RoboCar: Vision Control Unit: Using Ultra96 Board and Autoware Stack

Ravikumar V Chakaravarthy
Yang Zhang

9/23/2020
The mission of the foundation is to initiate, grow, and fund Autoware projects, starting with Autoware.AI, Autoware.Auto, and Autoware.IO. Autoware has been adopted by more than 200 companies in 20 countries, and demonstration tests are progressing with various vehicles such as automobiles, trucks, automatic conveyors, and last one-mile vehicles.

AutoCore is the founding premium member of Autoware Foundation and the leader of autoware.io project, responsible for contributing and designing the HCP (Heterogeneous Computing Platform) as well as tooling used by Autoware adopters all over the world. Today AutoCore is the developer of the first Autoware Reference platform.

**AutoCore System Architecture**
Xilinx & AutoCore: Development Platform for Open Source Autonomous Driving

PCU:
1. LS1046A
   - RJ45*4
   - UART*2
   - USB*2
2. TMS570
   - CAN*4
   - UART*2
   - PPS
3. SJA1105
   - AVB*3

VCU (96Boards Standard):
1. Based on Ultra96
2. LVDS * 6
3. Support AR0231
4. Support AR0143
5. CAN*2
6. RJ45*1

The first 96Boards Mobility Platform
The first Autoware Reference Platform
The first open source SLAM Platform

Product Features
- Heterogeneous computing platform for Mobility Applications
- Auto-grade platform with CAN/UART/AVB, TSN capable Ethernet
- Design with functional safety, redundancy, built-in customisation
- Integrated development environment through development, debug, validation and deployment
- Continuously platform software updates and development SDK
- Fully validated stack with functional algorithm modules
- Open source application software stack

https://github.com/autocore-ai/autocore-pcu-doc

© Copyright 2020 Xilinx
Autonomous Driving Development Setup

CCU (P2)
1. DNN perception
2. Camera perception

PCU (P1)
1. Localization
2. Sensor Fusion

PCU (P3)
1. Vehicle control
2. Autonomous Driving Stack such as Autoware

IDE/Dashboard/ROS

Simulator

© Copyright 2020 Xilinx
Vision Control Unit: Solution

**Product Features**

- 96Boards HS/LS connectors compliant with all the 96Boards Mezzanine standard.
- 2*Quad GMSL Deserializer, from MAXIM Integrated (MAX9286)
- Support for 2*4 ON Semiconductor image sensors
- 2*4Ports input High-Speed FAKRA Mini (HSFM) connectors
- 2*FD CAN interface, SPI to CAN bridge from HS/LS connector
- UART to USB debug connector
- ETH port (Converting USB2.0 from HS connector to ETH bridge) connecting to HS connector CSI-2 through muxed with GMSL to CSI-2
- I2C interface
- DC12V adaptor input
Vision Control Unit: Software requirements

**Use cases**
- Surround vision using 4 side cameras (Vision Pipeline)
- Object detection using front facing cameras (AI Stack)

**Open source stack**
- **V4L and vision pipeline drivers**: Sensors, ISP, serializer, de-serializer, MIPI
- **Peripheral drivers**: CAN, Ethernet
- **OS and Embedded stack**: PetaLinux, Yocto, Linux kernel and Xilinx Embedded SW stack
- **AI Stack**: XTA
- **Middleware and Message queues**: ROSv2

**Performance requirements**
- 8-10 fps end to end performance

**Over the Air update**
- OTA SW update
- OTA runtime parameter update
Vision Control Unit: Capture pipeline

- Capture pipeline is based on V4L2 framework
- Solution supports up to 8 camera sensors
- 1 de-serializer handles up to 4 streams
- DMA-BUF used to capture and share sensor image data
Vision Control Unit: Image processing pipeline

- Hardware cores used: A53, R5, FPGA and GPU
- Drivers, Middleware and Libraries: V4L2, Heterogenous scheduler, JIT runtime, OpenGLES
- Application running on A53 and accelerated using heterogenous cores:
  - Fisheye de-warp, downscaling, image fusion, Bounding box accelerated using GPU
  - Pre and post processing data using A53 and R5
  - Yolov3 tiny processing accelerated using XTA core in FPGA
Vision Control Unit: XTA - Heterogenous AI stack

- Supports multiple frontend formats
- Provides good graph and tensor optimization capability. Graph packing and splitting capability to support heterogenous cores.
- Supports scalable, configurable and adaptable framework for using system resources (compute and memory). Parallel and pipelined processing using heterogenous cores.
- Supports Auto-Tuning capability to achieve optimal performance using HW backends.


© Copyright 2020 Xilinx
XTA: AI model to machine code

XTA workflow

Graph packing and splitting
XTA: Heterogenous pipeline

- AI Pipeline uses A53, R5 GPU and FPGA cores for processing.
- GPU, A53 Core 0 and R5 Core 0 is used for pre and post processing of data.
- A53 Core 0-2 is used to process Subgraph1.
- FPGA + A53 Core 3 is used to process Subgraph2.
XTA: Heterogenous pipeline performance analysis

Parallel processing using the heterogenous cores provides significant performance boost compared to sequential processing of cores.
Over the Air Update

- Use Petalinux Over the Air update infrastructure to do both component and package updates
- Updates to hardware design, Boot and Application stack.
- AI training parameter updates
Vision Control Unit: Demo
Conclusion

- Good collaboration between Xilinx, Linaro and Autocore to build an automotive reference platform to showcase vision perception unit using open source components

- Open source collaboration in Autonomous Driving with Autoware Foundation and 96Boards

- Solution provides a cost and performance efficient vision control unit

- Plan to further extend the processing power of Xilinx MPSoC to add more autonomous driving capabilities
Thank you

Accelerating deployment in the Arm Ecosystem