Re: Questions on interfacing to current sense transformer

Source: http://sci.tech−archive.net/Archive/sci.electronics.design/2009−06/msg04321.html

- *From*: MooseFET <kensmith@xxxxxxxxx>
- *Date*: Sat, 27 Jun 2009 09:11:15 −0700 (PDT)

On Jun 26, 7:42 pm, John Larkin
<jjlar...@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote:
[....]

> I'm thinking of building a homemade power meter. I'm planning on using an Arduino with ethernet "shield" to allow me to read the power measurements over the internet.

[....]

> Your software should sample the voltage and current as close to simultaneously as possible... within 10 usec is OK. Sample fairly often, but it's not really important. I've done good meters that sampled as slowly as 27 times a second. Nyquist is not an issue here.

You can improve things a little if you sample V then A or A then V
depending on how you see the phase shifts stack up in your design.
The two errors can be made to cancel slightly.

> Process the samples as follows:
>
> Average blocks of, say, 1024 samples, and subtract the average from each sample. That removes the 2.5 volt offsets.

Make your sample rate such that "1024" samples is an integer number of
cycles of the 60Hz.

Also try to make it so that this integer is odd. This means that in
different cycles you samples land at different points. This removes
some of the errors.

> Square and average the volt and amp samples independently (block average or lowpass filter)

Assuming you don't want to keep lots of values around:

Since you are already computing the average, subtract the previously computed average from each value before doing the square but also remember the average of the numbers you are squaring. The previous average may be off a little but it is close enough that the some of squares is kept from growing way too many bits.

The corrected mean square is the mean square minus the square of the mean. The mean in this case is the mean of what you actually squared.

> and then square root.

Doing square roots can be a bit tricky and slow in smallish processors. You have a few things working for you. If you are doing:

while not good enough
Y = 0.5 * (Y + X/Y)

you can start off with the Y value from last time you did the sqrt. This is likely to be close to the right answer so you loop less.

There are methods that compute 1/sqrt(X) that don't divide inside the loop.

There are methods that work from the observation that the sum of the first N odd numbers is always N^2. This in practice is faster than methods involving multiplies or divides on machines that don't natively do those operations.

> That gives you true
> RMS volts and amps. Multiply them to get KVAs.
>
> Multiply and average (block or lowpass filter) the product of the voltage and current sample pairs. The result is true power.

Since your voltage is nearly a sinewave, you can make a 90 degree phase shifter in software so that you can compute and estimate of the imaginary part of the power. This number can be handy if you want to display the phase angle.

> You can integrate power to KWH. You can also compute power factor as
> true power over KVAs, but you lose the lead/lag sign.

If you have the imaginary part, you have the sign.

arctan is a very nasty function to compute when you are moderately far
from zero. A table look up may be the best for this step.

> Something like that.
>
> Add some random noise to the analog current signal, a few LSBs maybe,
> to smear out the ADC quantization error and extend the low−end
> resolution.
>
> John

.