Hardware In Loop Project Report

M1W1

Current Progress

Work	Description	Status
Build Risc-V Core	Build Risc-V Core on the Nexys Vedio Board	Built two cores on the board, verified the gpio driver, uart driver, xsdb debug of the cores.
Build CAN Controller	Build two CAN controllers for each core	Built CAN controllers with open-core code, using 8051 interface
Build CAN Transceiver	Because the CAN controller will verify the data it has already transmitted in a fixed time, to implement the transmission between two CAN nodes on one board without any physical transceivers, we need to build a software transceiver.	Finished
Build Bare-metal code for cores	Build bare-metal software code to make the RISC-V IP possible to controll the CAN controller, then process the communication	Finished GPIO driver, uart driver, CAN init, transimit and receive driver
Verify CAN communication on board	Verify the driver code and the software IP we have already built	Finished the testbench simulation of CAN controller and transceiver, finished the bare metal software of RISC-V to transmit and recerive an extened format CAN pack

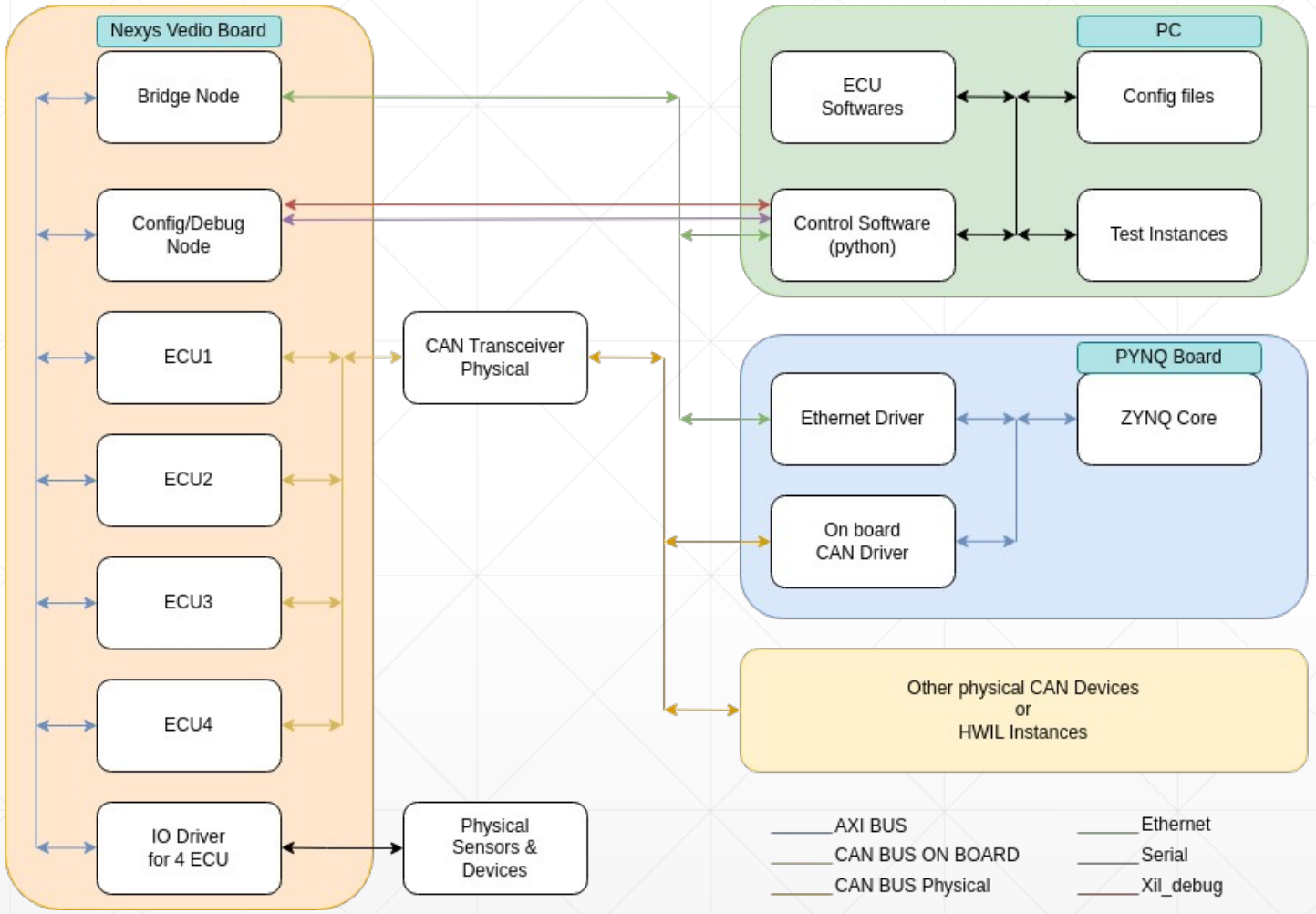
Next Step

Work	Description	Deadline
Build 8-cores Project	Because we need to simulate at least four ECUs at the same time ,current project with 2-cores is not enough, we need to rebuild the vivado project as a 8-core version	14, Jan
Check the Main Structure of the Project	Make an asumption of the structure of the whole project and check	14, Jan
Midterm Report	Make an outlilne of the midterm report and confirm after finished the structure of the whole system	Outline: 14, Jan Report:21, Jan
Verify physical CAN bus comm with a ZYNQ board	Get an pysical CAN transceiver and let the RISC-V core talk with the ZYNQ board with CAN bus	28, Jan
Build Ethernet interface	Buill up an ethernet interface for the board, for transmitting the status and command for the cores and PC	21, Jan

Needed Support

Support	Description
Check the main structure and main ideas	Could you please help me check the main structure and the main points of the system followed in the next slides and give me some advices
Hardware support	To check the communication between the ZYNQ board and the Nexys board, we need some hareware CAN transceivers and a ZYNQ board
Software download and data monitoring problems	At last, I think it will be necessary to monitor the status and send command and download the software to each RISC-V core with the scripts on the PC. I can only download the software of the core using xsdb command, I have no idea how to transmit these monitoring data and finish the programming automatically with PC software, could you please give me some advices
ZYNQ and desktop software demo	We discussed the software of ZYNQ and desktop before, and I want to know if there is a demo code for these two parts to excute a standard CAN test

Main Structure of the System



Main Ideas of the project

Ideas	Description
Build up the CAN communication between RISC-V cores	Build the main communication structure on the Nexys Vedio board and simulate
Build a ZYNQ core to monitoring and inject errors to the can bus line	Make it possible to inject errors to the CAN bus, and connect the Nexys board data to the PC, make it possible to use python scripts on the PC
Build desktop software	Build an UI desktop software to download program, monitor data, inject errors, excute auto testbench
Error injection and Speed up test	Use ZYNQ board to simulate error injections to the CAN bus like (DOS, Replay and Spoofting)
Error detection and solution	Use encryption algorithm or other methods to defeat the erros injection attact, maybe use some AI method to detect and predict the error status by monitoring the CAN bus data