ELECOMP CAPSTONE
VIRTUAL SUMMIT
FRIDAY, MAY 7, 2021

THE UNIVERSITY OF RHODE ISLAND

THINK BIG WE DO™

Network Control Signaling - Precise Timing Control via Ethernet

FARSOUNDER

Team Members: Scott Brennan (ELE), Justin Hall (CPE)

Technical Directors: Matthew Coolidge, Matthew Zimmerman

Project Motivation

FarSounder's sonar has a transmit system that must be synchronized with the receiver's data collection system. The receiver system electronics are stored in the in-water portion of the system (the Transducer Module). Transmit system electronics are stored in the on-board portion of the system (the Power Module). A custom bundled cable with ethernet, transmit signal, power supply, and DIO conductors connects the power module and transducer module. The cable currently being used by FarSounder is clunky, large, and heavy, making it difficult to install and more costly to ship. The cable also requires many potentially unnecessary parts that increase the price of the cable. For our project, we hope to reduce the size of the cable by removing the potentially unnecessary DIO conductors, and using the existing ethernet cable connection to transmit the necessary data (**Fig. 1**). Reducing the size of the cable and the number of conductors within it will allow for a cheaper and easier to use product.

Key Accomplishments

Communication Method Selection: An ethernet-based network protocol was selected that meets our timing requirements in terms of accuracy. This protocol was identified to be the Precise Timing Protocol (PTP) (**Fig. 2**). It offers the capability for a master-slave style clock synchronization that can be within 10μs of accuracy. It was also necessary to identify some peripheral communication methods for interfacing with other devices in the sonar system.

Microcontroller Selection: A microcontroller evaluation kit was selected that allows for prototyping an ethernet-based clock synchronization implementation and has capabilities to interface with other devices in the sonar system. The device is NXP's MIMXRT1060-EVK. This board features an ARM-based MCU, 100mbps Ethernet with support for the Precise Timing Protocol (PTP), multiple USB ports, and much more. Development for this platform is done with the MCUXpresso IDE.

Learned development for our platform of choice: Through the use of many included demos and functions available for the MIMXRT1060, a strategy for development was put together. This included learning how to demonstrate program execution without a debug host controlling the execution. In other words, show the board booting from a power source and running a program of our choice without the intervention of a PC or IDE. Other tasks included learning to write values to GPIO, as it may be done to send signals to other parts of the sonar system.

Demonstration of Network Activity: We identified a C library that allows for the implementation of a full TCP/IP stack used for sending and receiving data: the Lightweight IP library (lwip). Using this library and its included functions, we can ping the development board from an ethernet-connected PC.

Conversion of AN12149 Demo Project from IAR-EW to MCUXpresso: We found a demo project that provides the foundation for what we hoped to accomplish with this project. It allows for PTP clock synchronization between two boards connected via ethernet. This demo was originally for an IDE called IAR Embedded Workbench. In order to use this demo project, we had to first make several modifications to get it to work in our IDE of choice, MCUXpresso.

IEEE 1588 Implementation on two microcontrollers connected by ethernet: After getting the demo to work, we demonstrated a successful PTP clock sync between two MCUs, using one board as a PTP master and the other as a PTP slave. We confirmed the results by using an oscilloscope to watch the delay between the master and slave signals decrease with time (**Fig. 3**).

Determine how long it takes for the sync offset to reach 10μs: Our technical directors specified that a delay of 10μs was a good threshold to determine that the boards had achieved sufficient synchronization. We determined that synchronization generally takes an average of 30 seconds. However, it can take as little as 10 seconds and as long as 60 seconds.

Test Synchronization with more variables: We tested to ensure the clock synchronization method works with other nodes in the network. This was done by adding a switch in between two of the boards. In doing so, we found that the switch imposes additional clock offset (**Fig. 4**). The added transmission delay from the switch affects the accuracy to which the slave is able to synchronize, as the master's recommended adjustment becomes less accurate in keeping the slave's clock in time with the master.

Determine how much bandwidth is being used: We determined how much bandwidth of our 100mbps link is consumed by PTP traffic. This is to determine how much headroom there is for sending other data over the network connection, such as sonar data or other control signals. By using WireShark and an Ethernet hub to analyze the traffic between the two boards, we were able to find that the protocol consumes only about **5.88kbps** of bandwidth on average.

Develop a set of instructions so that our work can be easily replicated: We have developed a set of clear and concise instructions that FarSounder employees can use to expand upon our work. We have provided all necessary steps to operate the IDE software and recreate all of our experiments from throughout the year. This is to help FarSounder with using our work to create a new and improved sonar system.

Anticipated Best Outcome

As originally planned, our ABO is to have successfully produced the sample source code for the development boards that meets FarSounder's requirements and can be used as a reference design by FarSounder for implementation in a future version of their electronics. This solution will be an entirely ethernet-based method to synchronize the transducer module and the power module in time. By implementing this method, FarSounder should be able to remove the 4 DIO conductors from their custom bundled cable. Overall, we hope that our solution can be used in the future in order to create a cheaper and more efficient product.

Project Outcome

We have achieved the Anticipated Best Outcome of our project. We completed all necessary tasks to create the foundation for an ethernet-based method to synchronize the transducer module with the power module.

Figures

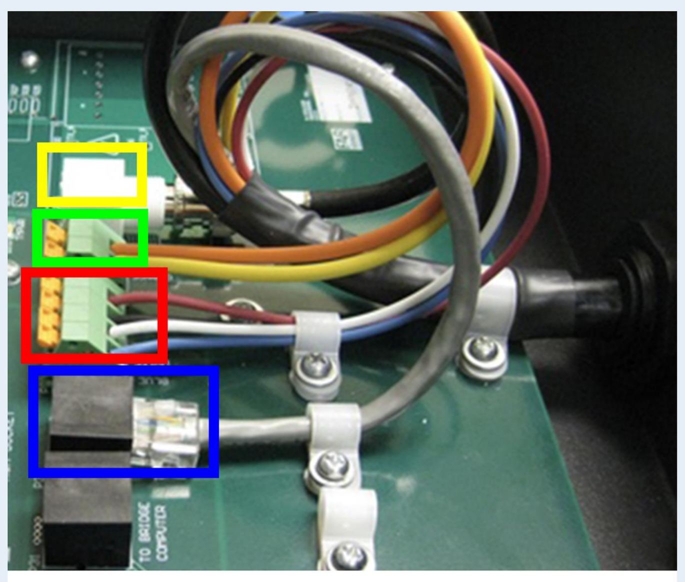

Fig. 1: FarSounder's currently used custom bundled cable

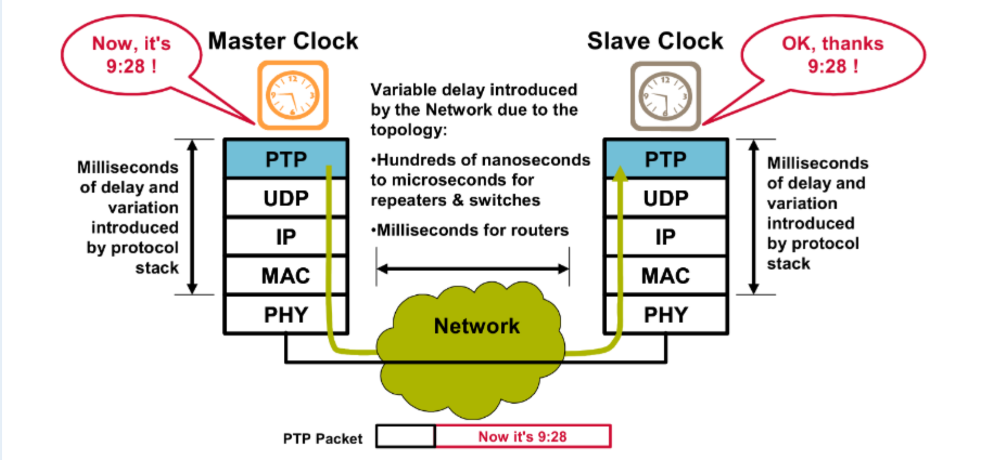

Fig. 2: Brief explanation of how PTP works on two microcontrollers connected via an ethernet network

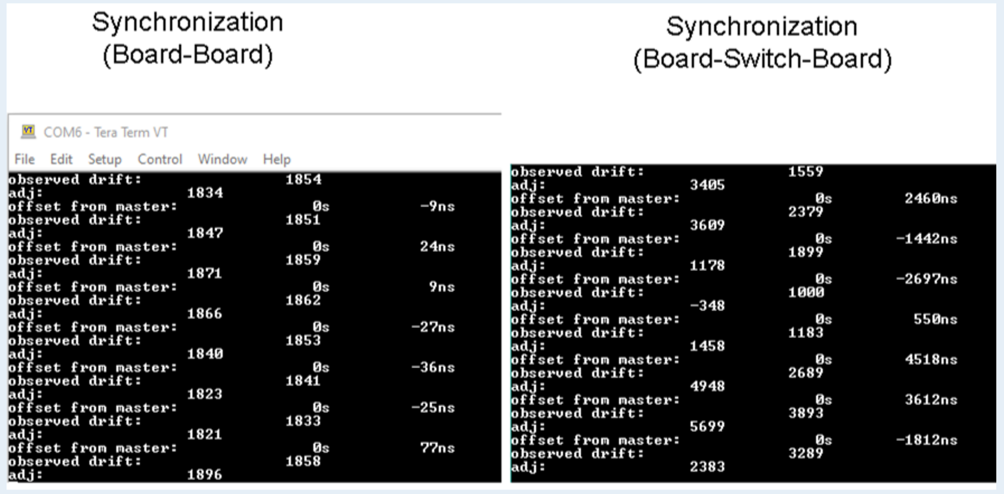

Fig. 3: Difference in achieved synchronization between a point-to-point connection and a connection involving an Ethernet switch.

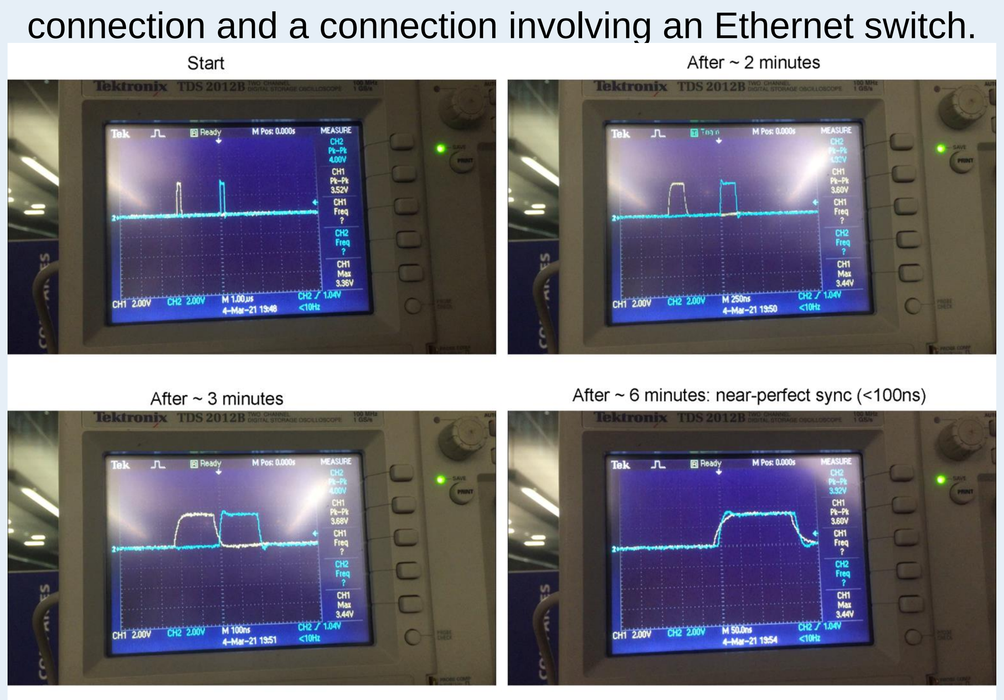

Fig. 4: PTP synchronization process shown on an oscilloscope