

```
In [1]: import os
import numpy as np
import math
import pandas as pd
import struct
```

```
In [2]: mcs_file_path = "./mcs_data/p200ns200.005"
```

```
In [3]: byte_array=[]
with open(mcs_file_path, "rb") as f:
    byte = f.read(1)
    byte_array.append(byte)
    while byte != b"":
        # Do stuff with byte.
        byte = f.read(1)
        byte_array.append(byte)
```

```
In [4]: ba = byte_array
```

```
In [5]: def byte_array_to_string(byte_slice):
    return "".join(list(map(lambda x: x.decode(), byte_slice)))
```

```
In [6]: def hex_array_to_int(byte_slice):
    byte_slice.reverse()
    return int("".join(list(map(lambda x: x.hex(), byte_slice))), 16)
```

```
In [7]: def create_data(byte_slice):
    int_array = np.empty(0)
    while(len(byte_slice)!=0):
        int_array = np.append(int_array,hex_array_to_int([byte_slice.pop(0),byte_slice]))
    return int_array
```

```
In [24]:
```

```
Out[24]: 65532
```

```
In [34]:
```

```
Out[34]: -4
```

```
In [15]: dwell_units_dict = {0:'µs', 1:'ms', 2:'sec', 3:'ns'}
dwell_units_multiplier = {0:-6,1:-3,2:1,3:-9}
dwell_time = struct.unpack('!f', bytes.fromhex("".join(list(map(lambda x: x.hex(), list(ba[4:])))))
dwell_units = dwell_units_dict[int(ba[4].hex(),16)]
dwell_time_in_sec = dwell_time*(10**(dwell_units_multiplier[int(ba[4].hex(),16))])
```

```
In [43]: photon_counts = create_data(ba[256:1256])
time = np.arange(0, len(photon_counts))*dwell_time_in_sec
distance = time*(150000000) + (time[1]*150000000)
```

```
In [44]: df = pd.DataFrame([time,distance,photon_counts],index=['time','distance','photon_count'])
df.to_csv((mcs_file_path.split("/")[-1])+'.csv')
```

In []: