

# 實驗、射頻載波聚合開關模組實驗

## 一. 實驗目的

載波聚合(Carrier aggregation)技術是在 3GPP R10 版本中提出，該技術除了可以有效的增加使用者使用的頻寬外，更可提升頻譜使用的效率。本實驗乃利用常見之微波元件設計出可與寬頻射頻收發機整合之寬頻高隔離度射頻切換模組，該模組可支援兩個分量載波(Component Carriers, CCs)，並藉由設定不同的切換路徑實現頻帶內(Intra-band)和跨頻帶(Inter-band)之載波聚合。

## 二. 實驗原理

本實驗所提出之載波聚合開關模組包含功率分配器，射頻前置放大器，RF 開關和功率合成器等元件，如圖 1 所示。射頻前置放大器用於補償訊號經過功率分配器後所產生的損耗，射頻開關和功率合成器用來選擇適當的路徑以實現所需的載波聚合模式。

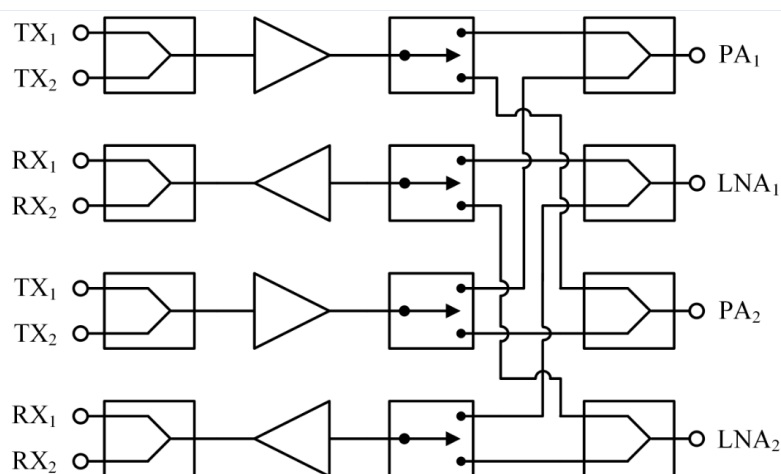


圖 1、本實驗載波聚合開關模組方塊圖

在發射路徑上，頻帶內和跨頻帶的載波聚合操作模式可以透過控制射頻開關選擇相對應的路徑來獲得，如圖 3.2 和圖 3.3 所示。此模組位於功率放大器之輸入端，相較於位於功率放大器輸出端之 T/R 射頻開關器所需之損耗與線性度要求較不嚴苛。

# 實驗 1 單級毫米波頻率合成器實驗

## 一. 實驗目的

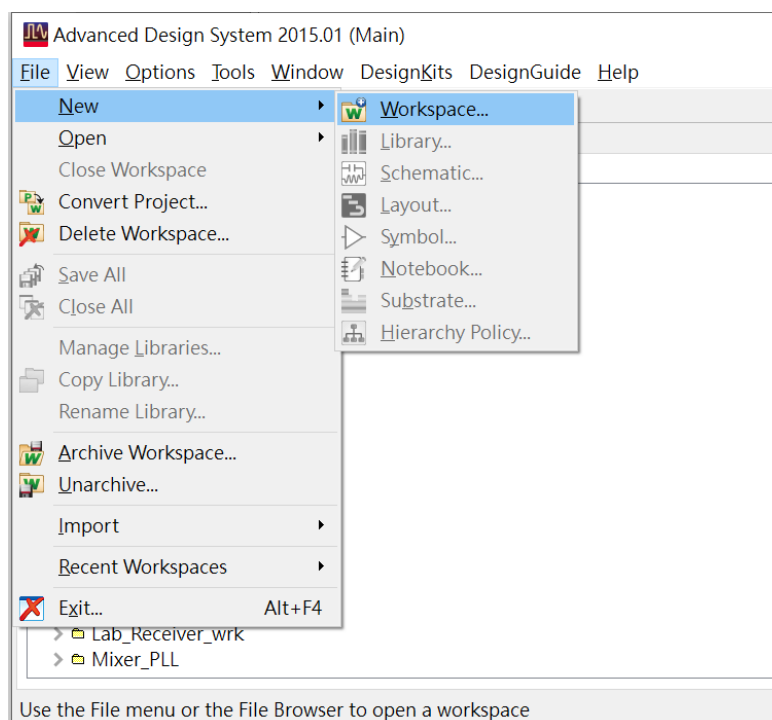
本實驗之目的在於學習並瞭解單級(single stage)毫米波頻率合成器的電路架構，透過運用 Keysight 高頻模擬軟體 ADS 實際設計一個操作頻段為 28 GHz 之單級毫米波頻率合成器。藉由 Keysight ADS 模擬所設計之 28 GHz 單級毫米波頻率合成器的頻譜輸出、鎖入時間以及相位雜訊，透過設計不同之頻率合成器迴路頻寬以及參考頻率以觀察頻率合成器之鎖入時間以及相位雜訊之變化。

## 二. 實驗設備

1. 個人電腦
2. Keysight ADS

## 三. 頻率合成器設計與模擬

1. 建立一個新的 Workspace(File->New->Workspace)



## 實驗 2 倍頻毫米波頻率合成器實驗

### 一. 實驗目的

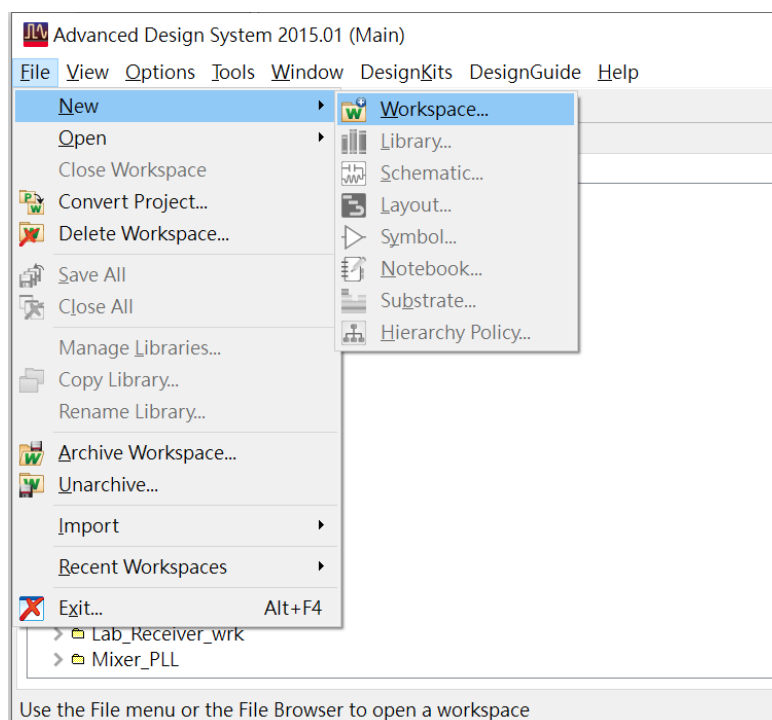
本實驗之目的在於學習並瞭解倍頻(frequency-multiplied)毫米波頻率合成器的電路架構，透過運用 Keysight 高頻模擬軟體 ADS 實際設計一個操作頻段為 28 GHz 之倍頻毫米波頻率合成器。藉由 Keysight ADS 模擬所設計之 28 GHz 倍頻毫米波頻率合成器的頻譜輸出、鎖入時間以及相位雜訊，透過設計不同之頻率合成器迴路頻寬以及參考頻率以觀察頻率合成器之鎖入時間以及相位雜訊之變化。

### 二. 實驗設備

1. 個人電腦
2. Keysight ADS

### 三. 頻率合成器設計與模擬

1. 建立一個新的 Workspace(File->New->Workspace)



## 實驗 3 串級毫米波頻率合成器實驗

### 一. 實驗目的

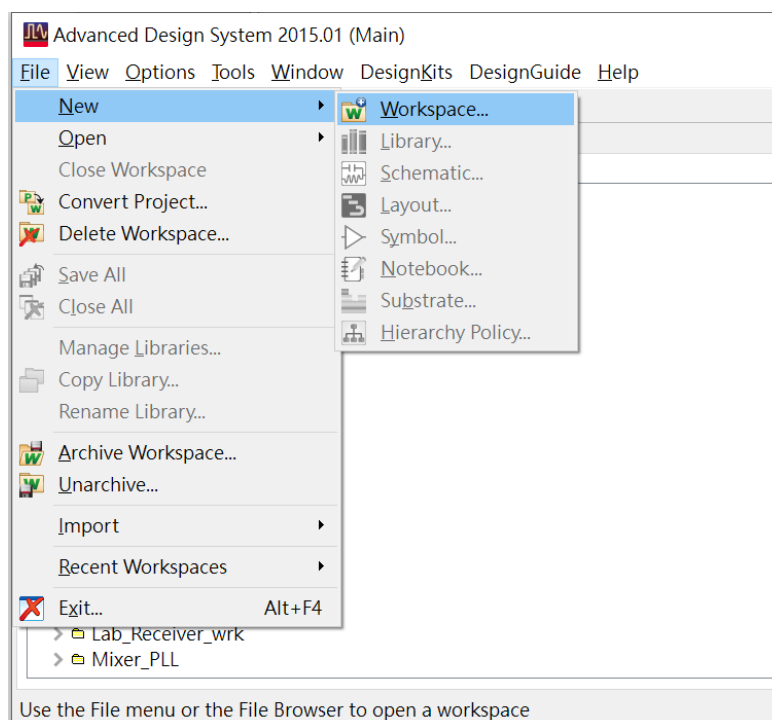
本實驗之目的在於學習並瞭解串級(cascade)毫米波頻率合成器的電路架構，透過運用 Keysight 高頻模擬軟體 ADS 實際設計一個操作頻段為 28 GHz 之串級毫米波頻率合成器。藉由 Keysight ADS 模擬所設計之 28 GHz 串級毫米波頻率合成器的頻譜輸出、鎖入時間以及相位雜訊，透過設計不同之頻率合成器迴路頻寬以及參考頻率以觀察串級毫米波頻率合成器之鎖入時間以及相位雜訊之變化。

### 二. 實驗設備

1. 個人電腦
2. Keysight ADS

### 三. 頻率合成器設計與模擬

1. 建立一個新的 Workspace(File->New->Workspace)



# 寬頻收發關鍵技術模組 - 功率放大器預失真線性化實驗

黃建彰  
元智大學電機系乙組



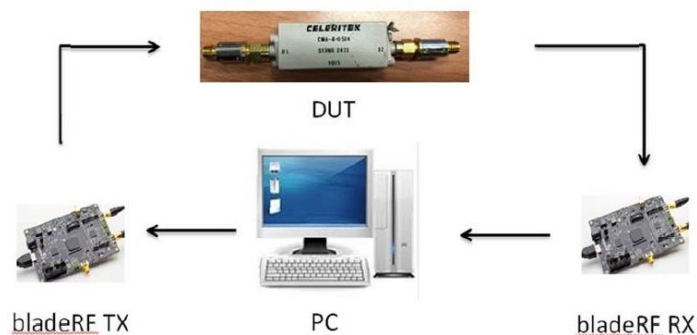
# 使用工具基本資訊

- 本實驗使用Matlab及Simulink作為平台。
- Nuand BladeRF x40 軟體定義無線電
- Celeritek Inc CMA-4-0504 功率放大器



# 實驗平台架設

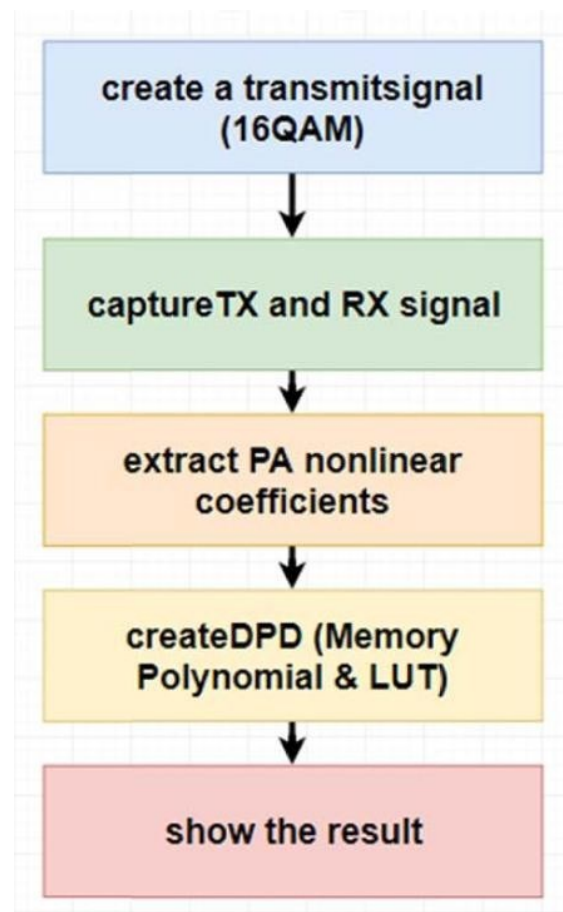
- BladeRF 連接至電腦
- BladeRF TX端連接至PA之input端
- PA之output端連接一衰減器後接至BladeRF之 RX端



# 實驗流程

主要流程會分為五步驟：

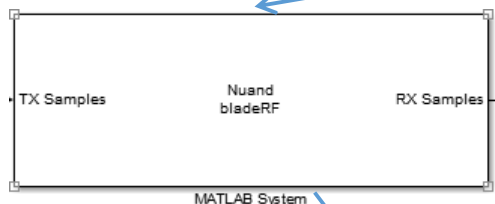
1. 產生調變訊號
2. 接收TX及RX訊號
3. 推導出功率放大器預失真係數
4. 創建DPD並應用到輸出訊號
5. 結果展示






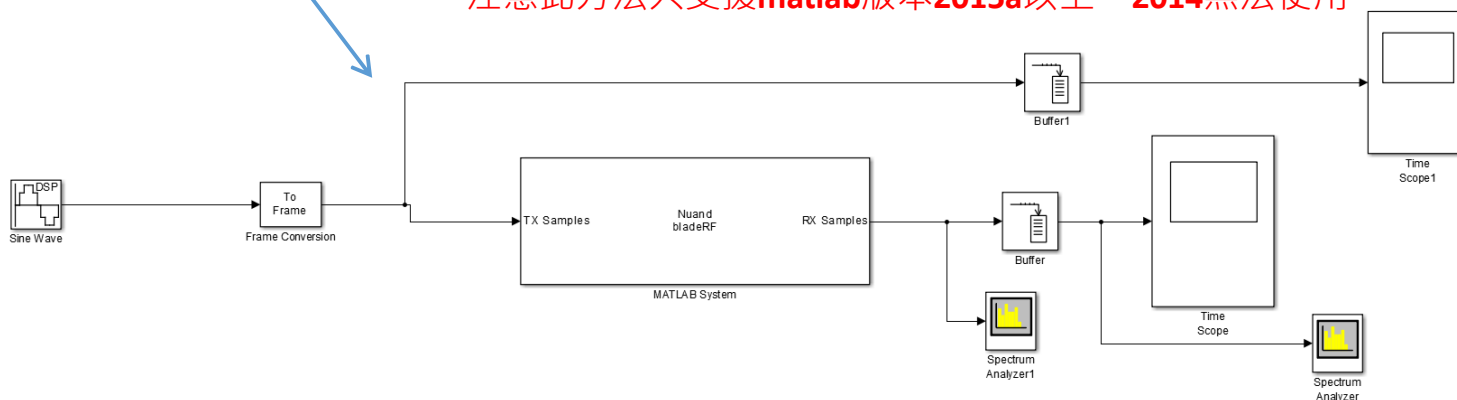
# BLADERF操作方法

操作方法主要分兩種，第一種為叫出並使用**SIMULINK**的**MODEL**，詳細方法看下面文件



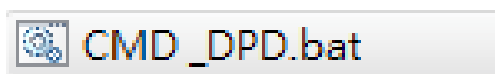
 bladeRF\_windows\_installer.pdf

**\*\*注意此方法只支援matlab版本2015a以上，2014無法使用**



# BLADERF操作方法

延續第二種方法，更為簡單的方式為，先寫好一個副檔名為**.bat**的檔案，在執行時一併執行**.txt**底下的內容。



→ E:  
cd E:\Users\robert\Desktop\DPD\_robert \_4\20180222\19\_-17\_18\_2  
bladerf-cli -s Test2400m.txt



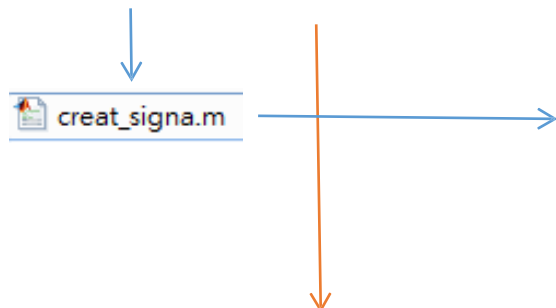
←  
set frequency 2400M  
set samplerate 20M  
set txvga1 -17  
set txvga2 18  
set lnagain 0  
set bandwidth 12000000  
tx config file=16QAM.csv format=csv repeat=10000 delay=0  
rx config file=16QAM\_return.csv format=csv n=32768  
tx start  
rx start|



# BladeRF 創建輸出訊號

以下將介紹3種BLADERF輸出訊號如何產生，分別為Single Tone, AM, 16QAM

## 1.Single Tone可參考檔案及網站



<https://github.com/Nuand/bladeRF/wiki/bladeRF-CLI-Tips-and-Tricks>

```
1 % Our samples must be generated at the samplerate we plan to run the device at
2 - SAMPLE_RATE = 2e6;
3
4 % Generate 10 seconds worth of samples. Bear in mind that when using the
5 % binary SC16 Q11 format, 1 sample consumes 4 bytes of memory/disk space.
6 % This quickly adds up - 10 seconds @ 2 Mps yields ~ 76.3 MiB. Be careful
7 % when using higher sample rates!
8 - NUM_SECONDS = 10;
9 - NUM_SAMPLES = NUM_SECONDS * SAMPLE_RATE;
10
11 % 250 kHz, in radians ( $\omega = F_{Hz} * 2\pi$ )
12 - SIGNAL_FREQ_RAD = 250e3 * 2 * pi;
13
14 % Generate a vector "t" which represents time, in units of samples.
15 % This starts at t=0, and creates NUM_SAMPLES samples in steps of 1/SAMPLE_RATE
16 - t = [ 0 : (1/SAMPLE_RATE) : (NUM_SECONDS - 1/SAMPLE_RATE) ];
17
18 % Create a sinusoid (signal =  $e^{j\omega t}$ ) with a magnitude of 0.90
19 - signal = 0.90 * exp(1j * SIGNAL_FREQ_RAD * t);
20
21 % Plot the FFT of our signal as a quick sanity check.
22 % The NUM_SAMPLES denominator is just to normalize this for display purposes.
23 - f = linspace(-0.5 * SAMPLE_RATE, 0.5 * SAMPLE_RATE, length(signal));
24 - plot(f, 20*log10(abs(fftshift(fft(signal)))/NUM_SAMPLES));
25 - xlabel('Frequency (Hz)');
26 - ylabel('Power (dB)');
27 - title('250 kHz tone');
```



# DPD操作流程

介紹完BLADERF如何使用及創造訊號後，要介紹如何使用Digital predistortion, DPD。

DPD主要分兩種，一種為Lookup table,LUT, 另一種為Memory polynomial。

在實現Lookup table,LUT及Memory polynomial上，我分別做了模擬情況(Simulation)及實際情況(Measurement)，用來交叉比對結果。

建議先看過以下幾份文件比較知道流程及運作方法

。

[https://www.mathworks.com/content/dam/mathworks/mathworks-dot-com/company/events/conferences/matlab-tour-china/2015/proceedings/matlab-rf-development-application-in-numbers\\_en.pdf](https://www.mathworks.com/content/dam/mathworks/mathworks-dot-com/company/events/conferences/matlab-tour-china/2015/proceedings/matlab-rf-development-application-in-numbers_en.pdf)

<https://ww2.mathworks.cn/videos/digital-predistortion-for-power-amplifiers-68727.html>

<https://ww2.mathworks.cn/matlabcentral/fileexchange/45890-adaptive-dpd-design>

[https://www.keysight.com/upload/cmc\\_upload/All/DigitalPreDistortion\\_MicroApps.pdf](https://www.keysight.com/upload/cmc_upload/All/DigitalPreDistortion_MicroApps.pdf)

# DPD操作流程 (Simulation)

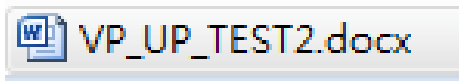
在**Simulation**上，做了五種情況，放置在不同資料夾中。分別為

資料夾名稱      情況

DPD	測試 <b>Simulation 16QAM memory</b>
DPD2	測試 <b>I/Q memory</b>
DPD3	測試 <b>16QAM LUT</b>
DPD_2	測試 <b>12BIT 16QAM memory</b>
DPD_2 - 複製	測試 <b>16BIT 16QAM memory</b>

使用的檔案順序皆在其資料夾底下的**VP\_UP\_TEST3**文件檔裡面，文件檔展示了流程及效果。

順帶一提，從**Simulation**可以發現，**12BITS**的**DAC**精準度不足以達到完美的預失真反函數，而**16BITS**上則可以實現。



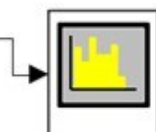
找到相對應的檔案，依序執行一次



(DPD 資料夾) QAM\_CODE\_TEST<sup>1</sup> data\_TEST<sup>2</sup> QAM\_CODE\_TEST<sup>3</sup>  
DPD\_ANALYSIS<sup>4</sup> DPD<sup>5</sup> DATA\_PA3\_SLX<sup>6</sup>↓  
data=32768↓

data\_PA\_in

From  
Workspace1



PA\_IN\_SPEC

1 QAM\_code\_test.m

2 data\_TEST\_slx.slx

3 QAM\_code\_test.m

4 DPD\_analysis.m

5 DPD.m

6 data\_PA3\_slx.slx



# Measurements Part 1

**\*\*Part 1**主要介紹如何開啟，讀寫以前**BLADERF**儲存在電腦上的數值，並在**MATLAB**及**SIMULINK**作呈現。



# DPD操作流程 (measurement)

首先選定一個PA，我們使用此PA做例子。

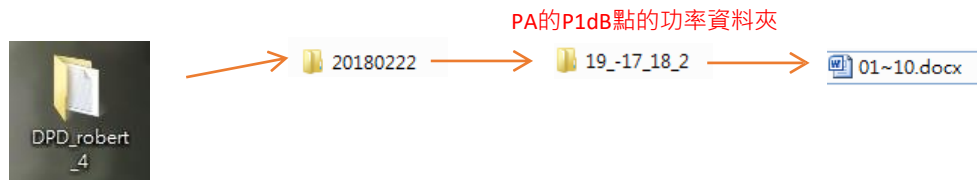


**DATASHEET**如右圖

## Specifications:

Model Number: CMA-4-0504  
Frequency Range: 0.5 GHz – 4 GHz  
Gain (min): 35 dB  
Gain Vs Temp at any Frequency: This Amplifier is not Temperature Compensated  
Noise Figure (max): 4 dB  
Output Power @ 1dB Compression: 18 dBm  
Gain Flatness Vs Frequency: +/-1.5 dB  
3<sup>rd</sup> Order Intercept (Typ): 27 dBm  
Power Supply:  
    Input Voltage: 11.5 to 15 VDC  
    Current @ 12 VDC (typ): 240 mA  
    Current @ 12 VDC (max): 265 mA  
Case Type (See attached Outline): SC4  
Temperature Range: +25

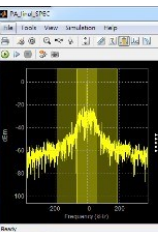
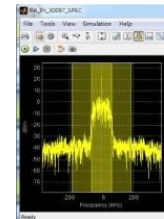
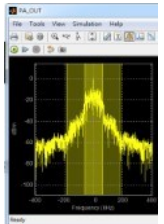
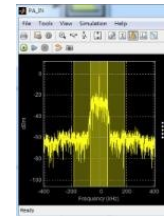
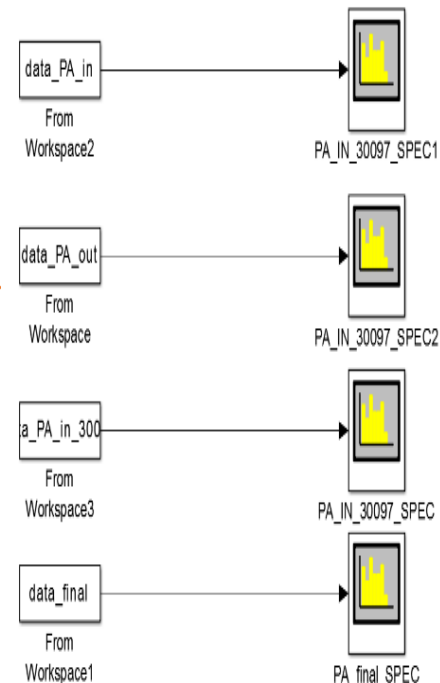
# Measurement (Memory Polynomial) Part 1



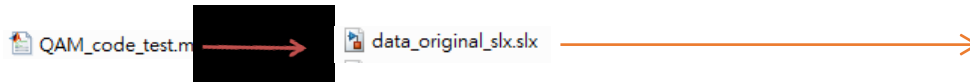
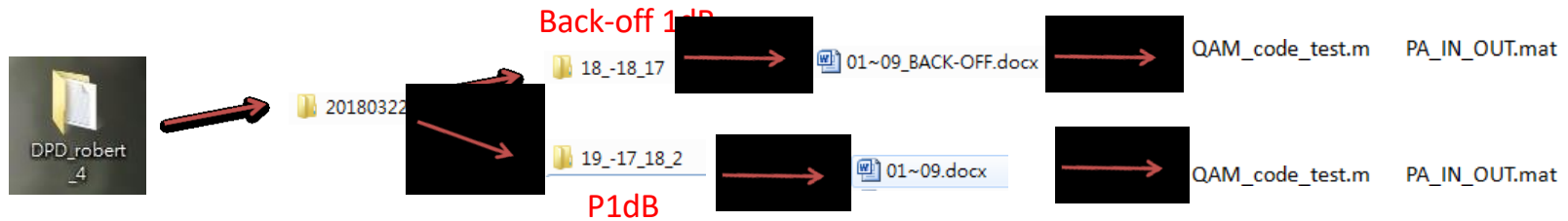
QAM\_CODE\_TEST DPD\_analysis DPD untitled3  
data\_PA2\_slx

QAM\_code\_test.m x DPD\_analysis.m x DPD.m x Untitled3.m x → data\_PA2\_slx.slx

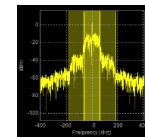
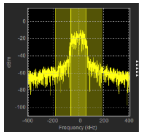
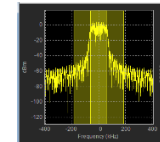
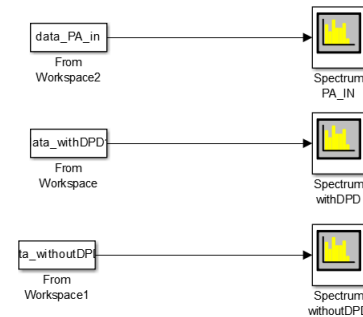
\*\*這部分流程，是觀測之前已經建立在電腦裡的DATA。  
如要重新量測MEMORY POLYNOMIAL，請參照PART 2



# Measurement (LUT) Part 1



**\*\*這部分流程，是觀測之前已經建立在電腦裡的DATA。  
如要重新量測LUT，請參照PART 2**



# Measurement (Memory Polynomial) Part 2

到 20180222

接上



運行

CMD.bat

自動抓取資料

```
Test2400m.txt - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)
set frequency 2400M
set samplerate 20M
set txvga1 -17
set txvga2 18
set lnagain 0
set bandwidth 12000000
tx config file=16QAM.csv format=csv repeat=10000 delay=0
rx config file=16QAM_return.csv format=csv n=32768
tx start
rx start|
```

RX資料送到這

**\*\*路徑要改**

```
CMD.bat - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)
E:
cd E:\Users\robert\Desktop\DPD_robert_4\20180222\19_-17_18_2
bladerf-cli -s Test2400m.txt
```



# Measurement (Memory Polynomial) Part 2

接上



運行

CMD\_DPD.bat

自動抓取資料

```
Test2400m_DPD.txt - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)

set frequency 2400M
set samplerate 20M
set txvga1 -17
set txvga2 18
set lnagain 0
set bandwidth 12000000

tx config file=16QAM.csv format=csv repeat=10000 delay=0
rx config file=16QAM_return_DPD.csv format=csv n=32768
tx start
rx start
```

調整這部分數值，來找到PA的P1dB  
這部分要跟上頁一致

RX資料送到這


\*\*路徑要改

```
CMD_DPD.bat - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)


E:
cd E:\Users\robert\Desktop\DPD_robert_4\20180222\19_-17_18_2
bladerf-cli -s Test2400m_DPD.txt
```

**\*\*將資料寫入並處理**

**\*\*推導出預失真係數**

運行  QAM\_code\_test.m

運行

 DPD\_analysis.m

```
QAM_code_test.m
1 - load time;
2
3 %AM original data in I/Q mode
4 - QAMTEST=csvread('16QAM.csv');
5 - original=(QAMTEST./2048);
6
7 - IW=original;
8 - IW(:, 2) = [];
9 - QW=original;
10 - QW(:, 1) = [];
11 - original2=IW+1j.*QW;
12
13 - WW = repmat(original2,[1 1 length(tout)]);
14 - data_original.time=tout;
15 - data_original.signals.values = WW;
16 - data_original.signals.dimensions=[5068 1];
17
18 - WWW = repmat(original,[1 1 length(tout)]);
19 - data_original2.time=tout;
20 - data_original2.signals.values = WWW;
21 - data_original2.signals.dimensions=[5068 2];
22
23 %AM without PA data in I/Q mode
24 - QAM_return=csvread('16QAM_return.csv');
25 - withoutPA=(QAM_return./2048);
26
27 - IX=withoutPA;
28 - IX(:, 2) = [];
```

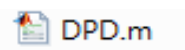
```
DPD_analysis.m
1 % Copyright 2013-2014 The MathWorks, Inc.
2 %% Test data used to derive coeffs
3 - pa_in = pa_in;
4 - pa_out = pa_out;
5
6 - toss = 500; % ignore initial data that could have transients
7 - pad = 600; % room to play at the end of the data set.
8 - traininglen = 30e3; % Arbitrary subset of pa_in data used to derive coeffs
9 - if traininglen > (length(pa_in)-pad-toss)
10 - error('Pick smaller subset for traininglen');
11 - end
12 %% PA and DPD model parameters
13 - memorylen_pa = 3; %M
14 - degree_pa = 3; %K
15 - degree_pd = 3;
16 - memorylen_pd = 1;
17
18 - coef_pd=complex(zeros(degree_pd*memorylen_pd,1));
19 - coef_pa=complex(zeros(degree_pa*memorylen_pa,1));
20 %% Compute input scaling factor with overrange
21 - u = pa_in(toss+1:(traininglen+pad)); % select data and transpose
22 - umax_inv = 1/(max(abs(u)));
23 - u = u*umax_inv;
24 - v = pa_out(toss+1:(traininglen+pad)); % select data
25 - vmax_inv = 1/(max(abs(v)));
26 - v = v*vmax_inv;
27 - if 1
28 - offset = 3;
```

**\*\*修改上方數字，以求得不同的K**  
**\*\*修改下方數字，以求得不同的M**



**\*\*將預失真係數重新結合16QAM訊號，並轉成CSV檔案**

運行



```
DPD.m x +
11
12     pipe(1:end-1) = pipe(2:end);
13     pipe(end) = up(n);
14
15     for k = 1:K
16         for m = 1:M
17             y(n) = y(n) + coef_pd((k-1)*M+m)*pipe(m)*abs(pipe(m))^(k-1);
18         end
19     end
20
21
22 end
23
24 %AM data withDPD
25 yy= repmat(y,[1 1 length(tout)]);
26 data_withDPD.time=tout;
27 data_withDPD.signals.values = yy;
28 data_withDPD.signals.dimensions=[30097 1];
29
30 IIX=real(y);
31 QQX=imag(y);
32 pa_in_DPD=complex(zeros(30097,2));
33 pa_in_DPD(:,1) = [IIX];
34 pa_in_DPD(:,2) = [QQX];
35
36 pa_in_DPD2=round(pa_in_DPD.*2048)
37 csvwrite('pa_in_DPD2.csv',pa_in_DPD2);
38
```



接上



運行

CMD\_DPD\_FORPA.bat

自動抓取資料

Test2400m\_DPD\_FORPA.txt - 記事本

檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)

```
set frequency 2400M
set samplerate 20M
set txvga1 -17
set txvga2 18
set lnagain 0
set bandwidth 12000000

tx config file=pa_in_DPD2.csv format=csv repeat=10000 delay=0
rx config file=pa_in_DPD3.csv format=csv n=30097
tx start
rx start
```

\*\*路徑要改

RX資料送到這

CMD\_DPD\_FORPA.bat - 記事本

檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)

```
E:
cd E:\Users\robert\Desktop\DPD_robot_4\20180222\19_-17_18_2
bladerf-cli -s Test2400m_DPD_FORPA.txt
```





**\*\*將資料寫入並處理**

**\*\*SIMULINK抓到資料並出圖**

運行

Untitled3.m

```

1 %AM data withDPD
2 yy= repmat(y,[1 1 length(tout)]);
3 data_withDPD.time=tout;
4 data_withDPD.signals.values = yy;
5 data_withDPD.signals.dimensions=[30097 1];
6
7 IIX=real(y);
8 QQX=imag(y);
9 pa_in_DPD=complex(zeros(30097,2));
10 pa_in_DPD(:,1) = [IIX];
11 pa_in_DPD(:,2) = [QQX];
12
13 pa_in_DPD2=round(pa_in_DPD.*2048);
14 csvwrite('pa_in_DPD2.csv',pa_in_DPD2);
15
16
17 %AM data DPD+PA
18 final2=csvread('pa_in_DPD3.csv');
19 final3=(final2./2048);
20
21 IZ=final3;
22 IZ(:, 2) = [];
23 QZ=final3;
24 QZ(:, 1) = [];
25 final4=IZ+ij.*QZ;
26
27 ZZ = repmat(final4,[1 1 length(tout)]);
28 data_final.time=tout;
  
```

data\_PA2\_slx.slx

doc\_mer\_and\_evm2.slx

data\_PA\_in  
From  
Workspace2

data\_PA\_out  
From  
Workspace

a\_PA\_in\_300  
From  
Workspace3

data\_final  
From  
Workspace1

PA\_IN\_30097\_SPEC1

PA\_IN\_30097\_SPEC2

PA\_IN\_30097\_SPEC

PA\_final\_SPEC

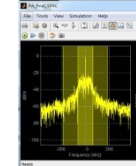
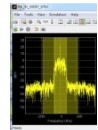
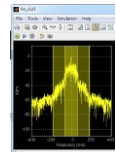
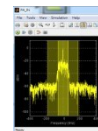
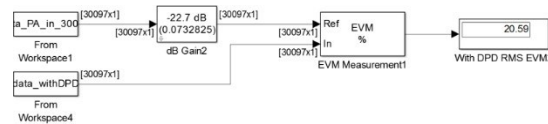
TX

RX

TX\_30097

DPD\_RX

**\*\*查看EVM**



教育部5G行動寬頻人才培育計畫  
5G天線與射頻技術聯盟中心

# Measurement (LUT) Part 2

到 20180322

接上



運行

CMD.bat

自動抓取資料

```
Test2400m.txt - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)

set frequency 2400M
set samplerate 20M
set txvga1 -17
set txvga2 18
set lnagain 0
set bandwidth 12000000


tx config file=l6QAM.csv format=csv repeat=10000 delay=0
rx config file=l6QAM_return.csv format=csv n=32768
tx start
rx start
```

\*\*路徑要改

```
CMD.bat - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)

E:
cd E:\Users\robert\Desktop\DPD_robert_4\20180322\19_-17_18_2
bladerf-cli -s Test2400m.txt
```



運行  QAM\_code\_test.m

**\*\*修改IQ的振幅，使增益保持線性，  
並重新出成CSV檔案格式**

```
QAM_code_test.m  x  +
1 - load time;
2 - load PA_IN_OUT;
3
4 %AM original data in I/Q mode
5 - QAMTEST=csvread('16QAM.csv');
6 - original=(QAMTEST./2048);
7 - original_08=round(QAMTEST.*0.7);
8 - csvwrite('16QAM_08.csv',original_08);
9
10
11
12 |
13 - IW=original;
14 - IW(:, 2) = [];
15 - QW=original;
16 - QW(:, 1) = [];
17 - original2=IW+1j.*QW;
18
19 - WW = repmat(original2,[1 1 length(tout)]);
20 - data_original.time=tout;
21 - data_original.signals.values = WW;
22 - data_original.signals.dimensions=[5068 1];
23
24 - WWW = repmat(original,[1 1 length(tout)]);
25 - data_original2.time=tout;
26 - data_original2.signals.values = WWW;
27 - data_original2.signals.dimensions=[5068 2];
28
```



接上



運行

CMD\_withDPD\_08.bat

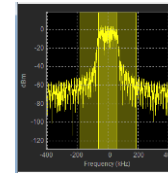
自動讀取資料

\*\*路徑要改

```
CMD_withDPD_08.bat - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)
E:
cd E:\Users\robert\Desktop\DPD_robert_4\20180322\19_-17_18_2
bladerf-cli -s Test2400m_08.txt
```

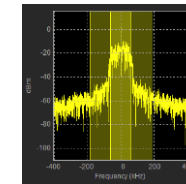
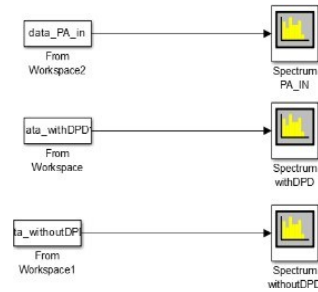
```
Test2400m_08.txt - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)
set frequency 2400M
set samplerate 20M
set txvga1 -17
set txvga2 18
set lnagain 0
set bandwidth 12000000
tx config file=16QAM_08.csv format=csv repeat=10000 delay=0
rx config file=16QAM_return_DPD.csv format=csv n=32768
tx start
rx start
```



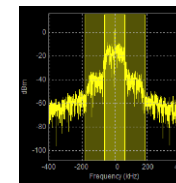


待傳送訊號

data\_original\_slx.slx

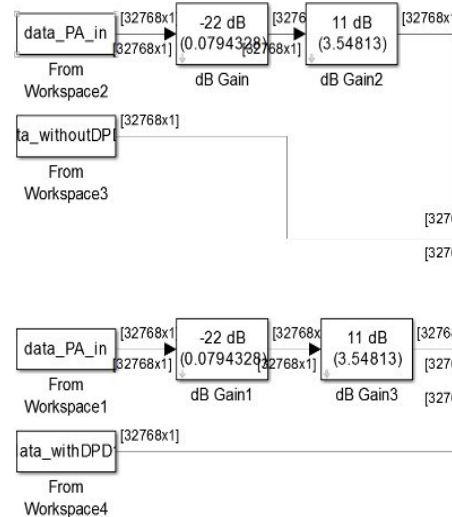


RX\_WITH DPD

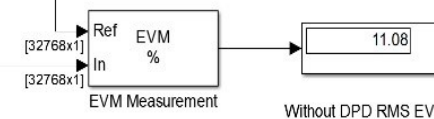


RX\_WITHOUT DPD

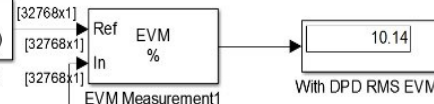
data\_original\_slx.slx



**\*\*查看EVM**



Without DPD RMS EVM1



With DPD RMS EVM2



教育部5G行動寬頻人才培育計畫  
5G天線與射頻技術聯盟中心

# 注意事項

- 以上所使用檔案之檔案路徑為此次示範所使用，請依實際檔案位置更改路徑。
- 請確實於RX端接上衰減器。如未接上衰減器RX端可能有損壞的風險(此款BladeRF所使用之RF傳收機RX最大接收功率為23dBm)