Python Example: Building an Arb with 16-bit steps (SDG2000X/SDG6000X)

March 07, 2018

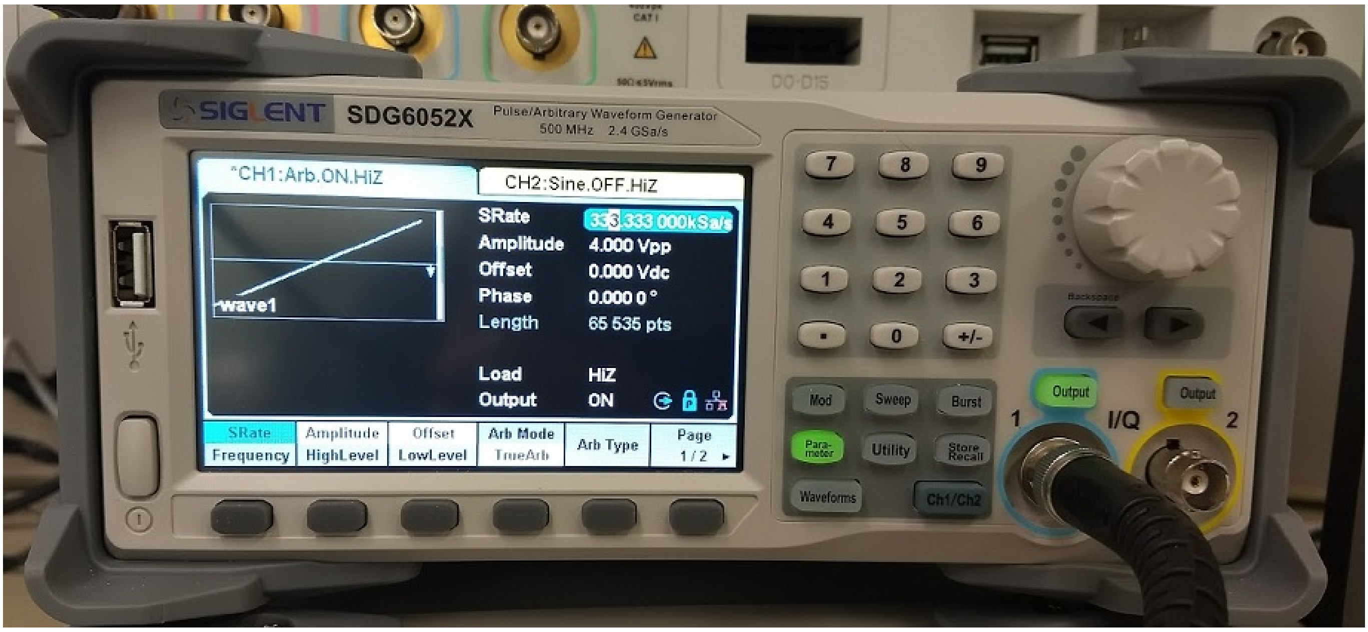
The SIGLENT SDG2000X and SDG6000X feature 16-bit voltage step resolution. This provides 65,535 discrete voltage steps that can cover the entire output range (20 Vpp into a High Z load) which can effectively be used to test A/D converters and other measurement systems by sourcing waveforms with very small changes.

In this example, we use Python 2.7 and PyVISA 1.8 to create a ramp waveform that is comprised of steps of the Least Significant Bit (LSB) from point 0 to 65535 on Channel 1.

We also implement the TrueArb function that allows you to specify the sample rate and also ensures that each sample is sourced.

NOTE: You will need to change the instrument ID to match your specific instrument. We also recommend setting the amplitude and other instrument parameters prior to enaling the output of the instrument.

Here is pic of the instrument after loading the waveform:



^CH1:Arb.ON.HiZ

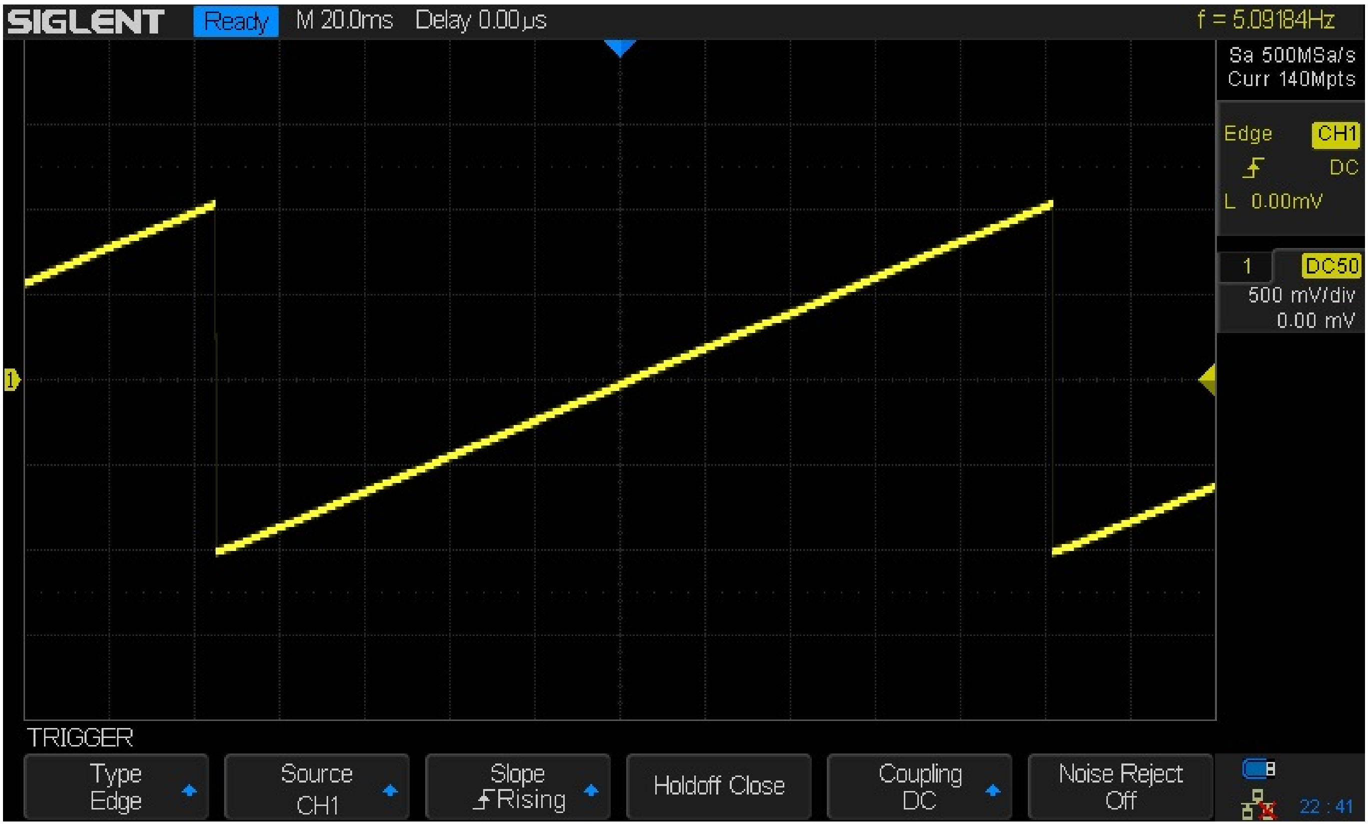
CH2:Sine.OFF.HiZ

SRate 333.333 000kSa/s
Amplitude 4.000 Vpp
Offset 0.000 Vdc
Phase 0.000 0 °
Length 65 535 pts
Load HiZ
Output ON

Arb Mode TrueArb

Arb Type

Here is a scope shot of the output:


SIGLENT Ready M 20.0ms Delay 0.00µs f = 5.09184Hz

Sa 500MSa/s
Curr 140Mpts

Edge CH1
DC
L 0.00mV

1 DC50
500 mV/div
0.00 mV

TRIGGER

Here is a link to a Zipped version of the .PY file: SiglentSDG16BBitSteps

Here is the text of the program:

```
##

#!/usr/bin/env python2.7
# -*- coding: utf-8 -*-
import visa #Uses PyVISA 1.8 and NI-VISA runtime Engine 15.5
import time
import binascii

#USB resource of Device
rm = visa.ResourceManager()
device = rm.open_resource(‘USB0::0xF4EC::0x1101::SDG6XBAQ1R0071::INSTR’)  #CHANGE TO MATCH YOUR INSTRUMENT ID

#Little endian, 16-bit 2’s complement
# create a waveform

wave_points = []
for pt in range(0x8000, 0xffff, 1):
wave_points.append(pt)
wave_points.append(0xffff)
for pt in range(0x0000, 0x7fff, 1):
wave_points.append(pt)
```

```
def create_wave_file():
#create a file
f = open(“wave1.bin”, “wb”)
for a in wave_points:
b = hex(a)
#print ‘wave_points: ‘,a,b
b = b[2:]
len_b = len(b)
if (0 == len_b):
b = ‘0000’
elif (1 == len_b):
b = ‘000’ + b
elif (2 == len_b):
b = ’00’ + b
elif (3 == len_b):
b = ‘0’ + b
b = b[2:4] + b[:2] #change big-endian to little-endian
c = binascii.a2b_hex(b) #Hexadecimal integer to ASCii encoded string
f.write(c)
f.close()

def send_wave_data(dev):
#send wave1.bin to the device
f = open(“wave1.bin”, “rb”) #wave1.bin is the waveform to be sent
data = f.read()
print (“write bytes:”,len(data))
dev.write_raw(“C1:WVDT
WVNM,wave1,FREQ,2000.0,TYPE,8,AMPL,4.0,OFST,0.0,PHASE,0.0,WAVEDATA,%s” % (data))
#”X” series (SDG1000X/SDG2000X/SDG6000X/X-E)
dev.write(“C1:ARWV NAME,wave1”)
f.close()

if __name__ == ‘__main__’:

create_wave_file()
send_wave_data(device)
device.write(“C1:SRATE MODE,TARB,VALUE,333333,INTER,LINE”) #Use TrueArb and fixed sample rate to
play every point
```

###

North American Headquarters

SIGLENT Technologies NA
6557 Cochran Rd Solon, Ohio 44139
Tel: 440-398-5800
Toll Free:877-515-5551
Fax: 440-399-1211
info@siglent.com
www.siglentamerica.com/

European Sales Offices

SIGLENT TECHNOLOGIES GERMANY GmbH
Liebigstrasse 2-20, Gebaeude 14,
22113 Hamburg Germany
Tel: +49(0)40-819-95946
Fax: +49(0)40-819-95947
info-eu@siglent.com
www.siglenteu.com

Asian Headquarters

SIGLENT TECHNOLOGIES CO., LTD.
Blog No.4 & No.5, Antongda Industrial Zone,
3rd Liuxian Road, Bao’an District,
Shenzhen, 518101, China.
Tel:+ 86 755 3661 5186
Fax:+ 86 755 3359 1582
sales@siglent.com
www.siglent.com/ens