Hydrophones

We use our hydrophones to locate the pinger in the pool, which can be utilized for localization purposes.

Introduction

The purpose of this page is to describe how the arrival times are acquired on a number of hydrophone channels. This documentation does *not* describe how that information is turned into bearings or how it is used in localization.

The arrival time calculations tend to be complicated by noise, which means that it is not possible to trivially detect rising edges. The ping can be thought of as an identical signal arriving slightly later on each of the hydrophones. In order to accurately calculate the time differences between each signal, a cross correlation is used. A cross correlation essentially slides one signal over the top of another, and at the point where the signals maximally overlap, the amount of shifting needed is equal to the time delay of the signal.

Further simplifications can be applied to the problem as well. By ensuring that the hydrophones are close enough, it can be guaranteed that the arrival time difference will never exceed one half wavelength. Therefore, the cross correlation only needs to be completed within +/- half a wavelength. The signals occur at a maximum frequency of approximately 40KHz, which means that the arrival time difference can never exceed approximately 12.5 microseconds. This means that the hydrophones must be spaced within 1.8cm of each other due to the speed of sound in water.

Now that the theoretical aspects are out of the way, the foundation of the problem starts to take shape. 1) Acquire signals on all four hydrophones (one reference and 3 along each X, Y, and Z axis) 2) Cross-correlate the X, Y, and Z axis signals against the reference to calculate time of flight delay.

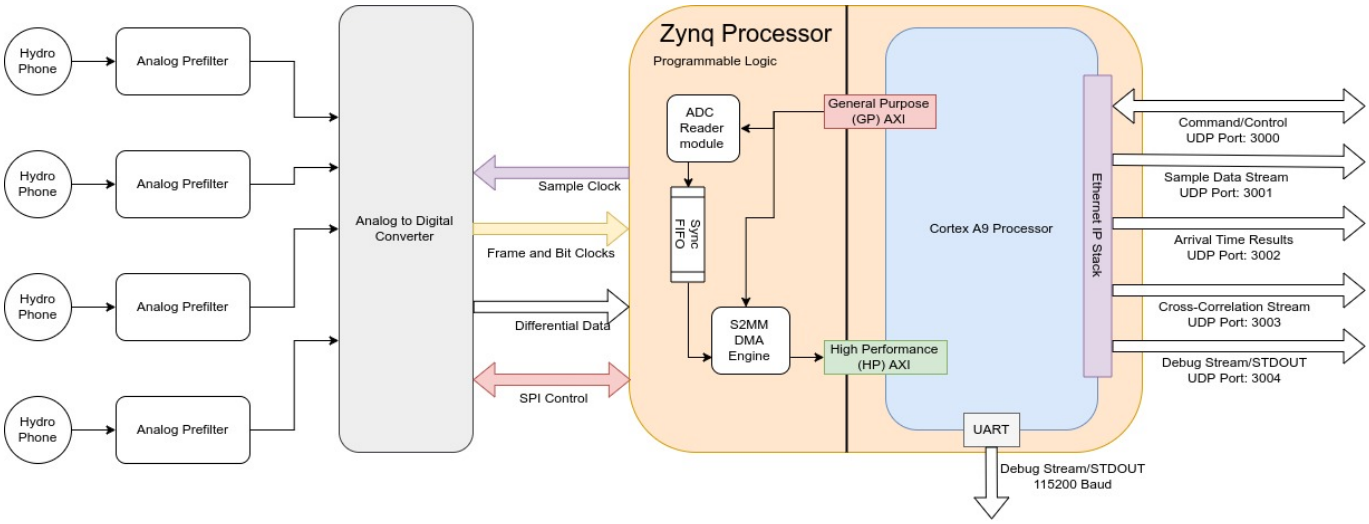
The cross correlation step is simple mathematics. Note that cross-correlation boils down to multiplying individual points of two signals together and accumulating, so if the signals are big it can take quite a bit of time. We will discuss how to handle this later.

The main difficulty exists in sampling the signals. It is important that samples are taken simultaneously on each of the hydrophones and that the signals are sampled fast enough to ensure that the precision in time of flight is accurate enough (for example, at a sampling frequency of 1MHz, the time of arrival can never be more precise than a signal microsecond). Therefore, the system must be capable of simultaneously sampling four channels at a minimum rate of 1MHz.

Hardware

Device	Part Number	Datasheet
Processor Carrier Board	MicroZed	Datasheet
Main Processor	XC7Z010	Datasheet
Hydrophones	AS-1	Datasheet
Analog-to-Digital Converter	LTC2171-14	Datasheet

System Design



The hydrophones have small piezo sensors that output very small voltages as sound hits them. To be able to read them, the hydrophone signals are filtered with a 1st order bandpass and then are gained by approximately 40-60dB. After the signals are amplified, they are then passed to the ADC for conversion. The ADC samples the hydrophone signals at 5MHz simultaneously and outputs the converted information over a custom, parallel interface. This interface is differential and utilizes DDR. In order to read the ADC values, the Zynq SoC (System on a Chip) is used. The Zynq has a dual core ARM processor embedded in FPGA fabric. A custom Verilog module was created for reading the parallel interface output by the ADC. The data is then sent over the AXI4-Stream protocol to allow it to be transferred into the processor's RAM through the HP-AXI interface (AXI is a common communication protocol for custom hardware modules written in Verilog). In order to accomodate the high data rate of the analog measurements, the AXI stream is connected to a DMA engine. At this point, the data has been successfully transferred into the computer and the cross correlation can be performed. The below sections delve into specifics of each of the different design areas.

Analog Design

The analog signal goes through a five-stage amplification and filtering process. First, the signal is sent through two gain stages, each of which apply an adjustable gain to the signal. After the signal is gained into a reasonable voltage range, it is passed through a low-pass filter and then a high-pass filter to result in a bandpass filter of the signal.

Once the signal has been filtered and gained, it is preprocessed for the ADC. In order to acquire more accurate measurements, the ADC requires signals to be differential. The analog signals are therefore put through a signle-ended to differential converter. Immediately before the signals enter the ADC, there is a low-pass anti-aliasing filtering. This filter guarantees Niquist criteria, which means that any frequencies detected in the sampled signal will not be aliases of higher frequencies.

FPGA Design

The FPGA has a number of parts to it, but its ultimate goal is to take samples sent from the ADC and transfer them into the processor's memory. Immediately, the differential signals are passed to the

FPGA IO buffers to convert them to single-ended. After the single-ended conversion, a custom Verilog block was written that takes in the ADC's clock and data signals to properly digitize the data. It then provides this data on an AXI-Stream interface. Because ADC reader block is driven off an external clock and the rest of the FPGA is clocked internally, care must be taken to cross the clock boundaries to prevent metastability. The Xilinx FIFO generator was used, as it generates an asynchronous read-write FIFO that can be used for synchronizing data with the internal FPGA clock.

After synchronization, the AXI stream is connected to a Stream-To-Multi-Master Direct Memory Access (S2MM DMA) engine. The DMA engine is a software-programmable peripheral that will asynchronously write the ADC samples into the processor's RAM.

There are a few other features implemented in the FPGA fabric, including the XSystemMonitor, which monitors the FPGA temperature. There is also a SPI module implemented that is used to program the ADC internal registers and verify its functionality. In Rev C of the HydroZynq, the manually adjustable potentiometers have been replaced with SPI-programmable potentiometers, which allows the analog gain of the signal to be controlled digitally.

Software Design

Getting Started

Code

Programming

Communicating