



OA-II VEH Camera System Design

DR00002

Rev: A03
Jinzhi Cai
2019-07-24

Table of Contents

1	Introduction	3
1.1	Scope	3
1.2	Purpose	3
2	Revision History	3
3	General Structure of Camera System	4
3.1	Introduction	4
4	Camera Sensor and Relative Interface	5
4.1	Introduction	5
4.2	DVP Interface	6
4.3	LVDS Interface	7
4.4	MIPI CSI-2 Interface	8
4.5	USB Interface (UVC)	8
5	Camera Interface Bridge Chip	9
5.1	Lattice CrossLink	9
5.2	Toshiba Camera Interface Bridge	9
6	H.264 Video Steam Encoding	10
6.1	Introduction	10
6.2	IP Core Description	10
6.2.1	SoC Technology	10
6.2.2	A2E Technology	10
6.3	Encoding SoC Chip	11
6.3.1	HiSilicon Hi3559	11
6.3.2	OmniVision OV798	12
6.3.3	Ambarella H22	13
6.4	DSP Chip	14
6.4.1	TMS320C6678 DSP	14
6.5	GPU Module	15
6.5.1	Jetson Nano	15
7	File System and Video Storage	16
7.1	Introduction	16
7.2	Comparsion	16
8	System Diagram	17
8.1	USB camera with SoC FPGA chip	17
8.2	MIPI camera + CrossLink with FPGA chip	18
8.3	LVDS camera with FPGA chip	19

8.4	MIPI camera with Multimedia SoC chip	20
8.5	MIPI camera with Multimedia DSP and FPGA chip	21

1 Introduction

1.1 Scope

This document discuss the current camera technology and construct a design that will fulfill the need for OA-II VEH system.

1.2 Purpose

The goal for this document is come up with a design that will provide four 1080p 60Hz video steam and storage to a central storage media by research the current camera technology.

2 Revision History

Rev#	Editor	Delta	Date
A01	Jinzhi Cai	Initialize	2019-7-13
A02	Jinzhi Cai	Add detail	2019-7-15
A03	Jinzhi Cai	Add Jetson Nano	2019-7-24

Table 1: Summary of Revision History

3 General Structure of Camera System

3.1 Introduction

The most basic camera system have three part. The camera sensor is the device that will receive data from the environment and transfer it via the camera interface. The data that flow out from the sensor is call raw data. It contain all the information that the camera get from the environment. A 1080p 60Hz camera have $1920 * 1080 = 2073600 \text{ pixels}$. One pixels usually take 3 Bytes to save. For each second, $2073600 * 3 * 60 = 373248000 \text{ Bytes}$ will be created. That will be 355.96 MB/s for one single camera. The encoder is use to compress the video steam smaller for it to transmission via long distance data link(usually with in 1 MB/s). The storage unit is use to save the video steam to a file and allow future replay.

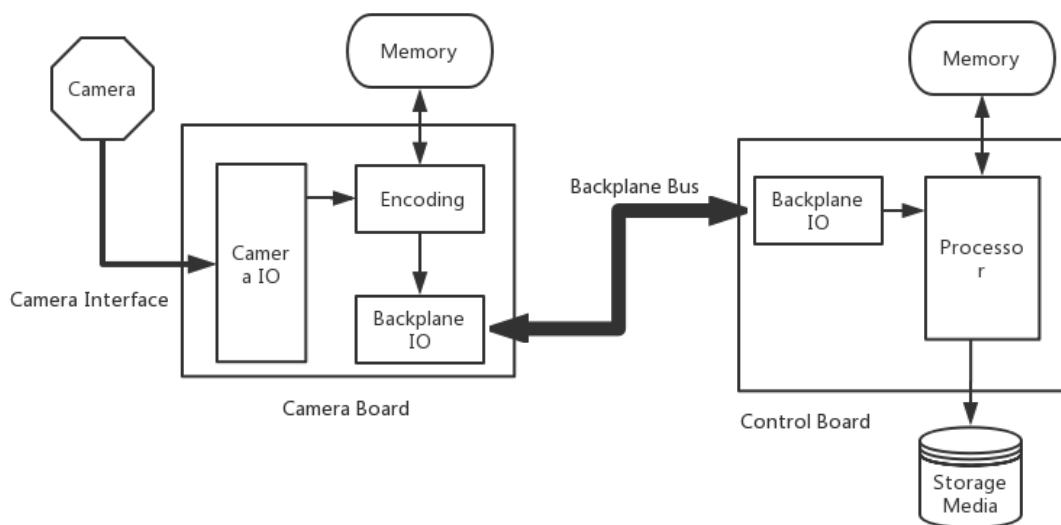


Figure 1: General Structure of Camera System

4 Camera Sensor and Relative Interface

4.1 Introduction

The camera sensor is the most important part in the camera system. The classic camera interface is used the same way to output the video signal as a monitor does. For each pixel clock, data of a pixel is being sent. After a few clocks (depend on the resolution), Line Sync rises to indicate one line of data is finished transmission. The Frame Sync indicates a frame of pixel data is sent. However, modern camera sensors usually use one of those three camera interfaces. Some of them are similar with the classic one, and others improve it.

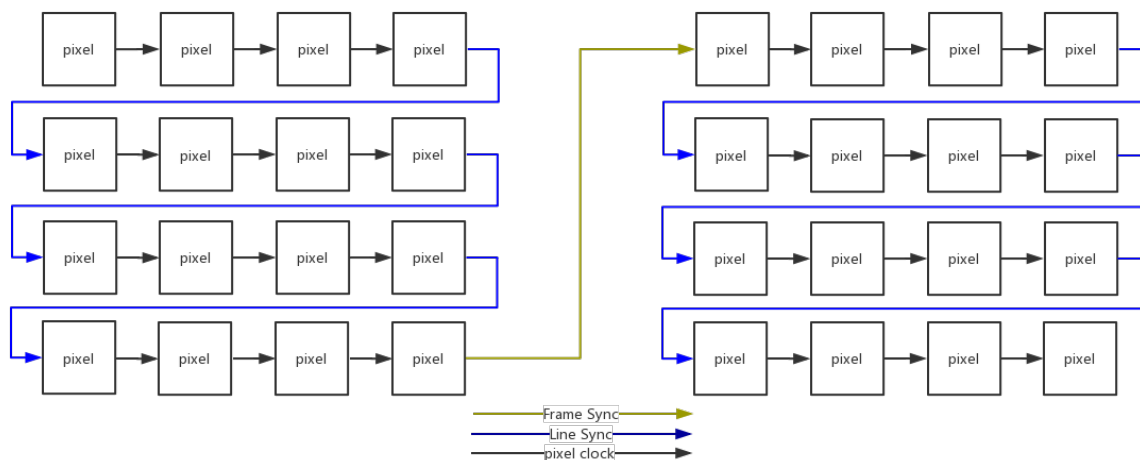


Figure 2: Classic Video Stream

4.2 DVP Interface

The DVP interface is use the classic video interface to transmit data out from the sensor. The I2C bus(SCK, SDA) that inside the interface is use to config the camera. The main clock is use to provide a clock for the camera. The VSYNC, HSYNC, PCLK is the clock system that help processor to indecate the weight and height of frame. The D OUT is data that comming out from the camera. It have low requirement on the trace and sample interface design. However, it can not offer long distance transmission and high speed clock. It usually use at low end 240p camera.

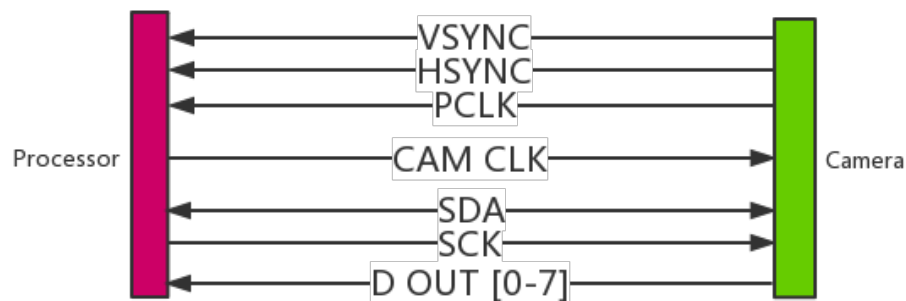


Figure 3: Physical Layer for DVP Interface

D OUT Data out put

VSYNC Lane Sync Signal

HSYNC Frame Sync Signal

PCLK Pixel Clock

CAM CLK Main Camera Clock

SDA I2C Data Line

SCK I2C Clock Line

4.3 LVDS Interface

The LVDS camera interface is use LVDS technology to improve classic DVP interface. It use differential signal to transmitting data. It increase the lane it need to send data, but differential signal can prevent common mode interference which limit the DVP clock for increasing.

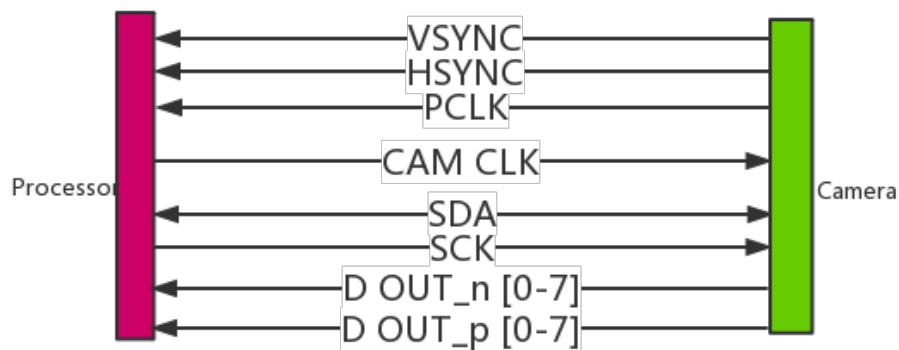


Figure 4: Physical Layer for LVDS Interface

DOUT n/p Differential Data Lane

VSYNC Lane Sync Signal

HSYNC Frame Sync Signal

PCLK Pixel Clock

CAM CLK Main Camera Clock

SDA I2C Data Line

SCK I2C Clock Line

4.4 MIPI CSI-2 Interface

The MIPI CSI-2 camera interface¹ is different from the old one. The goal for this bus is to cut down the lane count for camera module and allow more data flow to the processor. In structure, the MIPI CSI-2 camera interface use different way to communicate. The MIPI interface use package as the data communication unit. Inside the sensor, data will be pack into a package and send it via the two or four data lane. The same process will reverse and output the raw camera data. MIPI camera will allow 1080p 60Hz or 4K 30Hz.

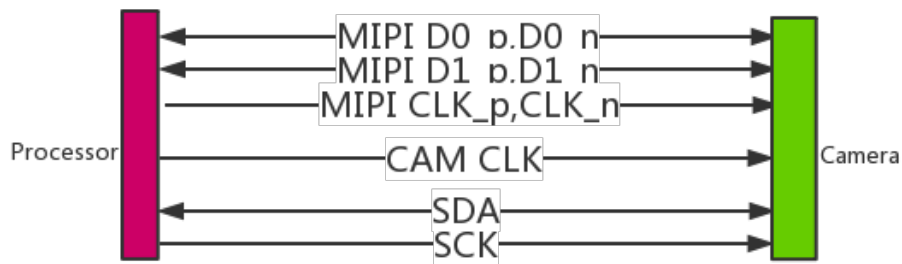


Figure 5: Physical Layer for 2 Lane MIPI Interface

MIPI D0n/p MIPI Data Lane 0

MIPI D1n/p MIPI Data Lane 1

MIPI CLKn/p MIPI Clock

CAM CLK Main Camera Clock

SDA I2C Data Line

SCK I2C Clock Line

4.5 USB Interface (UVC)

The USB camera interface² is different with the rest of the interface. In those camera, it include DSP chip that already finish the video compression. It is easy to use and support 60fps.³

¹ Might only available in high-end chip.

² It do have success example by using ZYNQ SoC FPGA.

³ With High End SoC chip

5 Camera Interface Bridge Chip

5.1 Lattice CrossLink

The CrossLink is Lattice made small scale MIPI bridge FPGA.⁴ It include two MIPI D-PHY and up to 8 mipi data lane. It also support interface translation (MIPI CSI-2 -> DVP).

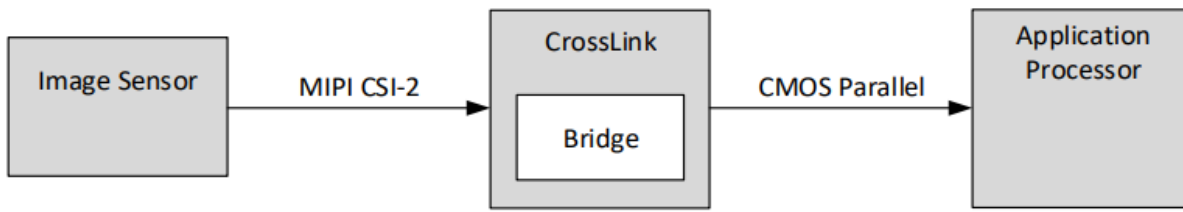


Figure 6: Block Diagram for CrossLink

5.2 Toshiba Camera Interface Bridge

The Toshiba Camera Interface Bridge is a stand alone chip that can converse MIPI CSI-2 bus to DVP bus. It also offer I2C bus to configure chip setting.

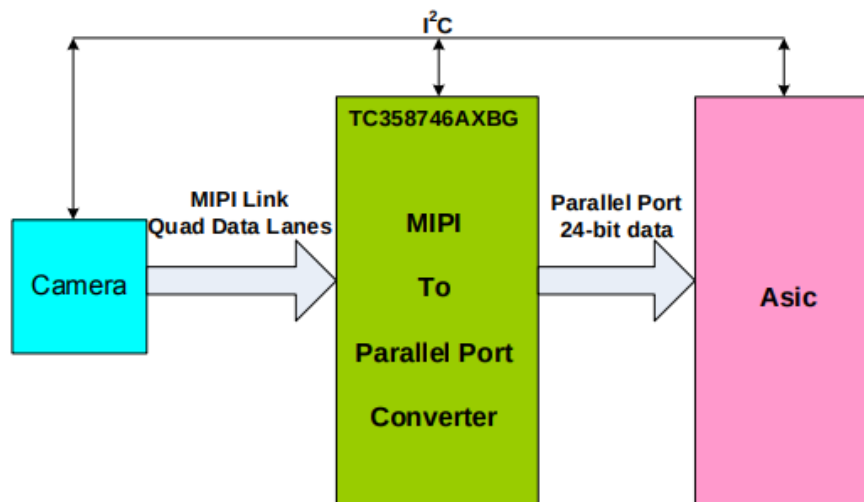


Figure 7: Block Diagram for Toshiba

⁴Too small for hardware encoder

6 H.264 Video Steam Encoding

6.1 Introduction

The H.264 Encoder is another important unit in the camera system. Due to the data flow of a camera sensor is too much for the backplane bus, compression is nessary.

6.2 IP Core Description

6.2.1 SoC Technology

This IP core fit for both **Intel** and **Xilinx**

Version	LUT use	Test Platform	Price for Platform
Standard Version	110k	Zynq-7 Z7045	\$1000
Slim Version	50k	Artix-7 XC7A200T	\$300
I-Frame Version	45k	Spartan-6 LX150	\$200

Table 2: SoC Summary

6.2.2 A2E Technology

Version	LUT use	Test Platform	Price for Platform
Xilinx(1080p 30Hz)	11K	Zynq-7 Z7020	\$300
Xilinx(1080p 60Hz)	11K	Kintex-7	\$500
Xilinx(1080p 180Hz)	11K	Zynq UltraScale	\$1000
Intel(1080p 30Hz)	8K	Cyclone V SoC	\$500

Table 3: SoC Summary

6.3 Encoding SoC Chip

6.3.1 HiSilicon Hi3559

Hi3559 is HiSilicon made new generation mobile camera soc. It support MIPI interface and H.264 Video encoding up to 1080P 60Hz. It can output video via USB 2.0 bus.

Functional Block Diagram

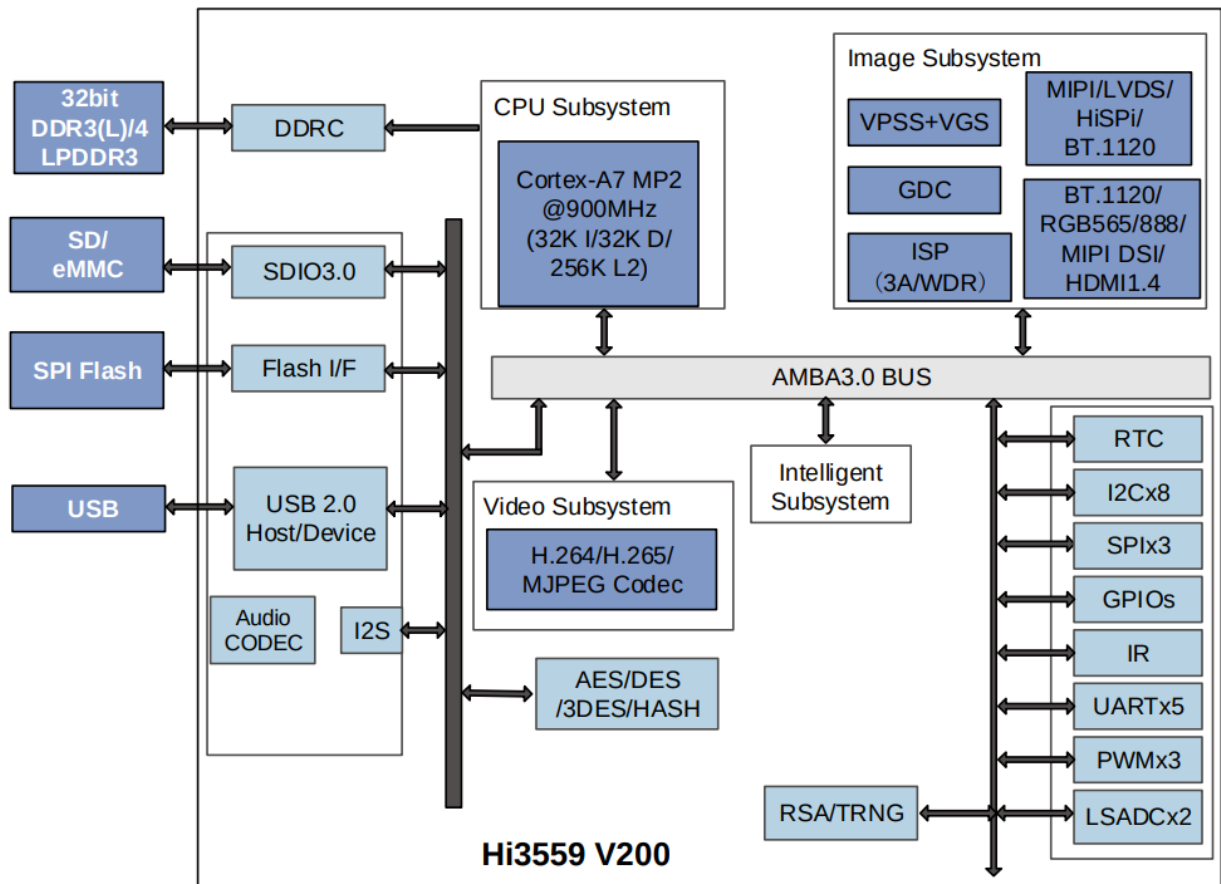


Figure 8: Block Diagram for Hi3559

6.3.2 OmniVision OV798

OV798 is OmniVision made camera soc. It support MIPI interface and H.264 Video encoding up to 720P 60Hz. It can output video via USB 2.0 bus.

Functional Block Diagram

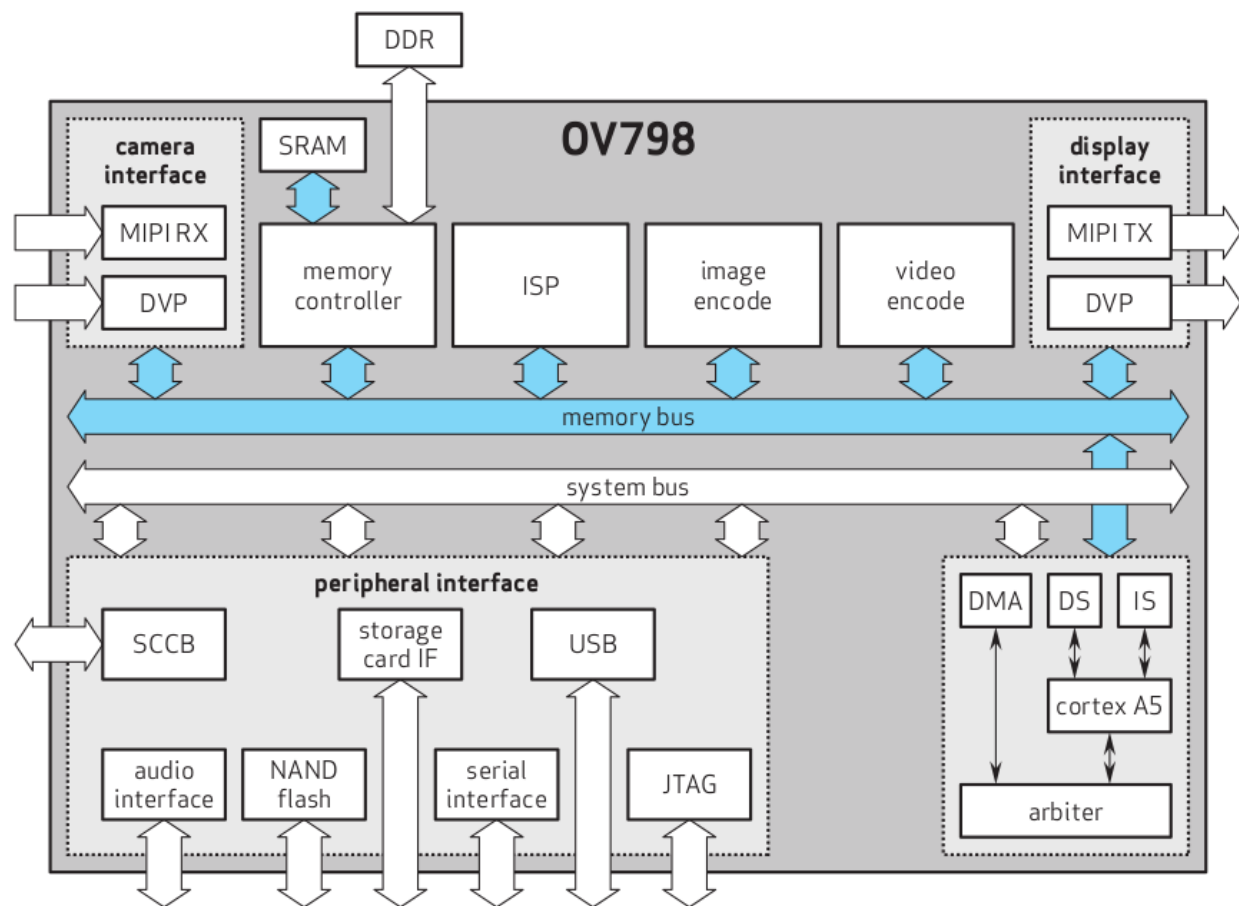


Figure 9: Block Diagram for OV798

6.3.3 Ambarella H22

The Ambarella H22 SoC for consumer applications is a system-on-chip that integrates an advanced image sensor pipeline (ISP), H.265 (HEVC) and H.264 (AVC) encoders, and a powerful Quad core ARM® Cortex™-A53 CPU.⁵

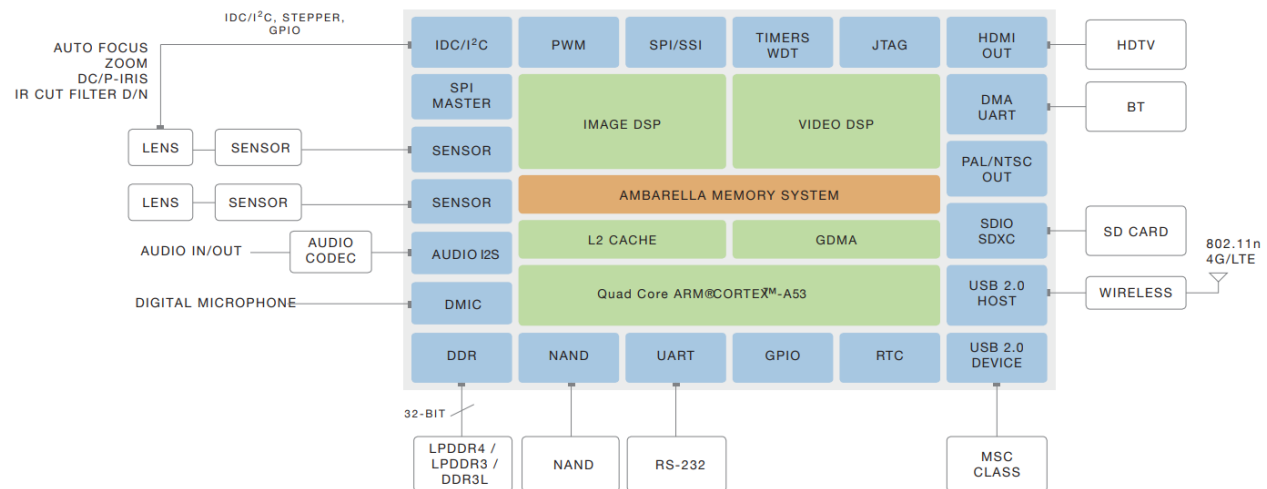


Figure 10: Block Diagram for Ambarella H22

⁵H22 Video SoC for Consumer Applications

6.4 DSP Chip

6.4.1 TMS320C6678 DSP

The TMS320C6678 DSP is a highest-performance fixed/floating-point DSP that is based on TI's KeyStone multicore architecture. Incorporating the new and innovative C66x DSP core, this device can run at a core speed of up to 1.4 GHz.⁶

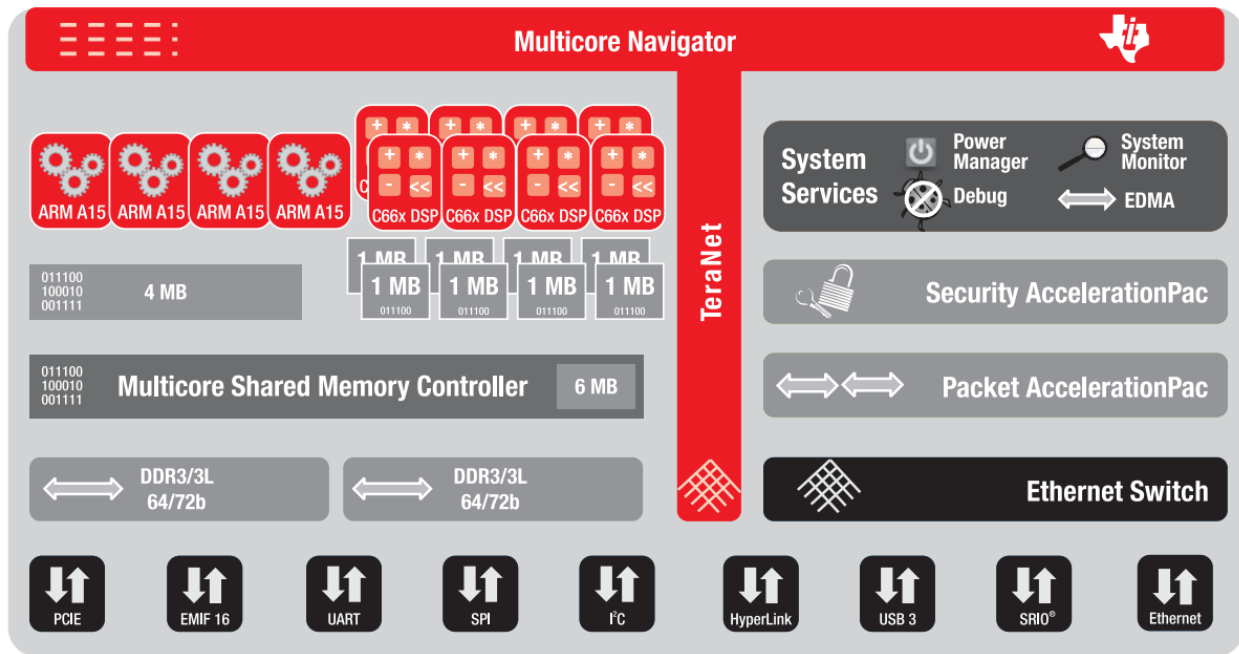


Figure 11: Block Diagram for Ambarella H22

⁶TMS320C6678 Multicore Fixed and Floating-Point Digital Signal Processor

6.5 GPU Module

6.5.1 Jetson Nano

The Jetson Nano is a GPU computing module what NVIDIA produce for GPU AI computer. It come with 6 MIPI CSI-2 interface and able to handle about 4K 60fps encoding. In the output option, it support PCIe output which can use as the interface. The only problem is it will require extra thermo solution due to the heat it produce and FPGA adoption program to translate the data to backplane bus.

7 File System and Video Storage

7.1 Introduction

The processor core selection will be down by another document⁷. In the camera system, the main focus in the storage media and the interface it had to connect to the processor. The whole OA-II VEH system will base on the embedded linux.⁸ So the main file system will be use ext4.

7.2 Comparsion

Technology	Interface	Speed	Complexity	Price
SSD Drive	NVMe	800MB/s	9	\$100/500GB
SSD Drive	SATA	715MB/s	9	\$100/500GB
eMMC	eMMC	45MB/s	5	\$10/128G
SD Card	SDIO	80MB/s	5	\$15/64G

Table 4: SoC Summary

SSD Drive The main problem for the SSD drive is this device need more research in order to intergrade into the VEH system. However, linux system do have exiting driver for them. It also have very high speed interface which allow fast data transformation.

eMMC The main problem for eMMC device is it is not removable from the board. It mean the recovery of this storge will be very difficult. However, it is a building interface for most of the SoC and it have no problem with vibration.

SD Card The main problem for sd card is the disconnection due to the vibration. It is the top choose due to its price and mobilability. In many mobile camera product, SD card is the storage media they use for multimedia.

⁷ES00005 - OA-II Payload Modules Architecture

⁸DR00003 - OA-II VEH COM System Design

8 System Diagram

8.1 USB camera with SoC FPGA chip

This plan will use USB 2.0 UVC interface to connect one camera to a SoC FPGA chip. The chip will translate the data to backplane bus and send to the storage device. It can use the builtin USB controller and system driver, which reduce the development time. It could upgrade to better camera set by using self developed module. But the builtin USB interface will limited the bitrate encoding use and might cause blur when the sensor meet frequently vibration.

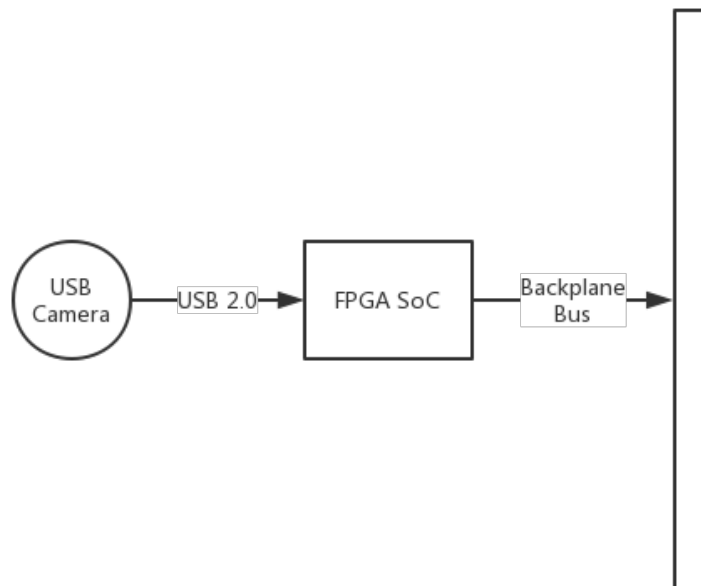


Figure 12: System Diagram for Plan A

Pro	Con
Easy for implement	Low Bitrate
Easy to find sensor	
Come with video encoding	

Table 5: The Pros and Cons Summary

8.2 MIPI camera + CrossLink with FPGA chip

This plan will use a crosslink chip to translate the MIPI signal to LVDS signal. The FPGA chip will receive the LVDS signal and use internal encoding IP core to process the data and send it out via the backplane bus. The CrossLink and the main chip are both FPGAs. It mean this system can easily increase the number of the camera. Four lane MIPI support up to 4K 180fps frame rate. It also allow future upgrade without change the hardware design due to the flexibility of FPGA chip.

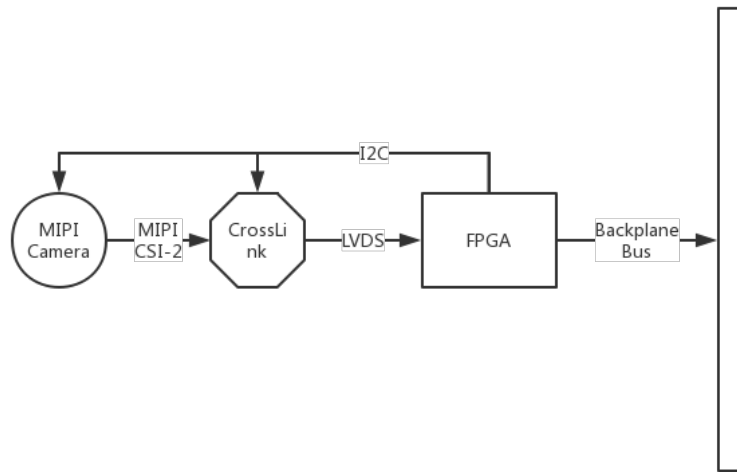


Figure 13: System Diagram for Plan B

Pro	Con
Can use any raspberry Pi Camera	Need Encoding Core
High Speed Sensor(Up to 180 FPS)	Hard in PCB design(100Ω impedance)
Allow futrue upgrade	Long development time
Prevent common mode interference	Need Extra chip for translation

Table 6: The Pros and Cons Summary

8.3 LVDS camera with FPGA chip

This plan will connect a lvds camera to a FPGA. The FPGA will encode the video and send it via the backplane. The video will be encoded in the FPGA IP core and pack it into the backplane bus packet. Because the camera sensor use the fpga support standard, it will reduce the problem of formate translation. It also allow future upgrade without change the hardware design due to the flexibility of FPGA chip.

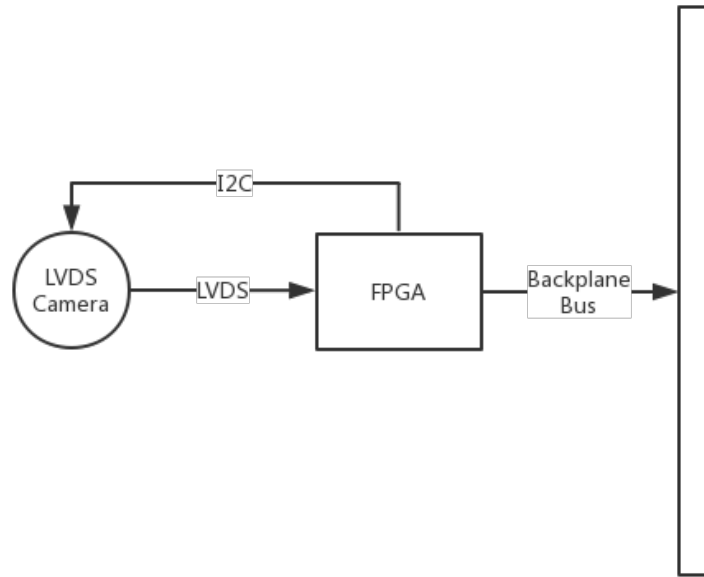


Figure 14: System Diagram for Plan C

Pro	Con
Do not need Extra chip for translation	Need Encoding Core
Cheap sensor	Hard in PCB design(100Ω impedance)
Allow futrue upgrade	Long development time
Prevent common mode interference	

Table 7: The Pros and Cons Summary

8.4 MIPI camera with Multimedia SoC chip

This plan will use the solution that company provide and add additional chip for the data translation from the USB to backplane bus. Because of the solution mostly will be provided by the company, it will be more stable. However, it lose the ability for upgrade. Also, because it use the MIPI bus, it require the PCB manufacture to use 100Ω impedance trace to keep the differential pairs stable.

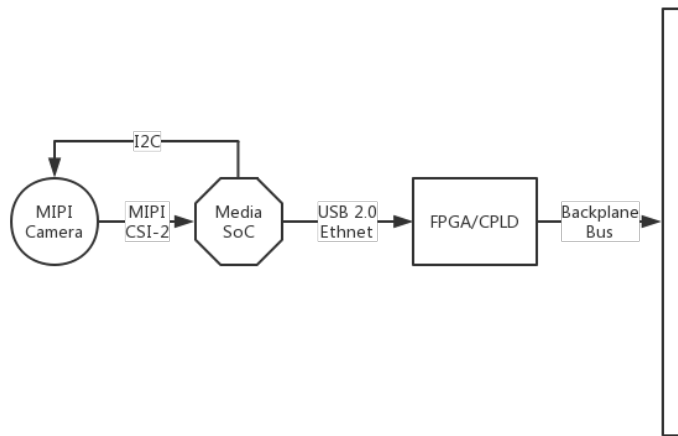


Figure 15: System Diagram for Plan D

Pro	Con
Do not need Extra chip for translation	Only support USB 2.0
Cheap sensor	Hard in PCB design(100Ω impedance)
Do not need Encoding Core	Long development time
Use existing resolution	Need selecting chip

Table 8: The Pros and Cons Summary

8.5 MIPI camera with Multimedia DSP and FPGA chip

This plan use DSP chip as the encoder. However the dsp do not have any video interface, it will need FPGA to finish the formate translation and feed it into the DSP chip. Also, because it use the MIPI bus, it require the PCB manufacture to use 100Ω impedance trace to keep the differential pairs stable. The DSP itself also have a set of SDK need to be take care off. It might require additional time to constructure the program.

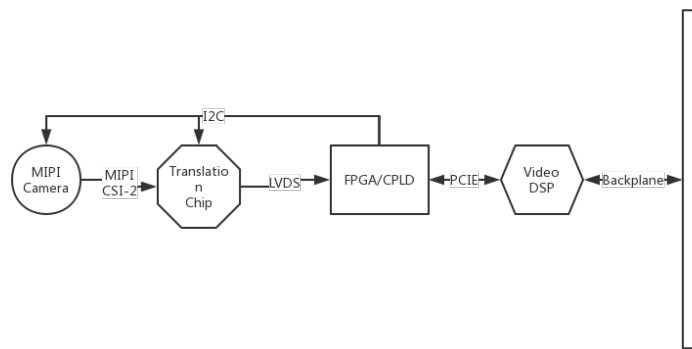


Figure 16: System Diagram for Plan F

Pro	Con
Do not need Encoding Core Use existing resolution	Need extra chip for translation and encoding Hard in PCB design(100Ω impedance) Long development time

Table 9: The Pros and Cons Summary