Sonar

Your task is to write a program that determines the distance to an object by measuring the sound propagation time. Therefore the FM-CW method shall be used. The program has to replay an audio signal with variable frequency ("frequency sweep") via sound speakers and has to record the audio signal which is reflected by an obstacle with a microphone. The cross-correlation between the transmitted and the received signal has to be calculated. The position of the maximum peak in the cross-correlation is equivalent to the runtime of the audio signal.

The harmonic oscillation with variable frequency has to be generated by using an IIR filter of 2^{nd} order, with two complex conjugates poles on the unit circle. Such a filter will oscillate. Vary the frequency from 1 kHz to 10 kHz within a period of 30ms. Use an up-and-down sweep (i.e. total length is 60 ms) as your sounding signal. Normalize the amplitude of the sweep signal (make it independent of the frequency) following the procedure described in the document "Amplitude of sine-waves generated by an oscillating IIR-filter".

The measuring range shall be from 1 m to 5 m distance. Since for the maximum measuring distance the signal will suffer a delay of about 10m / 340 m/s = 29 ms, be sure that you record the reflected signal for 30 ms longer than your sounding signal.

Use the EDMA for sending and recording the data. Start sending and recoding at the same time, but, as mentioned before, recording lasts longer.

After receiving the data, calculate the distance to the obstacle by finding the maximum of the cross-correlation between sent and recorded data. Do this in a SWI-function, which is posted by an appropriate hardware interrupt.

Calculation of the cross correlation can be "offline", i.e. while calculating the cross-correlation, you don't need to send in parallel a new sounding sequence .Start with a new sounding sequence as soon as measuring the distance has finished. Therefor you don't need to work with a double buffering scheme.

For calculating the cross-correlation, use two methods, which you select with a compile-time switch (#ifdef … #else … #end): computation in time domain and computation in frequency domain. Computation in time domain calculates the correlation sum as given by the well-known formula. Computation in frequency domain calculates the Fourier-transform of both the sended and the recorded signal, multiplies them point by point and then computes the inverse Fourier transform. You need to calculate the Fourier-transform of the sended signal only once, before starting the measurement. You have to take care to realize the aperiodic convolution, therefor use a FFT-length of 8192. It is convenient if you use for the forward FFT the decimation-in-time-algorithm and for the inverse FFT the decimation-in-frequency-algorithm.

Measure the time it takes to calculate the distance using time-domain and frequency domain method by setting a timer (use the Chip Support Library).

Simulate the algorithm in Matlab. Create a block diagram for your program, according to the required tasks and functions. Divide your program in several modules. Try to distribute the single tasks well among your group members. Define interfaces between the modules.
Consider testing concepts for your program to verify the functionality of the sub modules during the development stage. Think about verification for the complete functionality of the program.