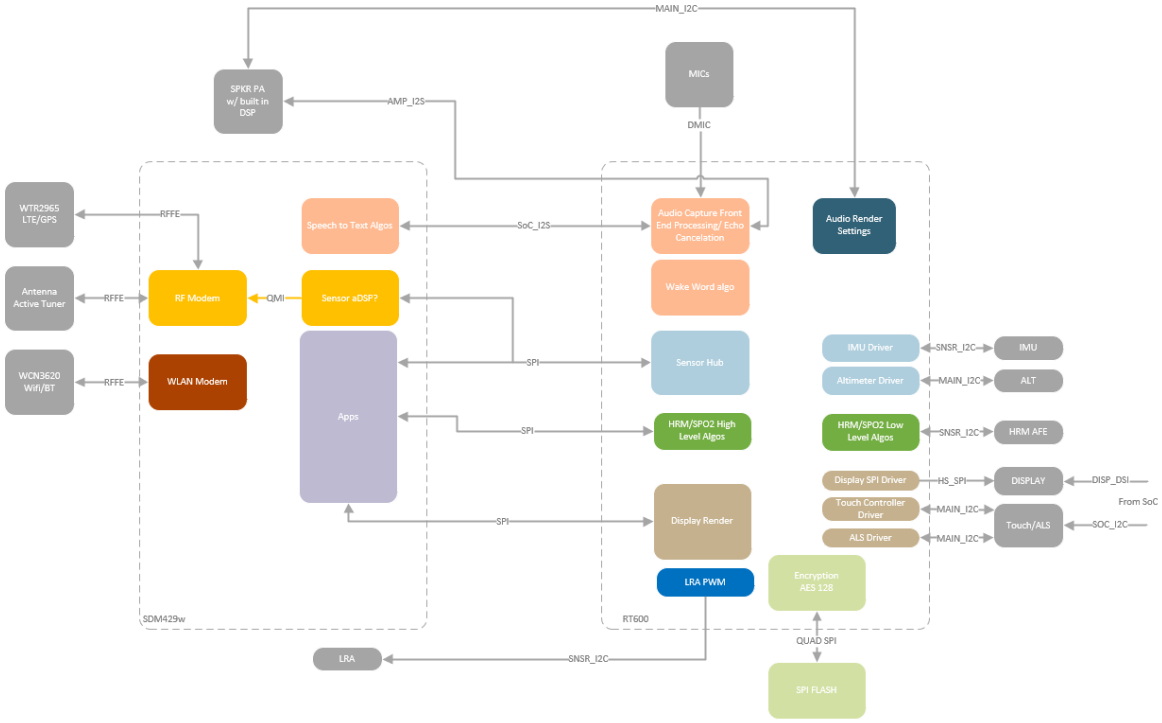
Detailed pint out can be found in this [document](), specifically, the digital interface mapping can be found below.

Flexcomm	0	UART_DEBUG
Flexcomm	1	I2S_MCU_TO_AMP
Flexcomm	2	I2S_SOC_TO_MCU
Flexcomm	3	SPI_SOC_TO_MCU
Flexcomm	4	I2S_SOC_TO_MCU
Flexcomm	5	I2C_MCU_MAIN (Touch/ALS/ALT/SPKR/PMIC/GG/HES)
Flexcomm	6	I2S_MCU_TO_AMP
FlexSPI HS		SPI_Display
FlexSPI Octo		SPI_Flash
I3C/I2C		I2C_MCU_SNSR (IMU,HRM,HRM BOOST,LRA)
USB		USBFS 2.0 DEBUG
SWDIO		SWD DEBUG

HW/SW RR

- Audio render (@Michael Asfaw)
- Audio capture front end (@Michael Asfaw third party DSPC?)
- Wake word algo (@Zhong Zhang, @Stanislav Sedov)
- Speech to text (@Zhong Zhang)
- Security (@Ali Pezeshk, @Marc Salem, @Stanislav Sedov)
- IMU / Altimeter drivers (Vendor TBD? @Amjad Obeidat)
- Sensor hub (@Rohan Mehta)
- HRM/SPO2 low level algos (@Michael Toksvig: Vendor TBD)
- HRM/SPO2 high level algos (@Michael Toksvig: Vendor TBD)
- Device Drivers (@Andrew Butler@Ric Wheeler)
    - Display SPI Driver (consult with Dalton/Stanislav)
    - ALS/Touch Controller Driver (consult with Dalton/Stanislav)
- Display Render (@Dalton Flanagan, Rohan Mehta)

High level algorithm distribution map on the MCU



Control/Handshake between SOC and MCU