

Lock-in Amplifier in a PSoC

Objectives

Implement a lock-in amplifier in a PSoC IC. This could be turned into a 'User Module' like the 'Low pass filter' to be used in the PSoC designer software.

Lock-in Amplifier basics

In the simplest form a lock-in amplifier provides a DC output directly proportional to the AC signal being measured. At the heart of a lock-in amplifier is a phase sensitive detector, which does the conversion. This is essentially a multiplier. It can be an analog multiplier, or a digital multiplier.

The converted dc signal is passed through a low pass filter to filter out the noise components giving better suppression of noise than is possible with a simple precision rectifier circuit. This makes it different from a filter and of use in a variety of applications such as noise measurement, spectrum analyzers, phase meter etc.

The amplifier needs reference signal of the same frequency and a constant phase relation as that of the signal of interest. Hence both are derived from the same source (usually). Thus the output of the amplifier will 'track' the changes in the frequency of the signal being measured as the reference is locked to it. This gives the amplifier its name.

The Q factor of a lock-in can be many 100's of times better than typical filters ($Q < 50$)

More details, simulations, articles on lock-in amplifiers can be found at www.lockin.de and www.signalrecovery.com

We will do a digital form of the lock-in amplifier taking into consideration the advantages offered by the PSoC device. This gives us a 'Lockin User Module' like the other user modules in the PSoC designer.

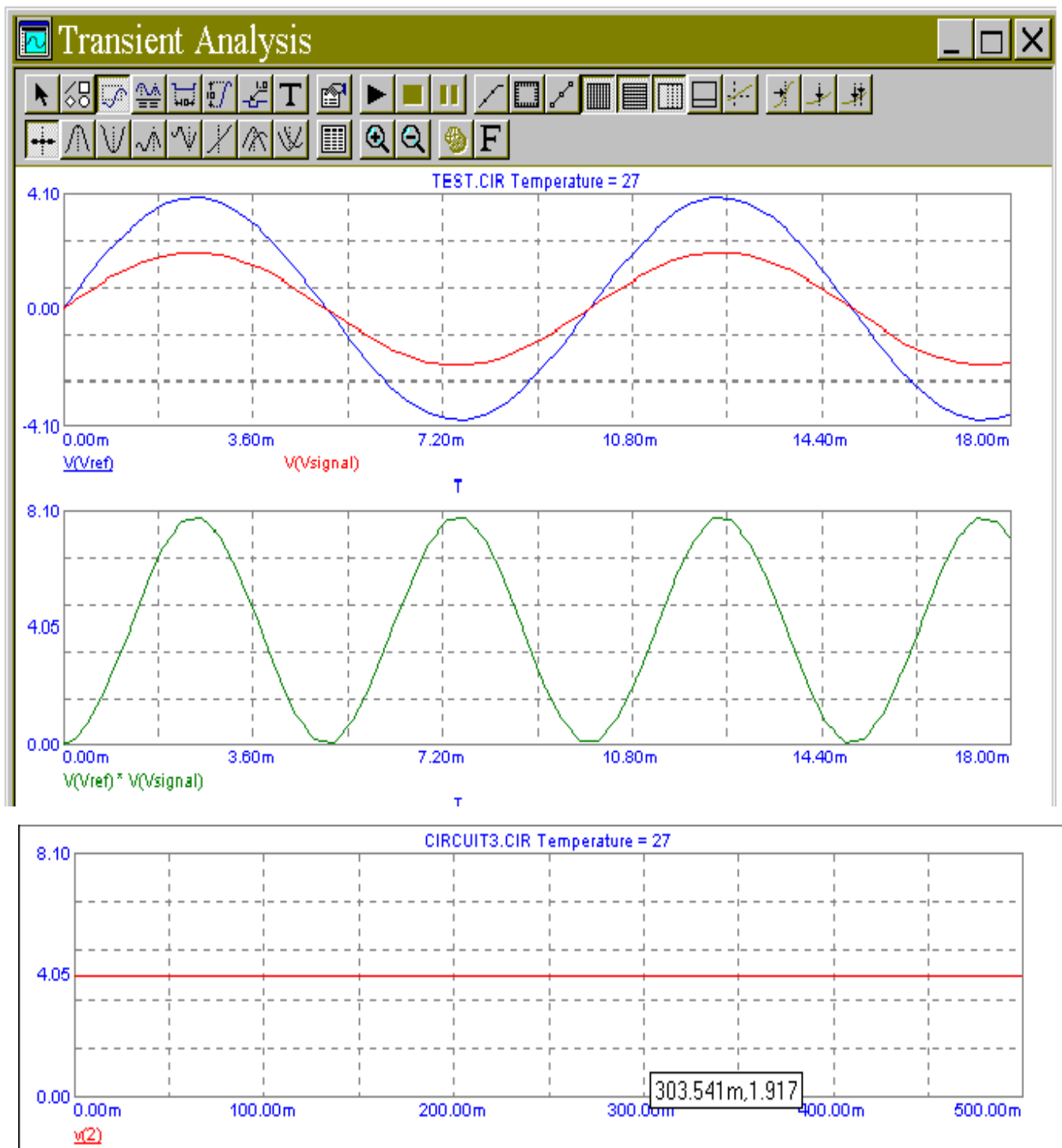
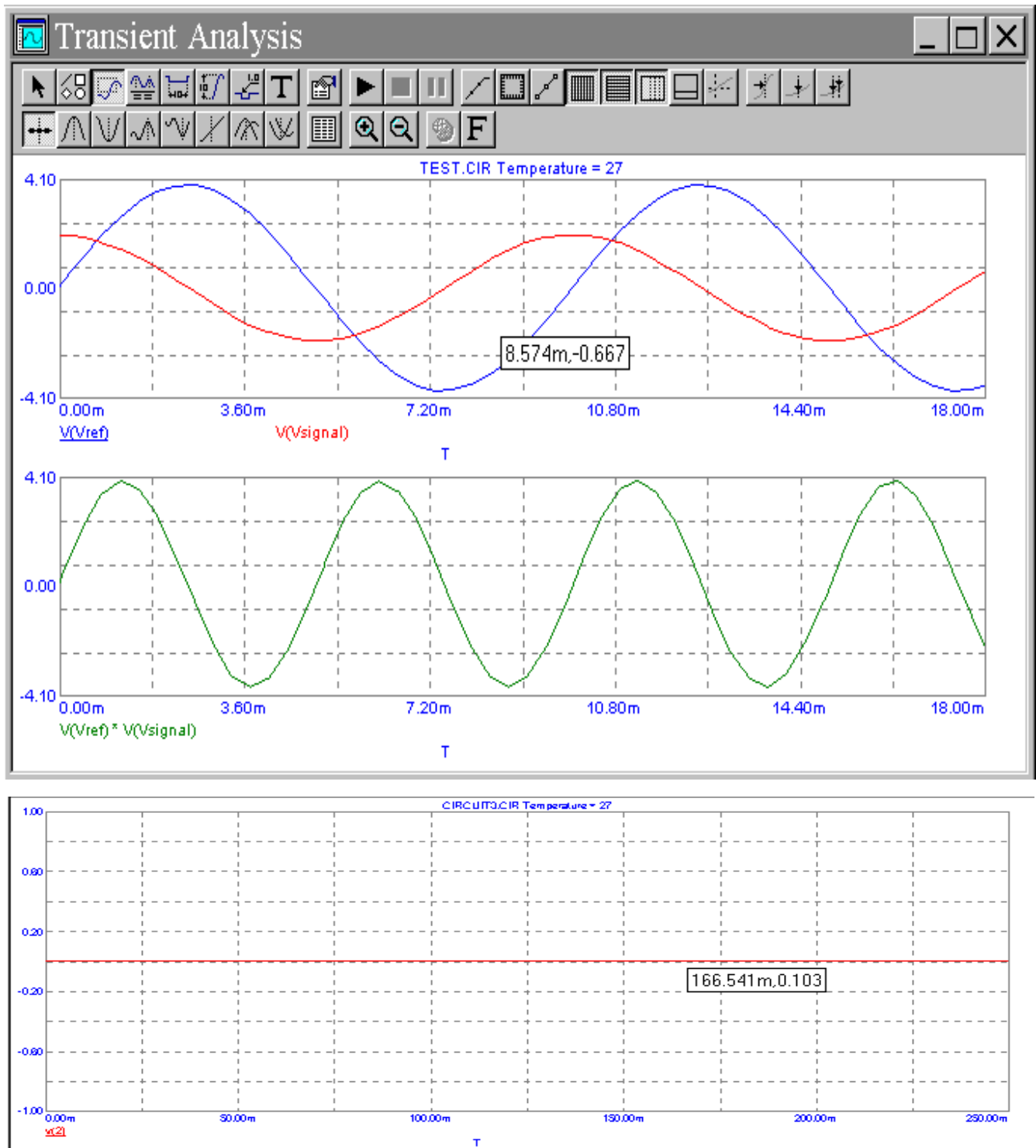


Figure 1

The simulation in figure 1 shows a reference signal multiplied by the input signal with 0-phase difference. This results in a signal at twice the reference frequency and with mean level above 0. Noise is ignored for now. This output when low pass filtered gives us a dc level proportional to the input signal.

Figure 2

Figure 2 shows the same two signals, but with 90 degrees difference. This yields a zero centered wave from the multiplier and consequently a zero volt output.



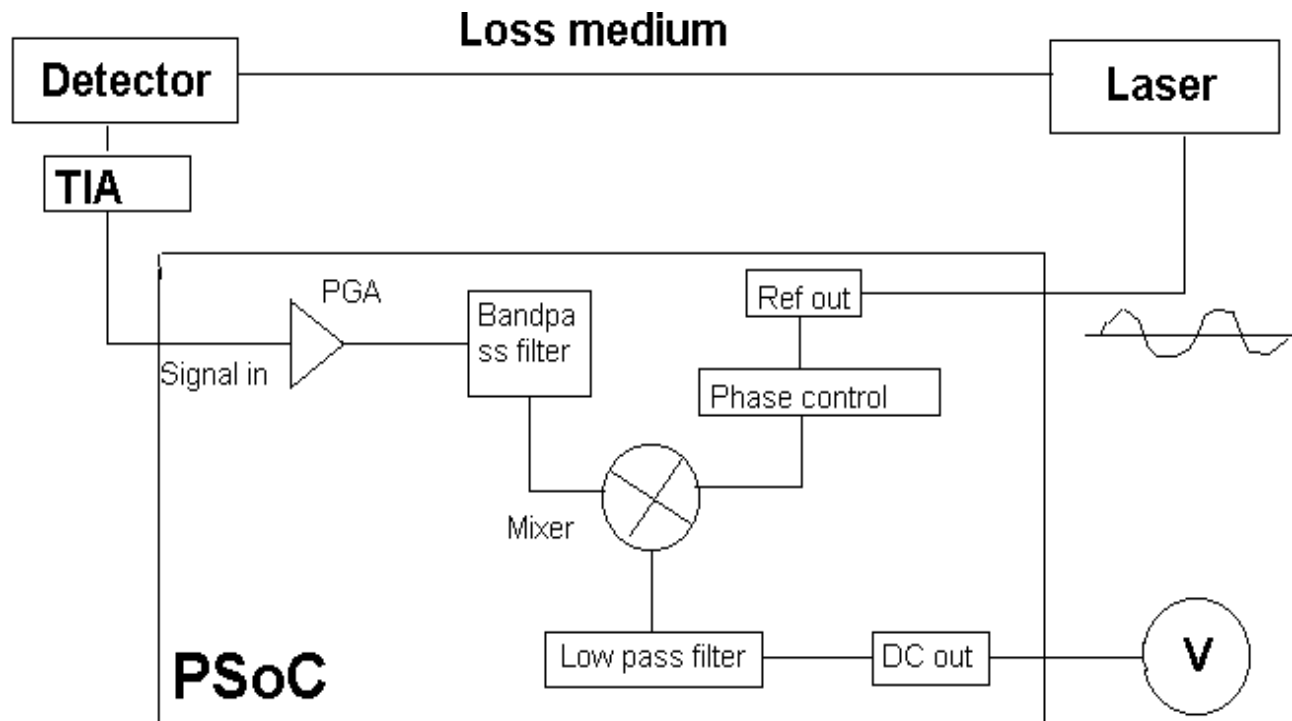
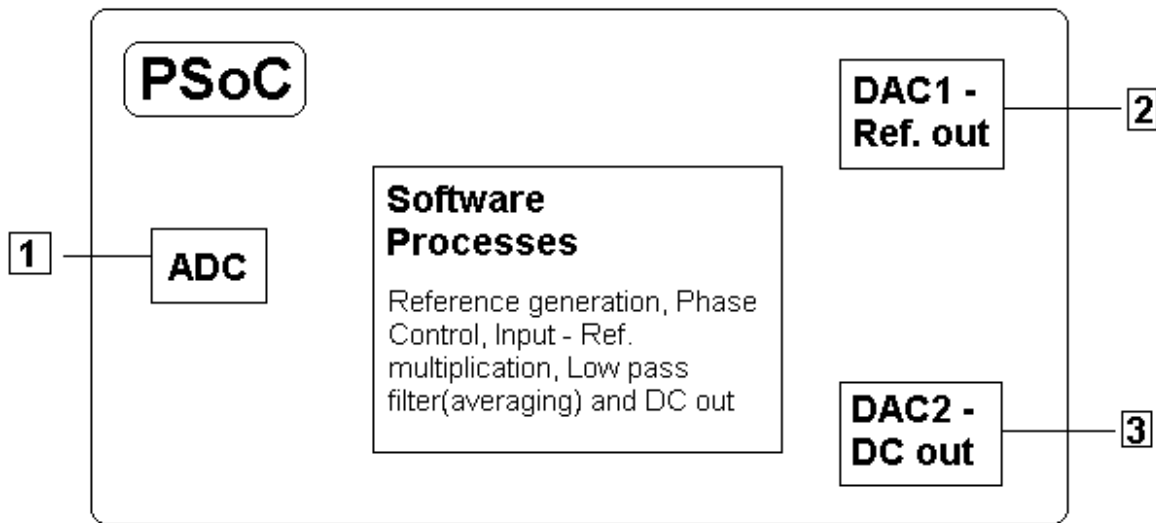


Figure 3

Figure 3 shows a complete system block diagram as to how a system built with PSoC in an optical measurement system may look like. It also shows which blocks could be inside a PSoC.

1. PGA – An auto ranging programmable amplifier stage to increase the dynamic range. This can be implemented using the PSoC PGA module
2. Bandpass filter - This can be implemented using the PSoC bp2 module
3. Ref. Out – Reference sine wave output. This can be implemented using PSoC DAC module
4. Phase Control – To compensate for phase difference between the Reference out and the Signal In. This is implemented in software
5. Mixer - This is implemented in software
6. Low pass filter – This could be implemented in PSoC LPF module, but then it requires a high resolution DAC and a filter. It is much simpler to do this using software.
7. DC out – Puts out the DC value proportional to the input signal. This can be implemented using a PSoC DAC module

Figure 4



Let us build a simple system that could be used as a Lockin amplifier 'User Module'. The DAC1 produces the reference sine wave output. The ADC is the signal input and DAC2 produces the scaled DC output proportional to the input signal.

The whole thing can be implemented in the PSoC PUP demo board without many external components. Hence the schematic diagram is not required. To test the system output shown as 2 in figure 4 may be connected to the input 1. To test out the 'phase control' algorithm, 2 may be connected to 1 through a low pass filter for phase shift.

Operation

As shown in figure 5 below, the reference signal is generated using the counter interrupt. The sine wave table value pointed by the 'index' variable is put out of DAC1. The system then reads the ADC, performs the multiplication of the input value and the sine table value pointed by 'index-phase' to correct for delays.

The 'lock_amplifier' routine tries to find the correct phase to lock onto. The interrupt is generating a sine wave and also averaging the input and reference multiplication all the while. The routine scans through the entire sine table array to find the 'index' at which the 'averager' value is at a maximum.

The main program loop takes care of all initializations, enable interrupt and within the interrupt scaling of averager value occurs to fit within the 8-bit DAC2 output to produce the DC output.

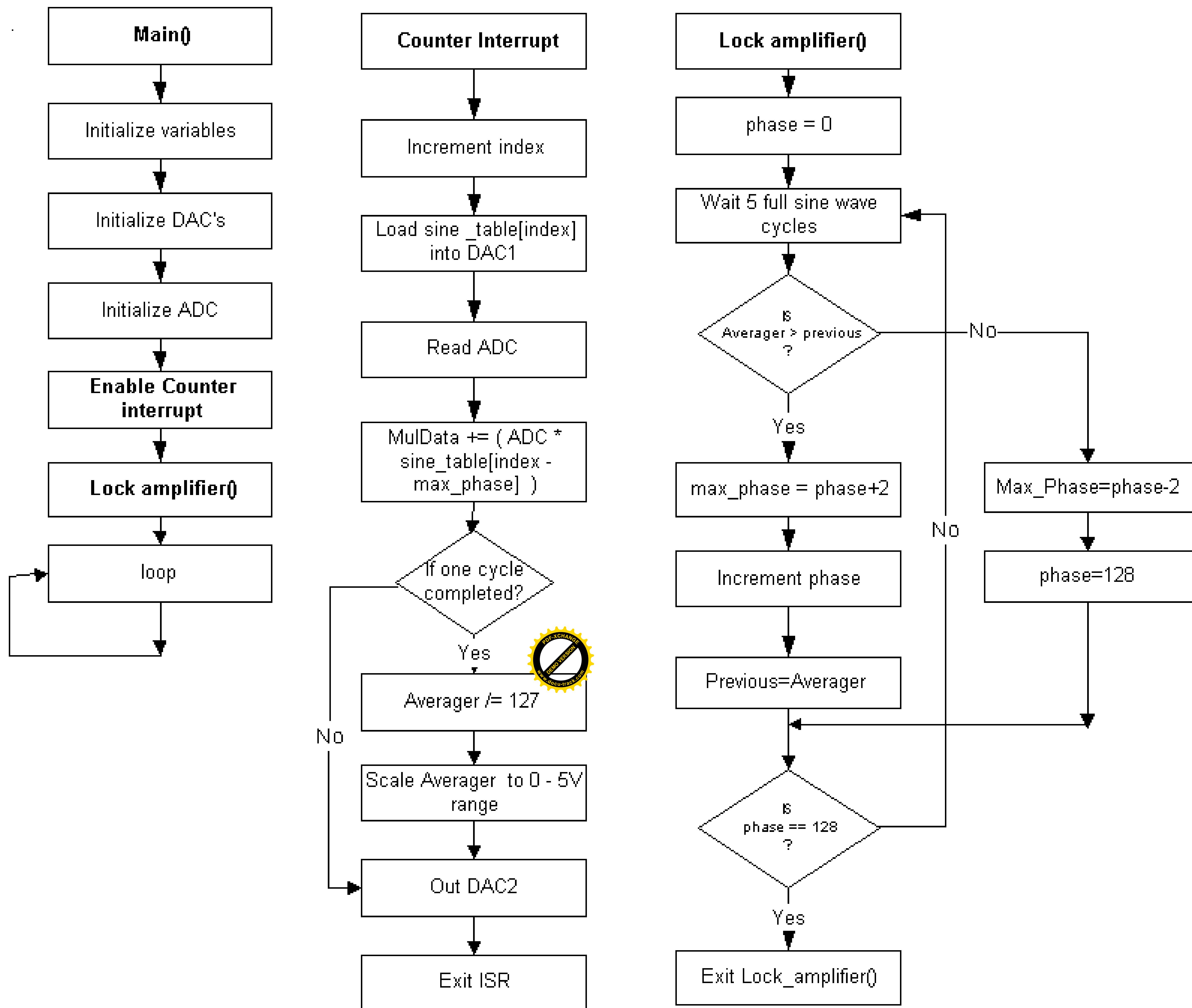


Figure 5

TESTING

In the absense of an emulator, the rs232 port was used for watching debug variables.

Another problem encountered was that the compiler produced too much code the DAC interrupts got delayed. So instead of writing a separate function for locking function, the code wrote within a while loop in the main body.

The DAC reference output on pin 3 was connected to pin 21 the ADC input. So we are putting in the same reference as the input signal. We should have no problem locking to it. But the ADC data appeared with a phase shift of 180 with respect to DAC.

So to investigate this a separete piece of code was developed. We removed the lockin function and a fixed reference phase was used. A simulation was done taking into account the voltage levels of DAC and ADC. The phase of the reference was swept from 0 to 72degrees. As seen from the simulation below, the Lockin DC output DAC should produce voltage from 3.3V down to about 2.8 volts.

The phase index was set ,the program built and downloaded into the PSoC device. The lockin output voltage was read. This was repeated for fixed phase relations from 0 to 72degrees. It was seen that the lockin DC output changed from 2.9v to 3.6v. We had accounted for a maximum of only 180 degrees phase difference in the locking function. So no locking occurs at this time.

Further test revealed that results as per the simulation is obtained if the phase relation is plotted from 180 to 252 degrees. This proved that we are encountering a 180 degree phase shift somewhere. We are unable to account for this phase shift at this time.

CONCLUSION

1. Unless assembly is used we may be able to do a reference frequency in the 10 to 20 Hz range. Compiler needs improvement.
2. A practical Lockin will require 2 DAC's(one for coarse one for fine control) for 65535 level DC output.
3. ADC resolution needs to be better or range switching need to be used for higher dynamic range.
4. Needs code simulation and analog block simulation tools.

