EMBEDDED - II PROJECT REPORT

We have applied a 2d approach to find out the angle of arrival of continuous audio when a valid event is detected using cross correlation.

List of Peripherals Used:

PORT E3 - AIN0 Mic 0

PORT E2 - AIN1 Mic 1

PORT E1 – AIN2 Mic 2

PORT D3 - AIN4 Mic 3

PORT F - RGB PWM

ADC0 SS1- Digital Comparator

ADC1 SS1- DMA Setup.

Design Approach:

ADC1 is set up to continuously transfer data from peripherals to memory using ping pong mode in DMA which stores the raw values in primary and alternate buffers. When a valid event is detected a timer is enabled for 200us which is storing data for (t/2) time after the event and the DMA is stopped. We consider valid amount of samples which is samples away and from the valid event index and separate out the values 4 microphone values into 4 different buffers. We then cross correlate with the mic1, mic2 and mic3 values with each other and find out the order of hits, then calculate the angle.

Process:

I have set up Sequential sampler 1 for both ADC0 and ADC1 to receive the raw digital values from the 4 microphones. This sampler contains a buffer length of 4 which takes in those values. The sampling rate is set to 1 million samples per second which ends up to 250k sampling for each microphone at continuous rate. I have used Adc0 for the digital comparator which will be later on used to detect a valid event because when using a digital comparator the raw values are transferred from FIFO to it and cannot be extracted out. So, I am using ADC1 for the DMA setup to transfer the raw data values from the peripheral to the memory directly.

DMA Setup:

We design a control structure of 1024 byte boundary, this creates a control structure where we allocate memory for primary and alternate control structures which are used for the ping pong transfer.

In each of the control structures there will be source end pointer, destination end pointer and control word (at an offset).

The DMA receives the data from the source end pointer which is ADC1 located at channel 25 with secondary channel assignment to the destination end pointer which are PRIMARY and ALTERNATE buffers.

The control word is set to destination size with an increase of 16bits, transfer size set to 255 and mode to ping pong with arbitration size of 2.

Usage of digital comparator:

The transfer is continuous until we detect a valid event which is when digital comparator 0 and 1 detect an interrupt in my case(reads the bits from raw interrupt status to determine the event). The threshold values are set based on the sensitivity of the microphones (Observed raw values) and set up the comparators for those microphones where we can only trigger when there is a valid noise. The trigger is detected on the high band.

After a valid event, We enable a timer for 200us because the distance between each microphone is 7cm gives 7cm/(343m/sec) and stop the DMA to take out the valid samples and separate them into 4 different microphone buffers and re enable the DMA.

Determining the order of hits:

We perform cross correlation using sliding window technique, This is performed between mic1 and mic2 which gives the time delay between x and y Txy, mic2 and mic3 gives the time delay between y and z Tyz and mic3 and mic1 gives the time delay between z and x Tzx. Then convert these into micro seconds {Txy,Tyz,Tzx} and find out the maximum value which might be our FIRST or LAST hit. We can find out out second hit from this, if our maximum value is Tzx then our second hit is Tyz. We then compare the remaining values which are Txy and Tyz.

if Txy < Tyz(second hit), then our first hit Txy and our last hit is Tzx.

if Txy>Tyz(second hit), then our first hit is Tzx and out last hit is Txy considering our second hit was be determined in both cases.

We then calculate the angle using,

theta = theta(first_hit) + k1 * sign * second_hit + k2(second_hit*second_hit);

here sign bit is dependent on the conditions mentioned above.

I have also implemented a color wheel map using PWM which changes its color depending on the angle of arrival.

Putty Commands:

The hold off Command is the time between one digital comparator interrupt to an other. This helps in calculating the angle of arrival and does not trigger any interrupt in between.

The tdoa command gives the time delays of arrival{Txy,Tyz,Tzx}.

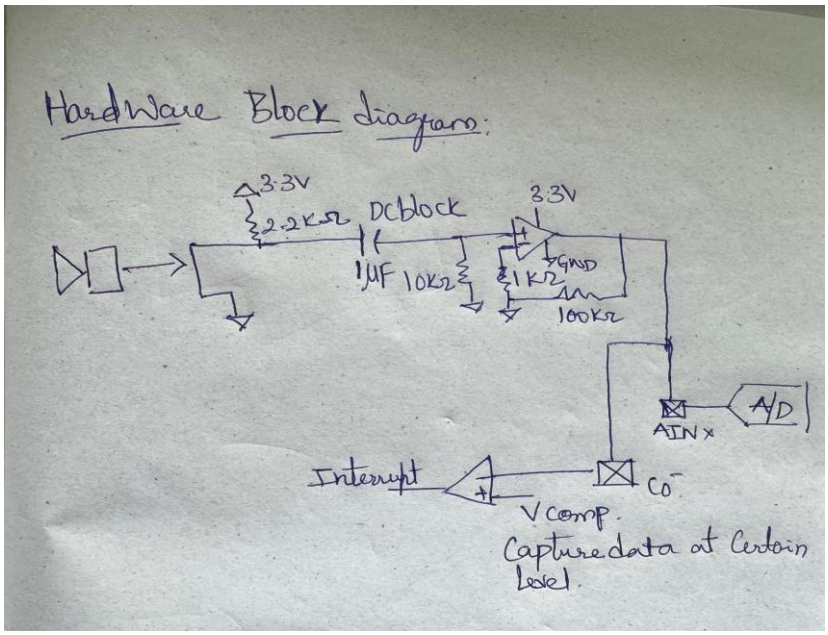
The AoA command gives the angle of arrival just once.

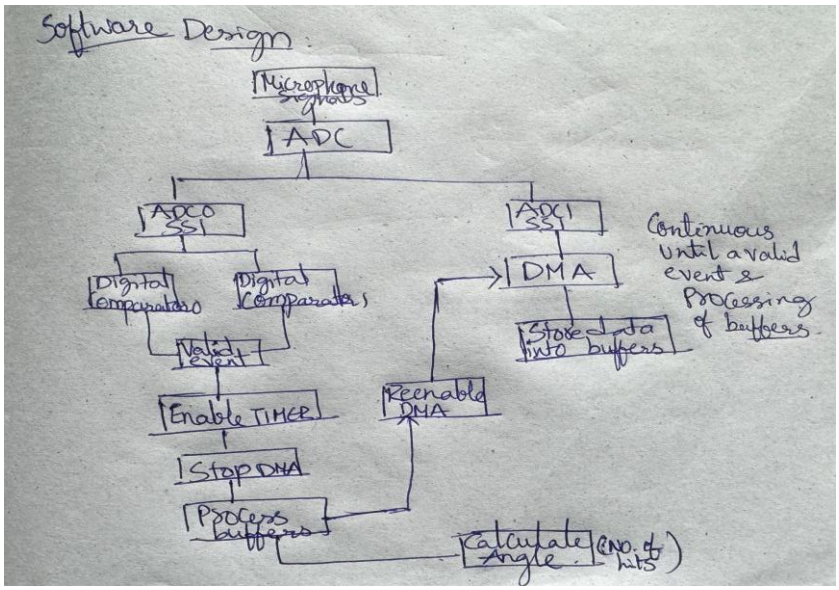
The AoA always command give the angle of arrival continuously.

Design Failings:

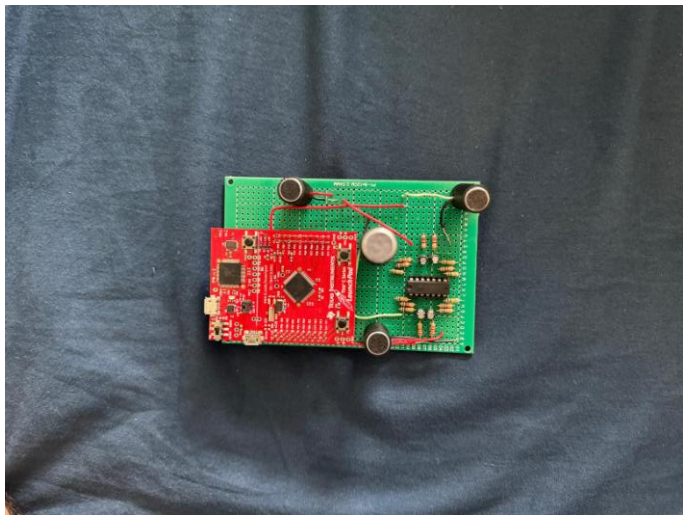
Detecting valid samples from the primary and alternate buffers. This can be improved using status of the buffer at a valid event and before stopping the DMA. This might have improved the accuracy of angle of arrival.

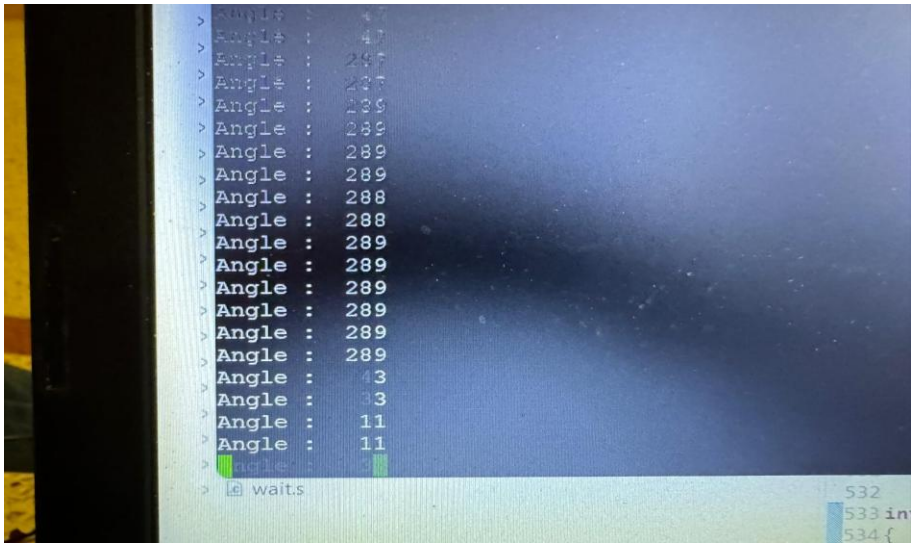
Hardware Block Diagram & Software Design





Hardware & Output:




Angle : 289
Angle : 289
Angle : 289
Angle : 288
Angle : 288
Angle : 289
Angle : 289
Angle : 289
Angle : 289
Angle : 289
Angle : 3
Angle : 3
Angle : 11
Angle : 11
wait.s
532
533 in
534 {