Kursus 1.

1. Aquire an Arduino shield and solder up an 8 bit R-2R circuit allowing (not PWM) DAC from Arduino Uno. It is suitable to map the 4 bits of lowest significance to port D4-D7 and the 4 bits of highest significance to port B0-B3.
2. Apply a function generator, an oscilloscope with spectrum analyzer (Fourier transform)

- Let the function generator set a sinusoidal voltage at 50Hz, in the voltage range [0,3]V
- Sample the sinusoidal voltage with an analog Arduino input at a sampling frequency of 200Hz
- Send the sampled value to your R-2R DAC and measure with an oscilloscope
- Let the oscilloscope create a spectrum and compare to results in the litterature note.
- Redo the above for sampling frequencies 100 and 50 Hz

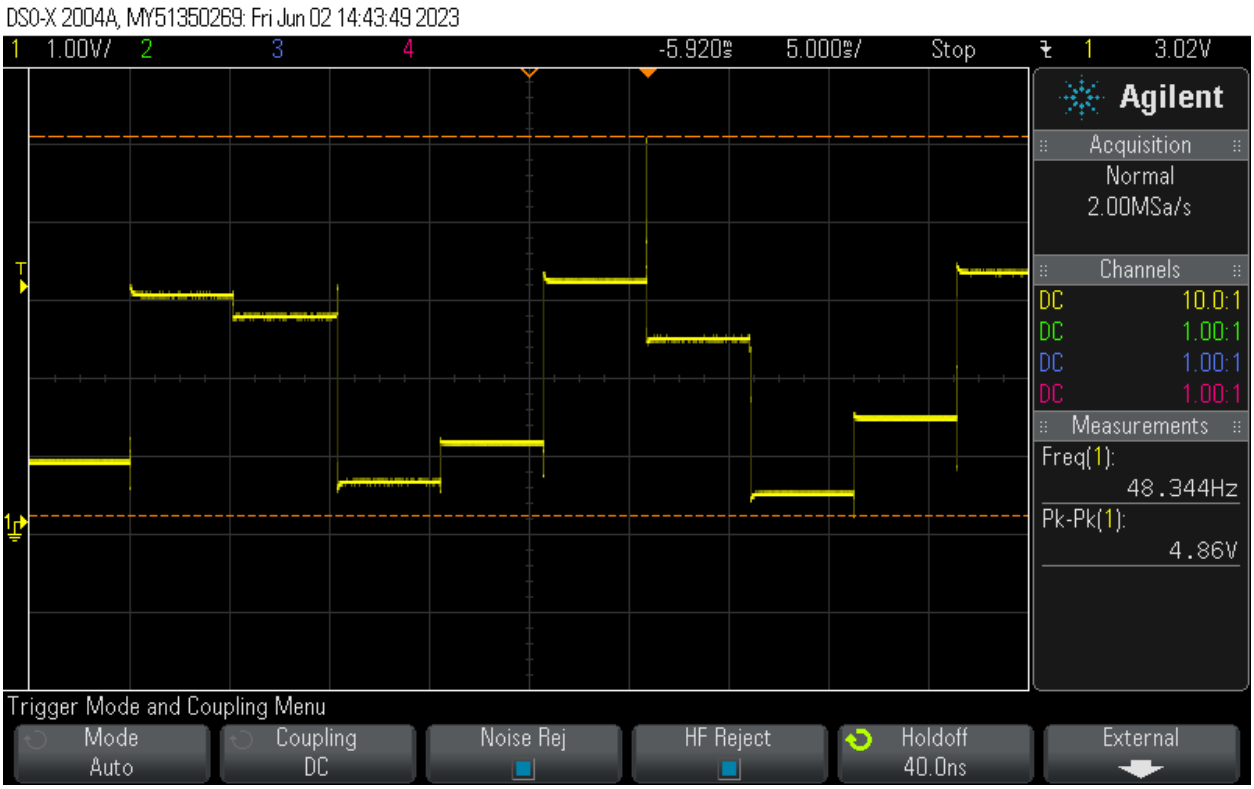
1. Output a constant duty cycle of 50% on the Arduino and observe the spectrum on the oscilloscope. Compare to literature
2. Output a constant duty cycle of 2% on the Arduino and observe the spectrum on the oscilloscope. Compare to literature

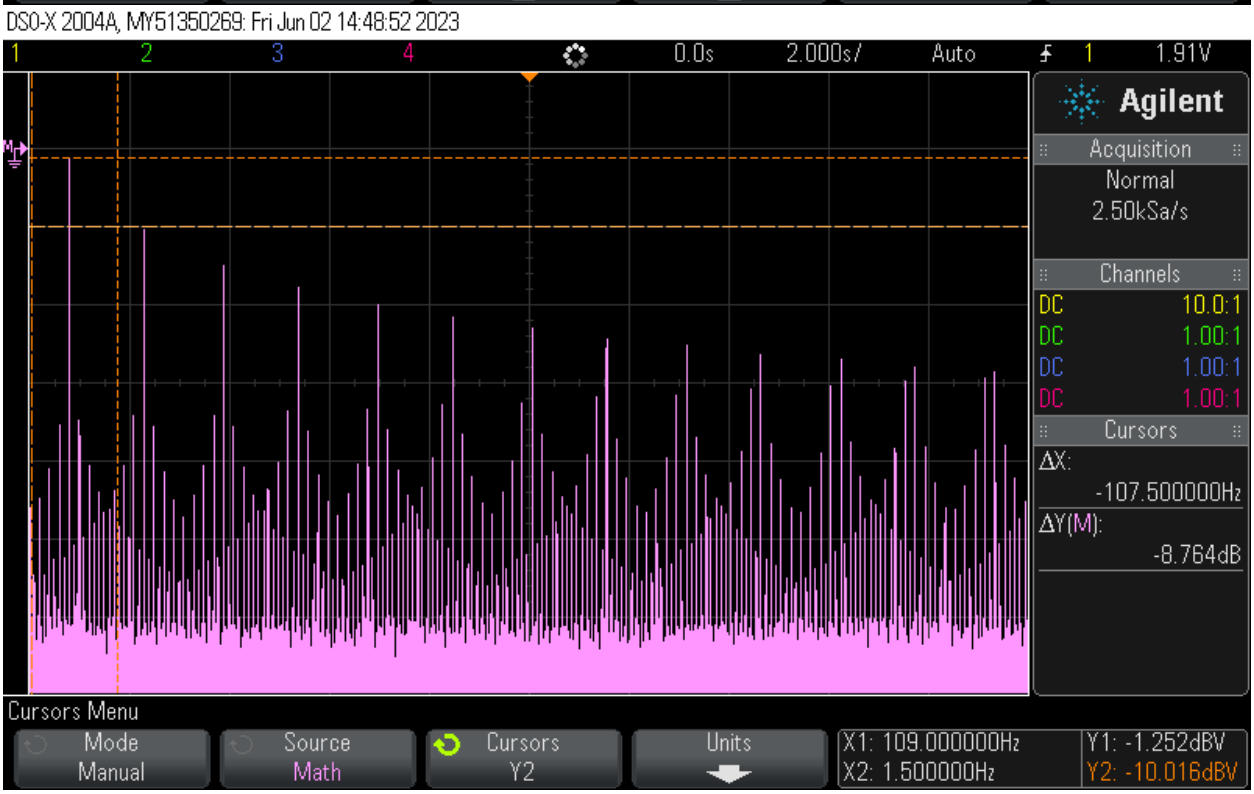
En 8 bit R2R DAC er brugt sammen med en Arduino uno. Et keysight DSOX2004 oscilloscope og en Keysight 33120 Function generator er blevet brugt.

Vi har observeret at når sampling frekvensen nærmer sig det man sampler i frekvens, bliver outputtet meget distortet, og FFT billedet er i store træk hoved frekvensen sammen med rigtigt mange ulige harmoniske, som forventet af en hovedsageligt firkant kurve. Når vi sampler med samme frekvensen som frekvensen af det vi gerne vil se, sår opstår der tydelig aliasing. Her er FFT billedet dog meget pænere og der er hele -40 dB til nærmeste harmoniske frekvens.

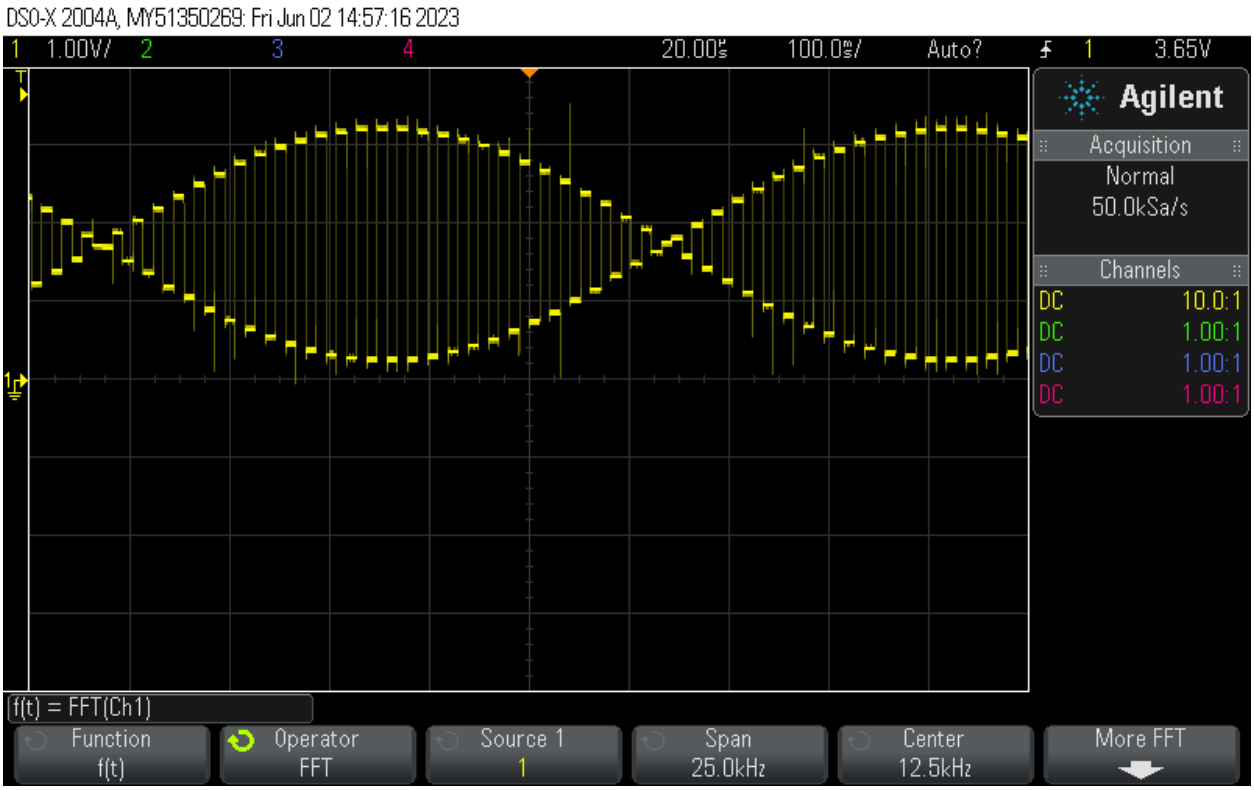
En 50% Duty cycle firkant består som forventet at uendeligt mange ulige harmoniske. Dette ser anderledes ud for en 2% duty cycle firkant.

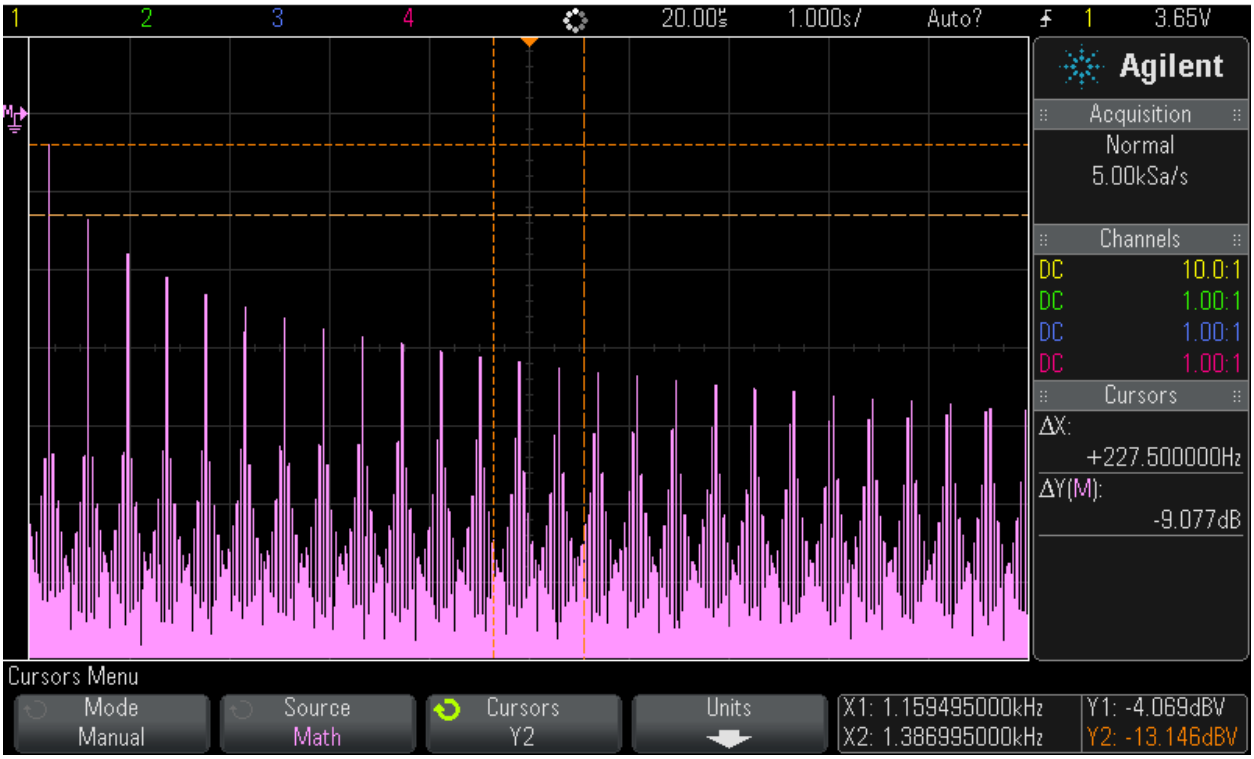
Følgende er malt for 200 Hz sample, sinus:



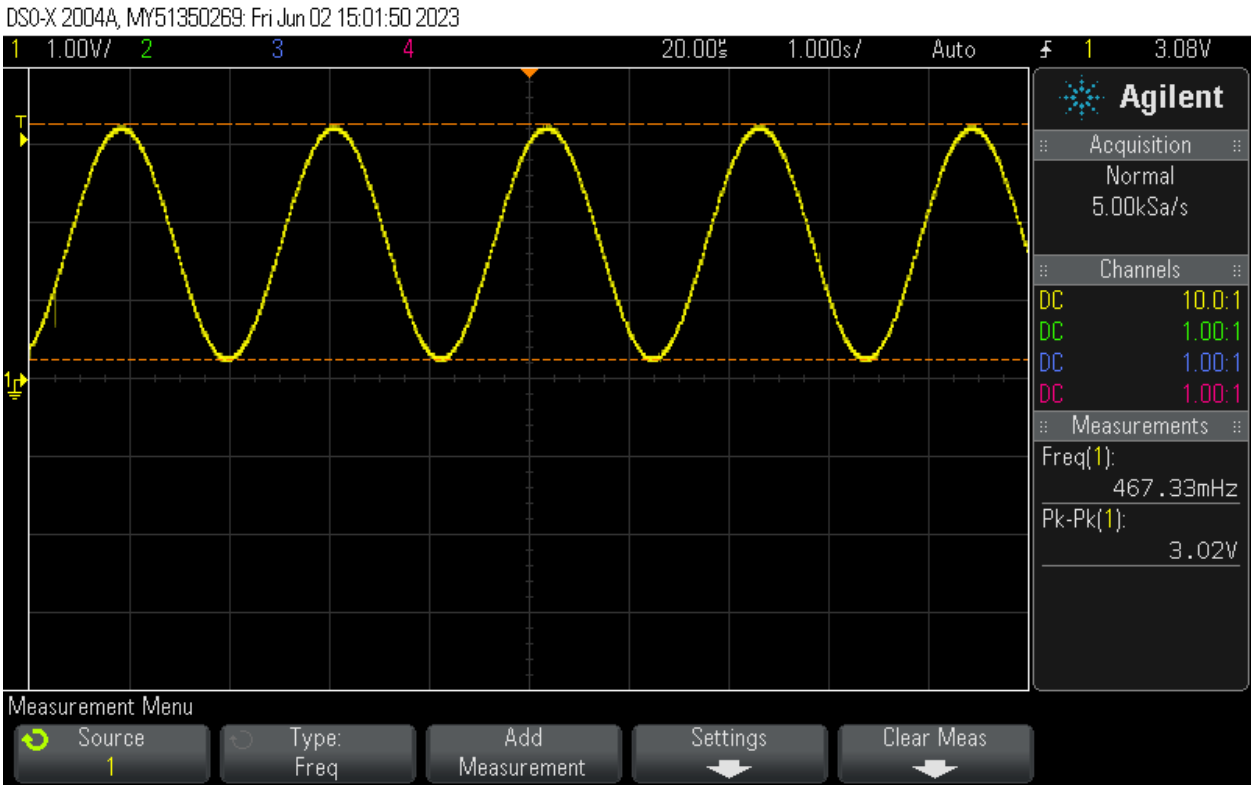


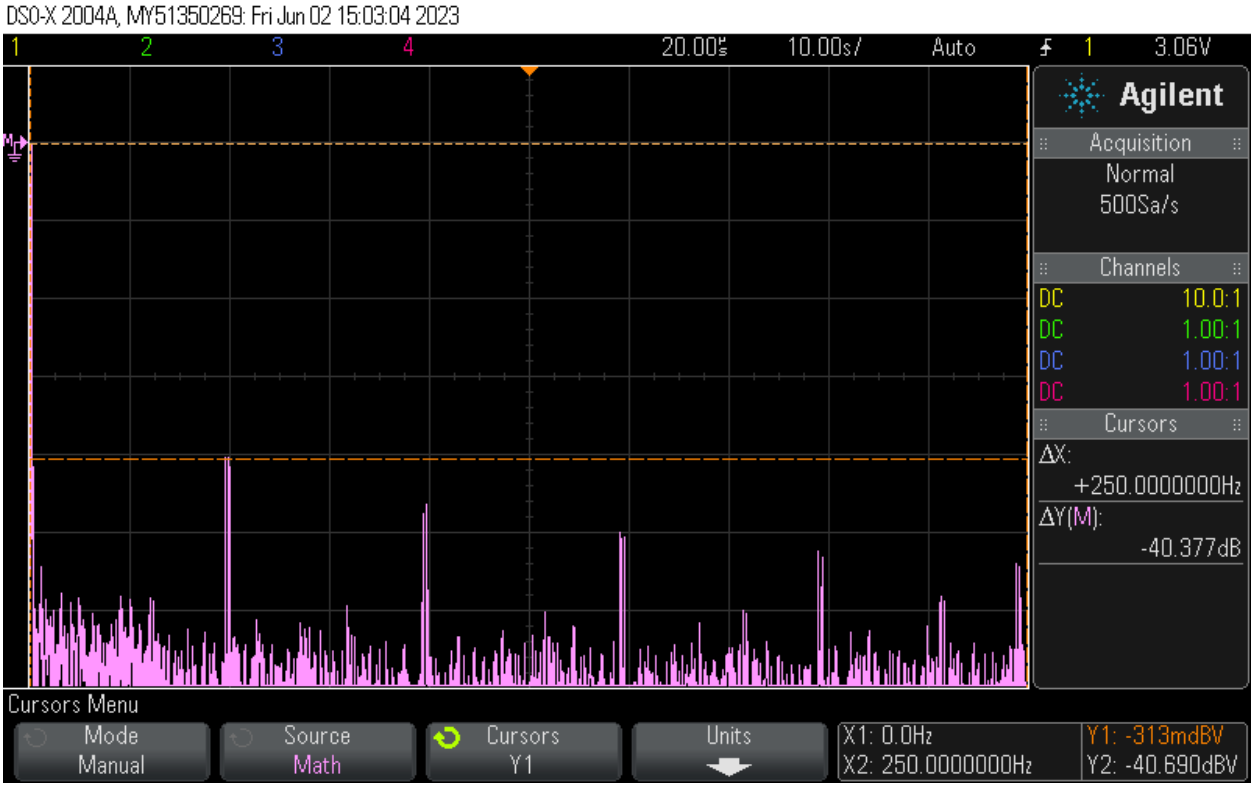
100 Hz sample sinus:



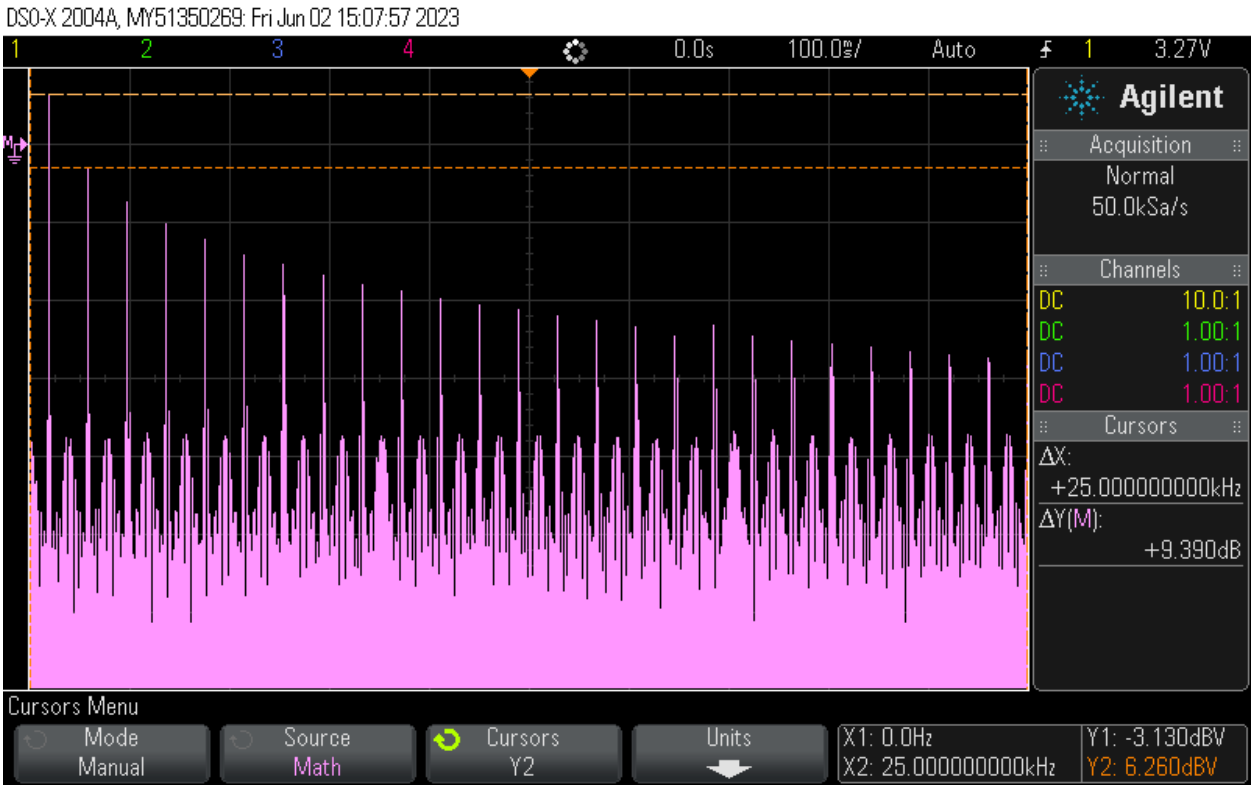


50 Hz sample sinus (Tydelig aliasing):





6 KHz sample 50 dcy square:



6 KHz sample 2 DCY square: