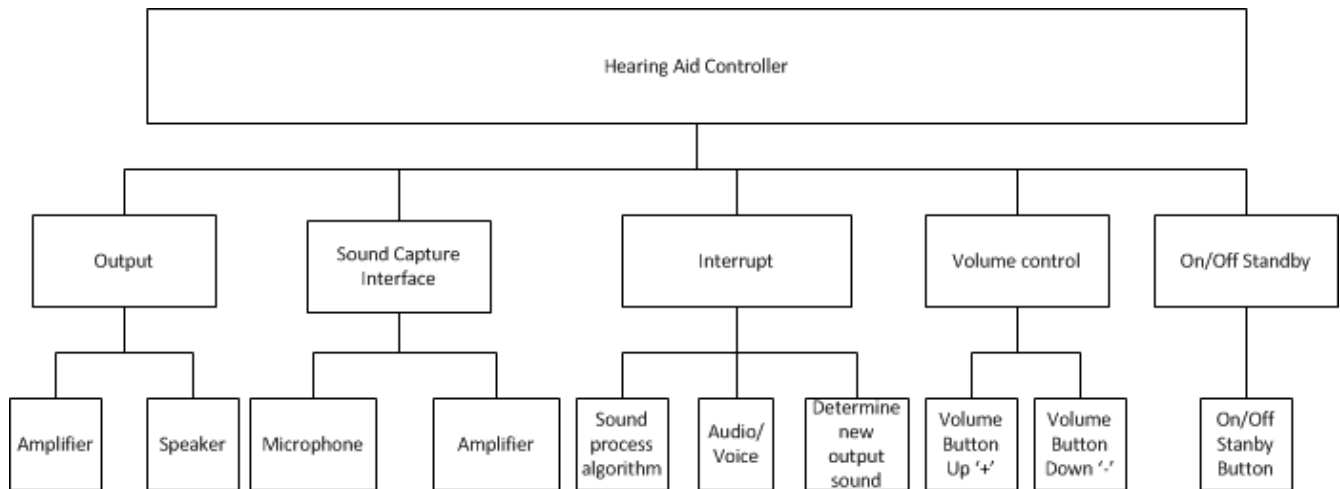
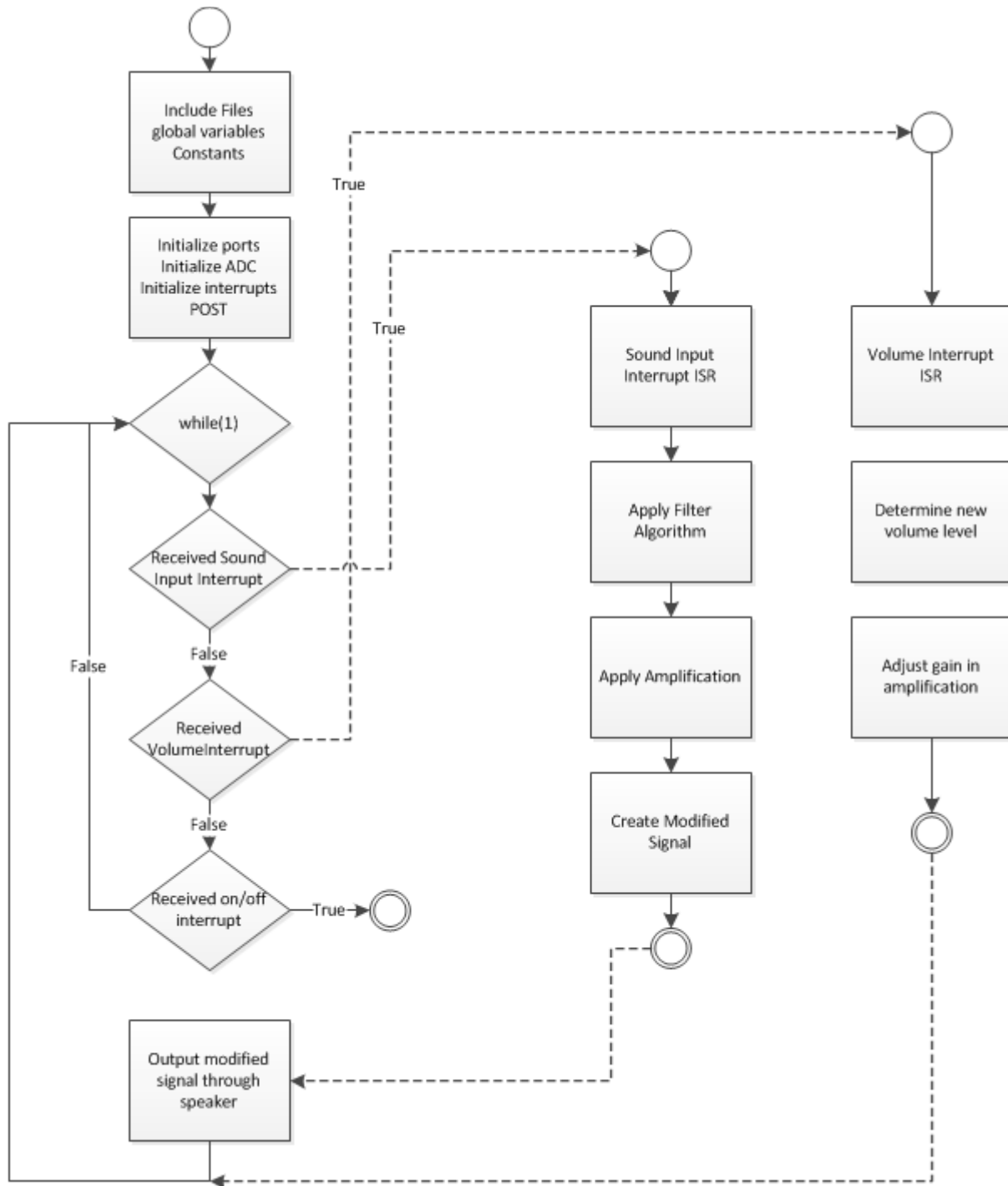


Software Structure



The diagram above shows the inputs and outputs that the hearing aid control software will have to initialize and handle. As well the interrupt which is generated when the sound is fed through the ADC. To conserve power methods have been researched to try and allow the DSP to run in a standby or power saving mode. Such methods include collecting samples and storing them, then when enough samples are collected then processing them. The idea is that most of the power consumption will be done while running the processing algorithms. So instead of processing every part of the signal we buffer samples and run the process on multiple segments we increase the power saving time and reduce the time we are actually running the processing algorithm.

UML Activity Diagram



This diagram shows the basic flow of the hearing aid controller code. The main idea with this is that we will setup the inputs, ADC and interrupts then jump into a main loop that will poll or wait for interrupts to occur. If an interrupt is thrown for the sound capture we will goto that ISR and process the data. As mentioned above to conserve power it would be beneficial to only perform the signal processing after collecting a significant number of samples. So the interrupt could save the samples as it collects them and then when enough samples have been collected then the program will process the data and output the processed signal to the user. The other interrupt that could be generated would be the volume

control buttons. When these are pressed then an interrupt could be generated which then would cause the program to have to adjust the volume level/ gain to the desired level.