



# **JUICE – JU**piter **IC**y Moons **E**xplorer **SWI – Sub**millimetre **W**ave **I**nstrument

FM LO 600 GHz test

Version 0.3 of 2020-09-24, by Michael Olberg

## **Abstract**

Results from FM LO scans September, 2020.

Technical Note: SWI/2020-xxx  
191 pages

# 1 Introduction

The FM 600 GHz LO chain was scanned for all possible combinations of MPA, FRM and bias settings in order to record the resulting output power levels. The procedure used the Python script listed in Appendix C.

Measurements were carried out on two sets of dates, Sep 4–7 and Sep 14–23. We look at statistics of achieved power levels as well as repeatability of results.

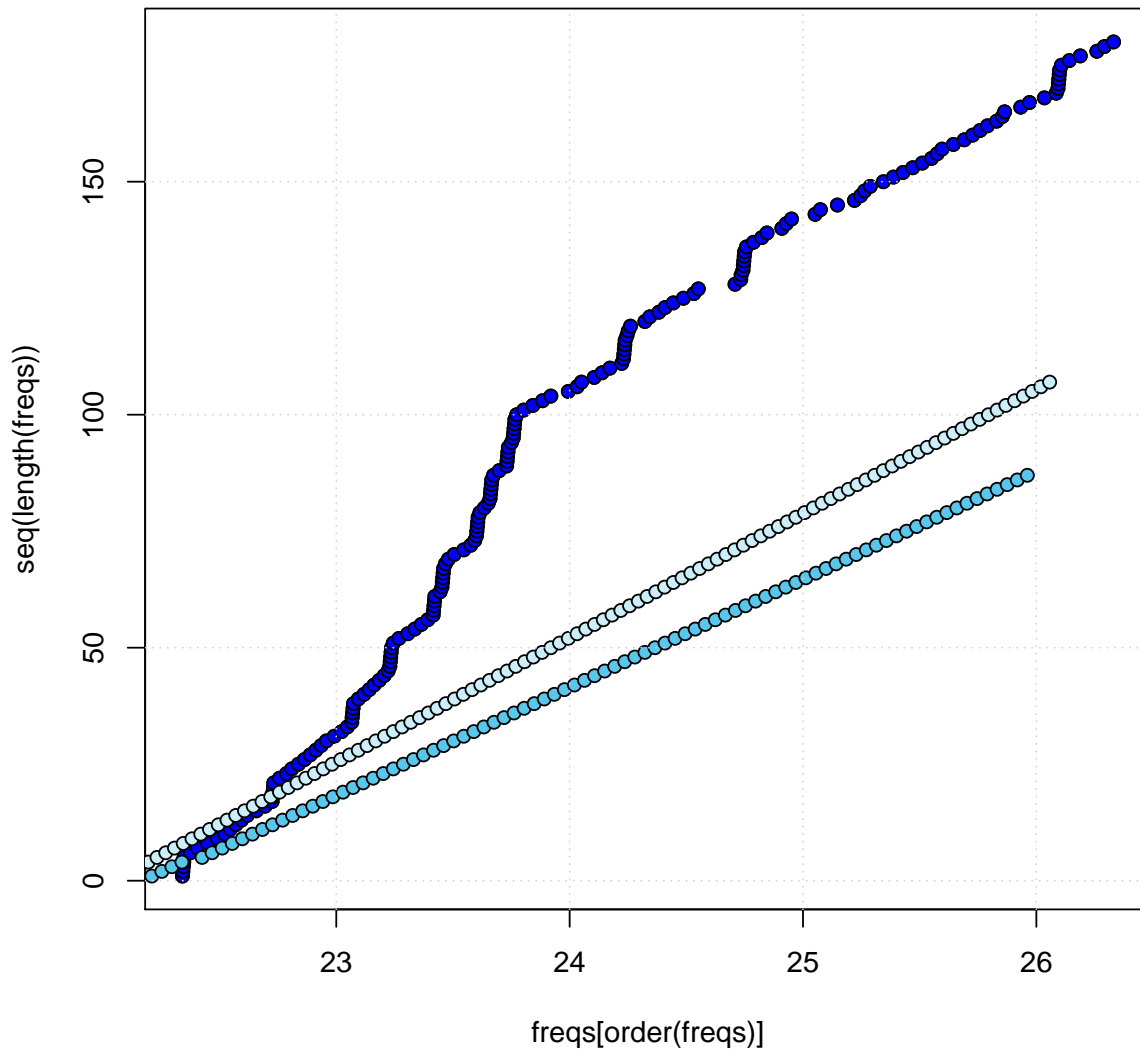


Figure 1: Scanned frequencies of the FM LO chain (dark blue dots). Dots in light blue colours show frequencies that were scanned in TVAC for the EM-2 LO chain on a more regular, but sparser grid.

## 2 Statistics on output power levels

As a strategy for automatic tuning of the 600 GHz receiver, it has been suggested to search among those settings, which produce LO output powers of  $2.0 \pm 1.0$  mW. The histogram in Fig 2 shows the distribution of the observed output power levels during the FM LO scan.

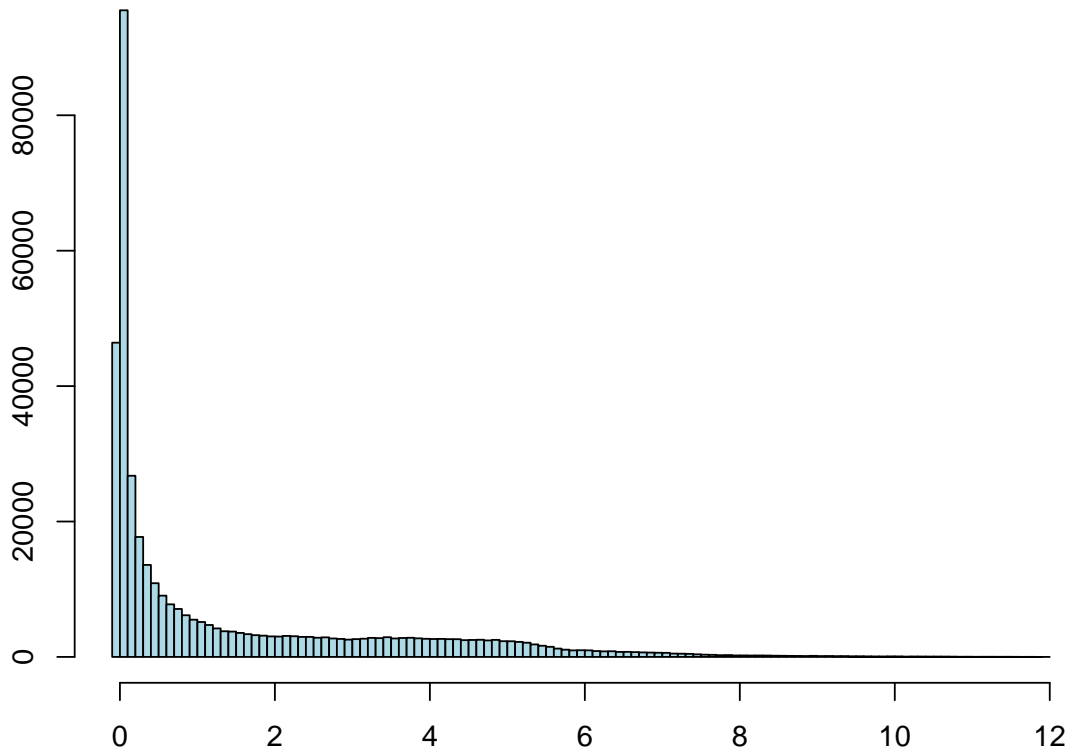


Figure 2: Histogram over achieved LO output power.

Of 396675 measurements, 37905 (9.6%) produce output powers in the range of 1–2 mW, and 28743 (7.2%) in the range of 2–3 mW. A detailed table over how many the histogram counts broken down by frequency can be found in Appendix A.

### 3 Repeatability

The last frequency to be scanned, before the scans were interrupted due to intervening EM-2 RU tests, was 23.14125 GHz. That same frequency was also scanned when the procedure was resumed on Sep 14, so there are two data sets for this frequency. Fig 3 shows the results from those two scans in comparison.

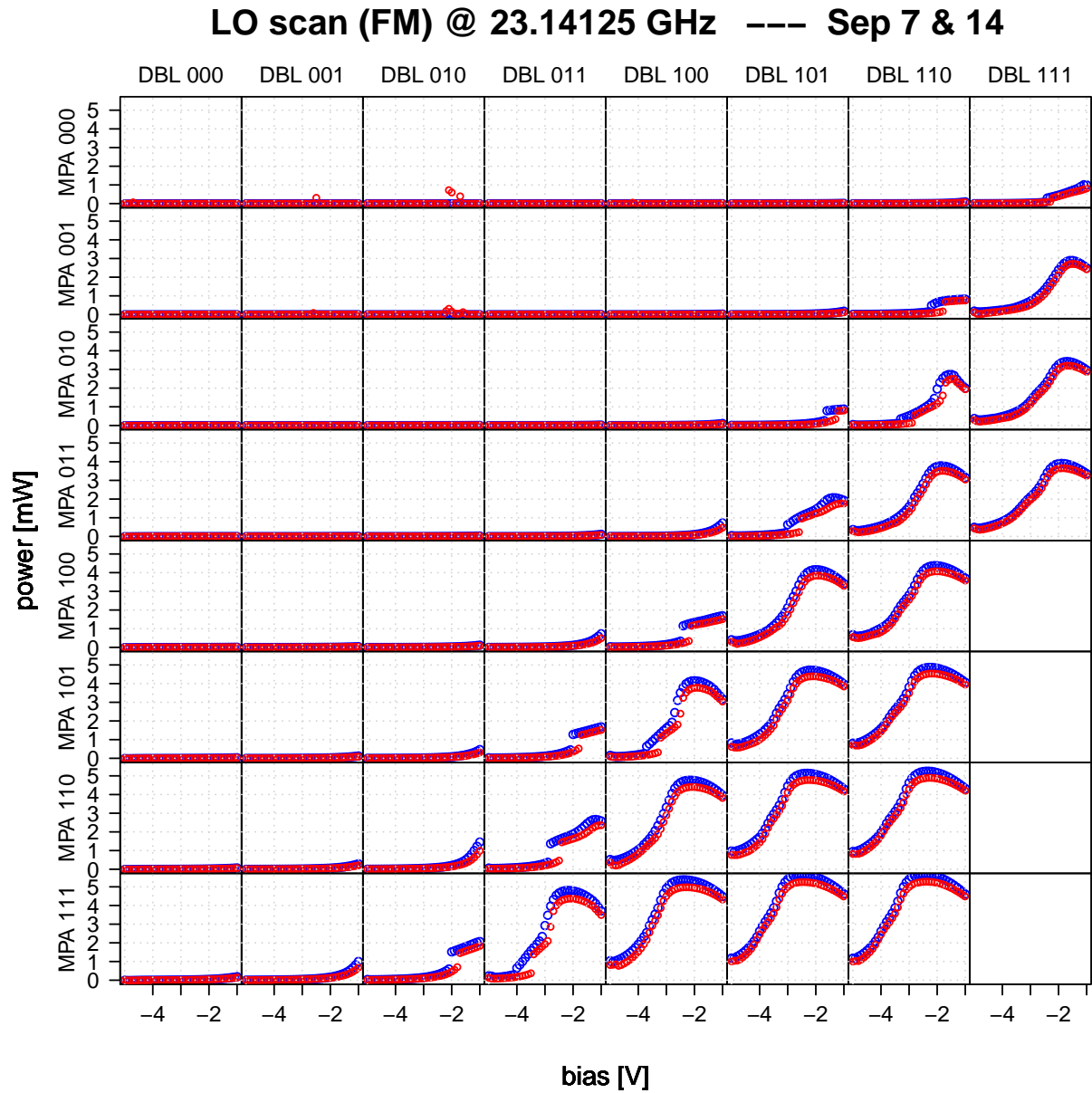


Figure 3: Comparison of LO scans at 23.14125GHz on two different dates. Data from Sep 7 shown in blue, Sep 14 in red.

## A Power level statistics

	-1..0	0..1	1..2	2..3	3..4	4..5	5..6	6..7	7..8	8..9	9..10	10..11	11..12
22.340625	1514	752	93	142	0	0	0	0	0	0	0	0	0
22.342500	1694	569	93	145	0	0	0	0	0	0	0	0	0
22.344375	1673	580	94	154	0	0	0	0	0	0	0	0	0
22.346250	1659	589	93	160	0	0	0	0	0	0	0	0	0
22.348125	1650	591	94	166	0	0	0	0	0	0	0	0	0
22.374375	1436	749	88	216	12	0	0	0	0	0	0	0	0
22.404375	1163	908	103	282	45	0	0	0	0	0	0	0	0
22.447500	671	1106	171	237	316	0	0	0	0	0	0	0	0
22.490625	241	1115	291	255	309	266	24	0	0	0	0	0	0
22.522500	63	1478	193	211	308	299	72	0	0	0	0	0	0
22.545000	55	1571	219	218	336	225	0	0	0	0	0	0	0
22.569375	422	1381	201	193	320	107	0	0	0	0	0	0	0
22.593750	650	1277	168	164	263	102	0	0	0	0	0	0	0
22.618125	737	1303	155	134	223	72	0	0	0	0	0	0	0
22.657500	885	1137	175	121	223	83	0	0	0	0	0	0	0
22.695000	810	1148	196	129	210	131	0	0	0	0	0	0	0
22.725000	626	1204	250	150	150	191	53	0	0	0	0	0	0
22.726875	565	1255	258	149	149	186	62	0	0	0	0	0	0
22.728750	525	1276	264	154	145	188	72	0	0	0	0	0	0
22.730625	538	1246	278	149	153	183	77	0	0	0	0	0	0
22.732500	500	1268	287	155	151	178	85	0	0	0	0	0	0
22.756875	287	1479	283	157	172	159	87	0	0	0	0	0	0
22.786875	122	1901	139	86	129	152	95	0	0	0	0	0	0
22.809375	26	2044	94	70	129	173	88	0	0	0	0	0	0
22.837500	656	1369	127	94	150	214	14	0	0	0	0	0	0
22.863750	1027	955	147	96	127	227	45	0	0	0	0	0	0
22.888125	1004	989	136	117	123	212	43	0	0	0	0	0	0
22.912500	1015	913	155	93	125	182	59	0	0	0	0	0	0
22.935000	692	1180	182	108	134	177	69	0	0	0	0	0	0
22.957500	605	1231	194	120	136	162	94	0	0	0	0	0	0
22.991250	533	1119	257	153	149	169	152	10	0	0	0	0	0
23.023125	475	1030	272	232	173	137	140	83	0	0	0	0	0
23.047500	328	1331	228	173	179	114	105	74	10	0	0	0	0
23.066250	209	1626	216	126	122	117	85	41	0	0	0	0	0
23.068125	235	1622	210	117	125	110	84	39	0	0	0	0	0
23.070000	308	1578	194	115	122	106	84	35	0	0	0	0	0
23.071875	263	1660	168	110	120	105	84	32	0	0	0	0	0
23.073750	196	1741	156	109	120	108	86	26	0	0	0	0	0
23.094375	533	1477	119	105	138	123	47	0	0	0	0	0	0
23.118750	1001	941	142	95	130	122	29	0	0	0	0	0	0
23.141250	1031	832	184	95	102	143	73	0	0	0	0	0	0
23.161875	811	1102	154	102	101	152	38	0	0	0	0	0	0
23.184375	552	1336	166	97	125	157	27	0	0	0	0	0	0
23.205000	85	1732	193	97	139	173	41	0	0	0	0	0	0
23.221875	612	1102	243	88	148	179	88	0	0	0	0	0	0
23.229375	755	940	240	103	143	175	104	0	0	0	0	0	0
23.231250	556	1128	243	105	135	177	116	0	0	0	0	0	0
23.233125	702	1012	231	100	155	192	68	0	0	0	0	0	0
23.235000	78	1583	245	112	133	174	135	0	0	0	0	0	0

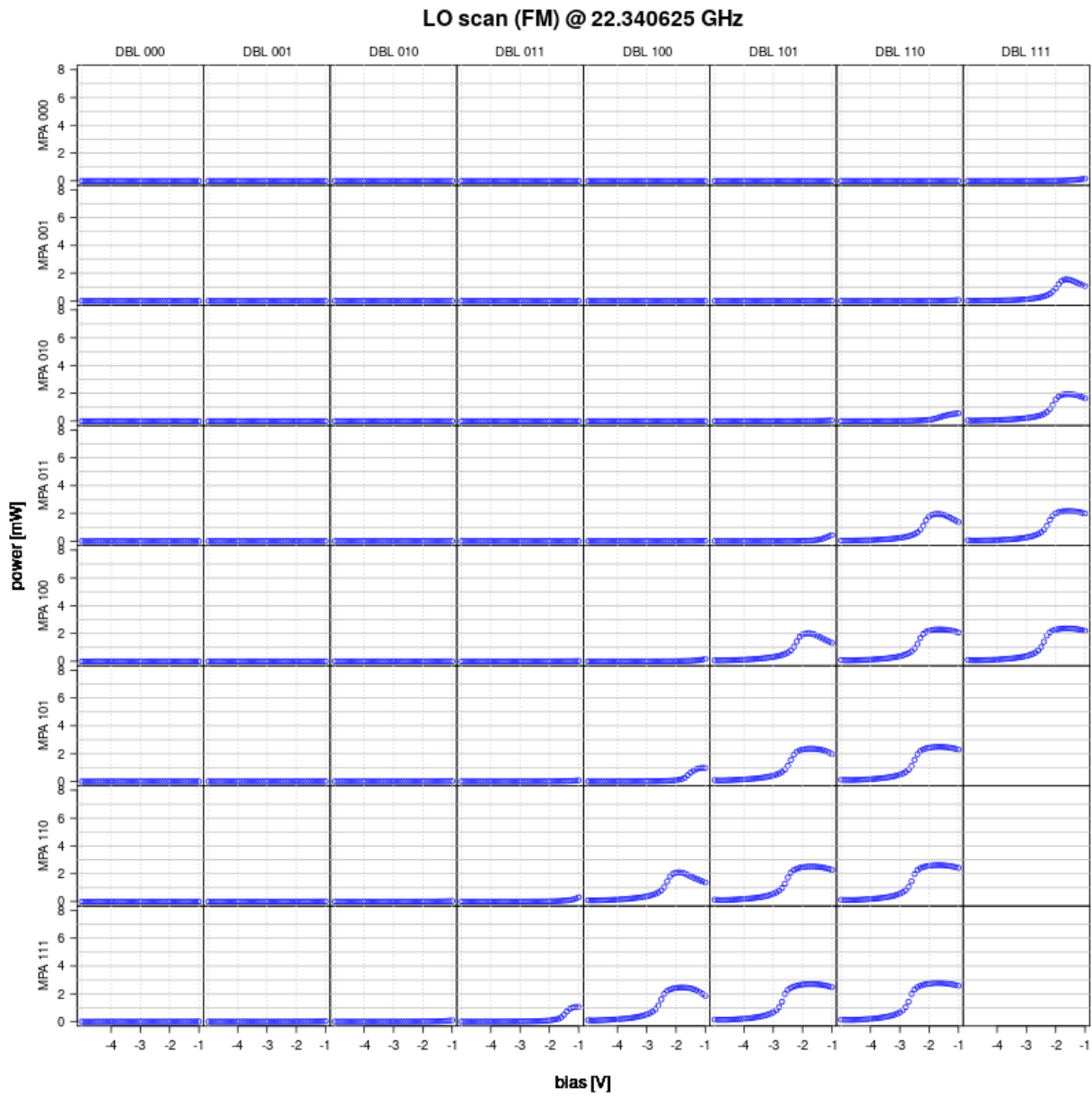
23.236875	470	1188	235	121	129	168	149	0	0	0	0	0	0
23.244375	559	1061	222	144	135	163	157	19	0	0	0	0	0
23.268750	379	1099	208	166	139	135	129	0	0	0	0	0	0
23.306250	318	865	300	161	163	136	101	111	18	0	0	0	0
23.336250	47	1056	267	181	160	155	122	100	80	5	0	0	0
23.364375	7	1221	246	121	126	89	68	54	36	0	0	0	0
23.392500	118	1165	136	114	146	149	114	26	0	0	0	0	0
23.415000	328	887	134	143	178	208	90	0	0	0	0	0	0
23.416875	312	892	142	148	178	211	85	0	0	0	0	0	0
23.418750	348	851	149	149	175	209	87	0	0	0	0	0	0
23.420625	268	921	157	147	185	207	83	0	0	0	0	0	0
23.422500	306	876	165	150	185	211	75	0	0	0	0	0	0
23.445000	25	1096	193	170	166	190	128	0	0	0	0	0	0
23.452500	27	1104	191	163	165	186	132	0	0	0	0	0	0
23.454375	25	1105	197	160	168	182	131	0	0	0	0	0	0
23.456250	263	886	191	162	162	189	115	0	0	0	0	0	0
23.458125	84	1051	196	162	165	180	130	0	0	0	0	0	0
23.460000	360	776	197	159	166	183	127	0	0	0	0	0	0
23.467500	95	1036	203	151	162	178	143	0	0	0	0	0	0
23.480625	198	941	182	150	163	195	139	0	0	0	0	0	0
23.505000	126	1055	193	173	214	265	106	0	0	0	0	0	0
23.546250	81	967	239	243	260	217	125	0	0	0	0	0	0
23.576250	23	815	348	280	243	182	153	88	0	0	0	0	0
23.593125	25	653	388	306	252	191	151	135	31	0	0	0	0
23.600625	5	613	398	325	261	196	146	132	56	0	0	0	0
23.602500	3	608	394	327	270	186	152	131	61	0	0	0	0
23.604375	4	608	395	325	282	179	148	133	58	0	0	0	0
23.606250	3	606	393	325	281	187	143	129	65	0	0	0	0
23.608125	3	606	396	319	291	182	145	126	64	0	0	0	0
23.615625	3	636	413	312	271	189	131	115	62	0	0	0	0
23.634375	0	736	486	300	232	149	88	95	46	0	0	0	0
23.651250	0	1176	417	220	139	100	71	9	0	0	0	0	0
23.658750	0	1394	276	178	126	90	58	10	0	0	0	0	0
23.660625	0	1413	258	174	129	90	57	11	0	0	0	0	0
23.662500	0	1442	230	176	125	92	58	9	0	0	0	0	0
23.664375	0	1457	212	173	131	95	56	8	0	0	0	0	0
23.666250	0	1471	195	175	132	94	57	8	0	0	0	0	0
23.673750	0	1513	160	159	138	100	55	7	0	0	0	0	0
23.698125	12	1523	172	132	151	106	36	0	0	0	0	0	0
23.730000	41	1548	245	136	135	154	37	0	0	0	0	0	0
23.731875	252	1342	245	134	131	154	38	0	0	0	0	0	0
23.733750	99	1491	251	130	126	156	43	0	0	0	0	0	0
23.735625	161	1430	256	125	119	156	49	0	0	0	0	0	0
23.737500	374	1215	258	123	121	153	52	0	0	0	0	0	0
23.750625	216	1366	250	127	116	149	72	0	0	0	0	0	0
23.758125	266	1341	223	127	124	143	72	0	0	0	0	0	0
23.760000	355	1261	214	127	119	147	73	0	0	0	0	0	0
23.761875	164	1460	205	127	120	147	73	0	0	0	0	0	0
23.763750	386	1253	187	131	115	153	71	0	0	0	0	0	0
23.765625	375	1277	173	130	117	153	71	0	0	0	0	0	0
23.773125	354	1310	154	136	117	158	67	0	0	0	0	0	0
23.803125	301	1301	175	166	131	156	66	0	0	0	0	0	0
23.842500	293	1202	250	159	155	142	95	0	0	0	0	0	0

23.883750	227	1151	298	196	144	125	109	79	8	0	0	0	0
23.919375	4	1424	310	236	173	143	101	77	33	0	0	0	0
23.994375	15	2110	133	139	116	29	0	0	0	0	0	0	0
24.031875	16	2146	103	91	128	58	0	0	0	0	0	0	0
24.050625	1009	1145	100	90	118	80	0	0	0	0	0	0	0
24.105000	999	1044	146	110	133	110	0	0	0	0	0	0	0
24.138750	35	1906	184	94	122	130	71	0	0	0	0	0	0
24.172500	298	1521	218	118	133	114	118	22	0	0	0	0	0
24.223125	9	1653	178	207	148	122	96	80	8	0	0	0	0
24.230625	5	1693	172	188	158	106	96	73	10	0	0	0	0
24.232500	5	1699	176	182	158	105	100	66	10	0	0	0	0
24.234375	17	1702	173	174	156	104	104	65	6	0	0	0	0
24.236250	9	1722	167	172	156	106	102	64	3	0	0	0	0
24.238125	3	1738	173	176	139	110	103	59	0	0	0	0	0
24.245625	4	1754	156	180	147	117	89	54	0	0	0	0	0
24.251250	4	1767	154	137	116	76	63	20	0	0	0	0	0
24.260625	3	1804	120	140	116	80	62	12	0	0	0	0	0
24.322500	1	1550	126	165	210	146	16	0	0	0	0	0	0
24.343125	49	1414	185	158	173	199	36	0	0	0	0	0	0
24.382500	58	1310	172	137	138	142	52	0	0	0	0	0	0
24.408750	9	1255	186	157	136	159	107	0	0	0	0	0	0
24.444375	217	802	257	145	165	177	179	67	0	0	0	0	0
24.487500	102	553	244	179	198	178	146	153	133	0	0	0	0
24.532500	6	532	182	238	204	154	138	113	87	27	0	0	0
24.551250	2	521	181	206	179	147	132	89	74	27	0	0	0
24.708750	2	396	172	139	151	162	146	103	0	0	0	0	0
24.733125	4	350	175	159	160	144	148	131	0	0	0	0	0
24.735000	4	351	172	157	161	145	139	136	6	0	0	0	0
24.742500	4	329	176	164	153	143	132	138	32	0	0	0	0
24.744375	5	325	174	164	155	142	131	140	35	0	0	0	0
24.746250	4	321	176	160	156	141	130	142	41	0	0	0	0
24.748125	3	312	179	161	152	141	133	137	53	0	0	0	0
24.750000	3	303	185	160	154	138	135	131	62	0	0	0	0
24.757500	2	281	184	165	141	151	127	128	92	0	0	0	0
24.787500	0	212	183	164	157	150	114	104	109	78	0	0	0
24.823125	0	190	213	178	171	119	119	89	110	93	30	0	0
24.845625	0	385	293	188	173	129	125	102	91	88	25	0	0
24.909375	0	978	218	126	166	176	137	100	67	0	0	0	0
24.928125	2	1134	219	168	164	203	133	68	0	0	0	0	0
24.950625	3	1205	238	159	164	189	144	30	0	0	0	0	0
25.051875	4	1431	297	206	171	130	117	81	23	0	0	0	0
25.074375	129	1260	325	218	174	134	111	86	23	0	0	0	0
25.147500	2	1631	290	156	168	93	83	50	28	0	0	0	0
25.220625	179	1634	104	97	128	146	121	71	21	0	0	0	0
25.246875	388	1358	108	135	130	164	142	76	0	0	0	0	0
25.263750	50	1598	142	141	131	133	122	61	0	0	0	0	0
25.288125	17	1510	199	147	162	129	108	88	18	0	0	0	0
25.344375	19	1167	285	246	142	130	120	96	72	19	0	0	0
25.387500	8	1169	278	174	185	123	100	69	50	17	0	0	0
25.428750	24	1149	284	141	153	158	100	70	63	27	4	0	0
25.470000	35	1149	191	132	125	158	100	92	58	42	9	0	0
25.511250	9	996	132	77	118	157	115	98	65	56	22	0	0
25.550625	81	631	113	131	154	170	166	118	109	77	13	0	0

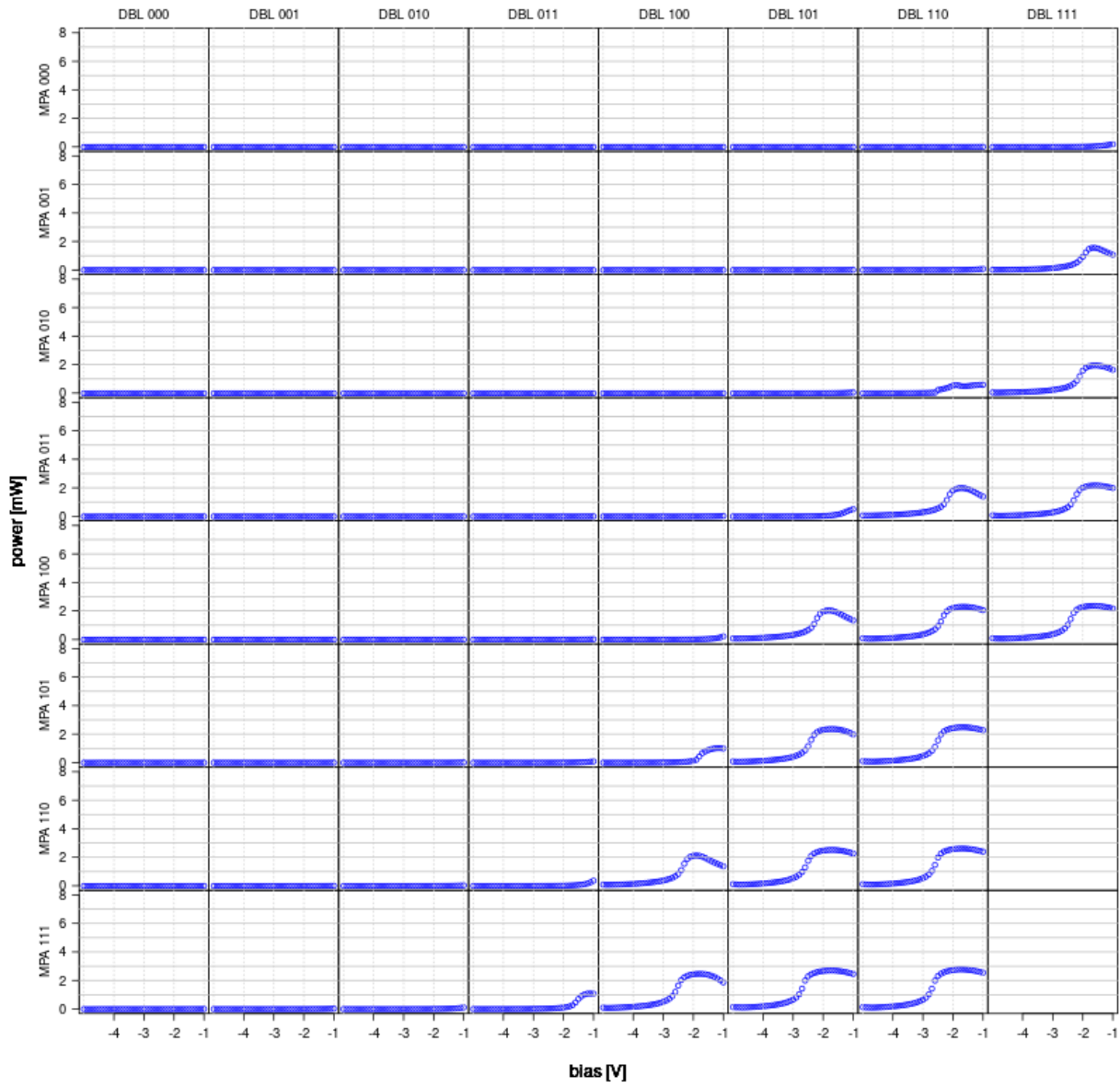
25.575000	18	536	141	121	199	184	165	158	122	102	17	0	0
25.595625	12	428	144	208	184	235	184	152	119	97	0	0	0
25.644375	11	291	181	165	200	197	179	149	126	105	74	3	0
25.691250	2	204	179	173	159	189	168	158	127	109	86	45	0
25.726875	0	187	169	146	166	151	182	155	113	83	65	18	0
25.758750	0	160	176	147	150	162	160	125	112	76	60	25	0
25.790625	0	149	155	135	138	166	149	115	94	77	60	33	0
25.830000	0	137	173	156	139	184	163	117	120	79	86	71	10
25.854375	0	109	176	133	165	150	160	131	122	108	96	86	40
25.863750	0	110	179	120	179	155	156	135	116	119	101	85	21
25.933125	0	313	267	239	191	143	183	109	125	101	102	73	40
25.970625	0	379	280	233	182	198	141	146	112	94	84	52	26
26.034375	0	867	340	223	166	161	147	104	95	33	37	0	0
26.085000	3	1252	331	196	150	121	84	50	20	7	0	0	0
26.092500	6	1311	323	179	145	106	82	44	17	1	0	0	0
26.094375	20	1312	321	172	142	107	79	43	18	0	0	0	0
26.096250	5	1565	286	135	102	74	38	9	0	0	0	0	0
26.098125	24	1679	230	126	85	53	17	0	0	0	0	0	0
26.100000	5	1735	224	110	92	39	9	0	0	0	0	0	0
26.107500	120	1677	201	114	68	32	2	0	0	0	0	0	0
26.141250	85	1179	291	205	180	138	104	84	41	24	6	0	0
26.188125	79	1647	303	161	108	90	65	7	0	0	0	0	0
26.259375	106	1766	307	161	113	7	0	0	0	0	0	0	0
26.291250	247	1597	267	176	117	56	0	0	0	0	0	0	0
26.330625	346	2135	20	0	0	0	0	0	0	0	0	0	0

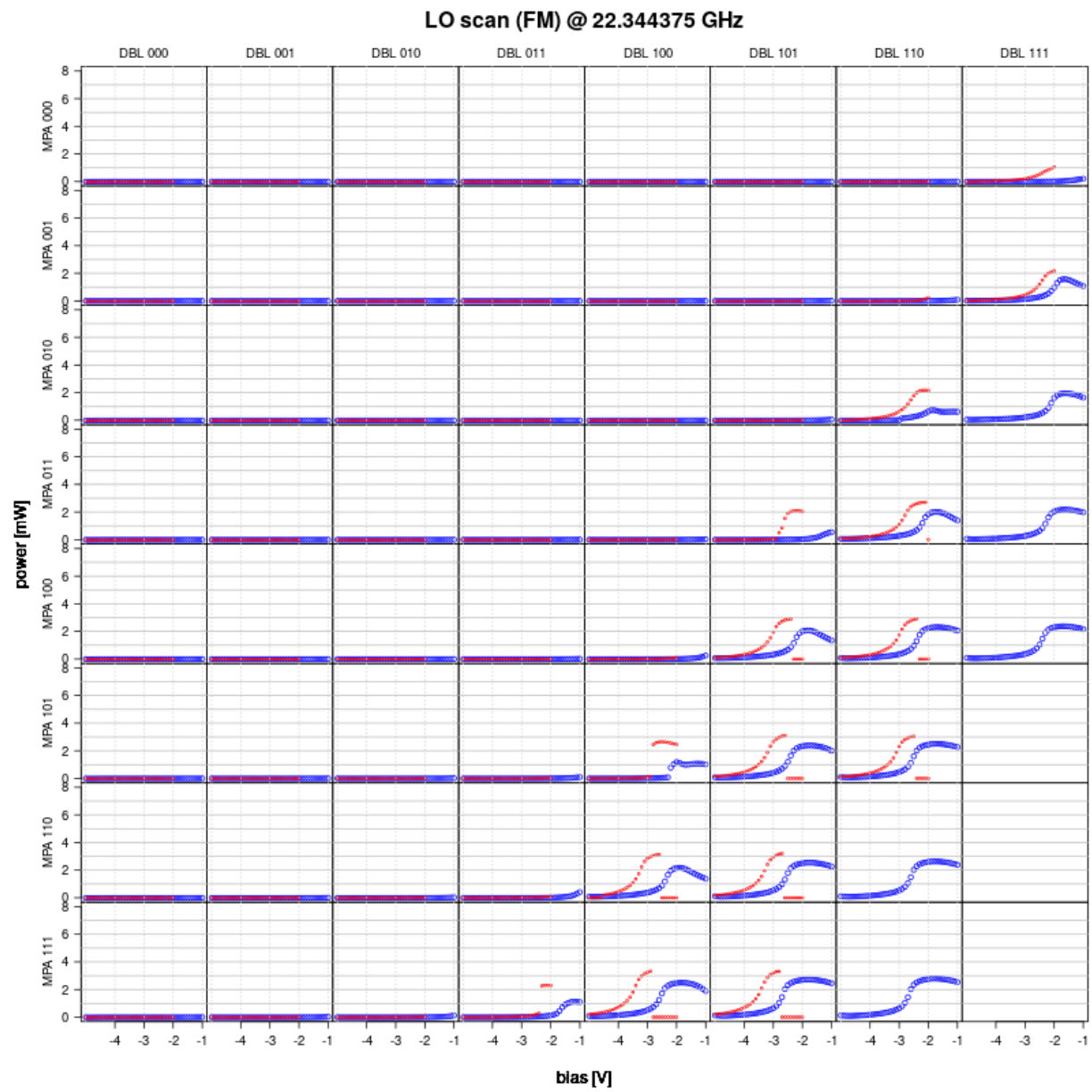


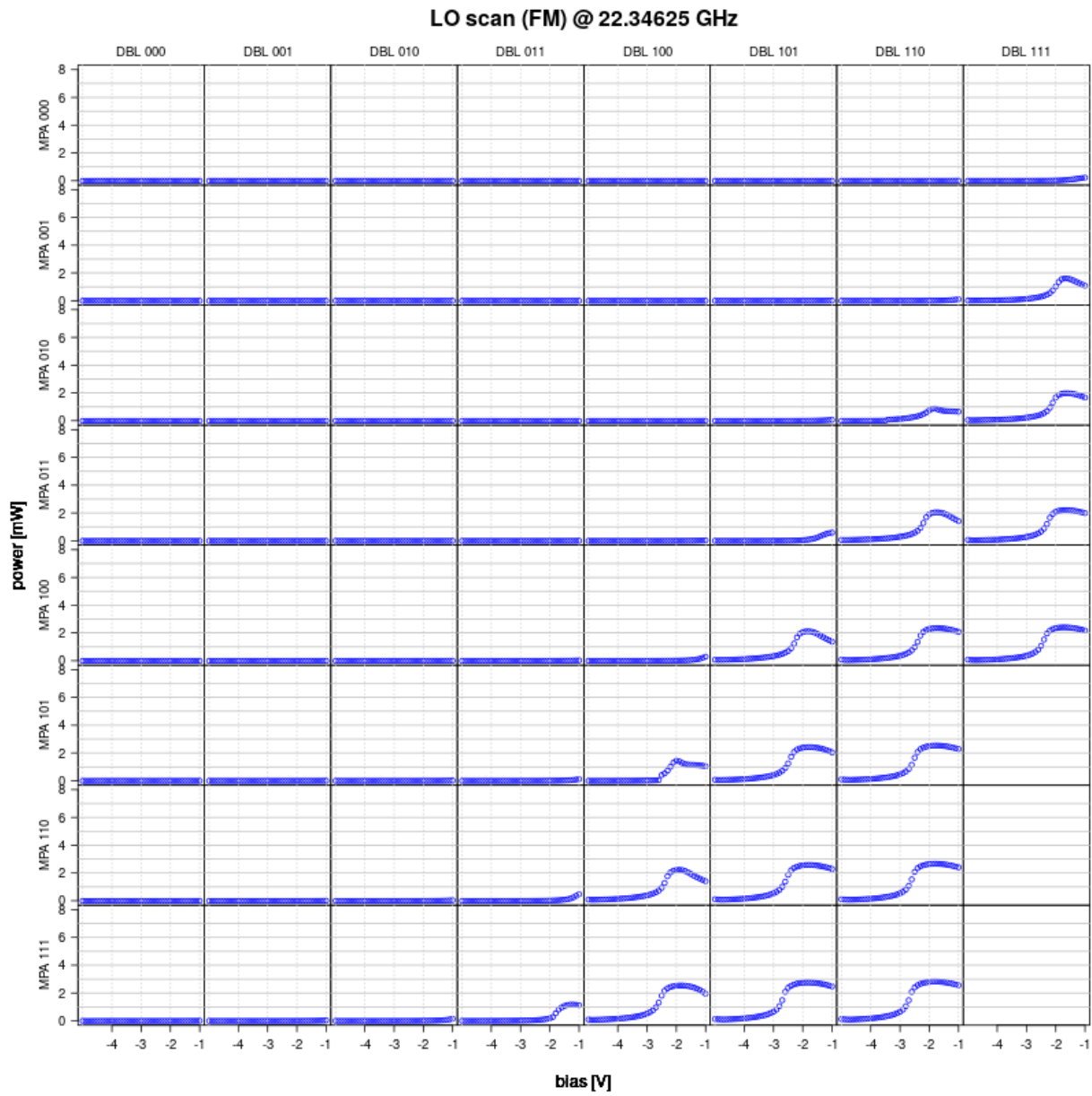
## B Results

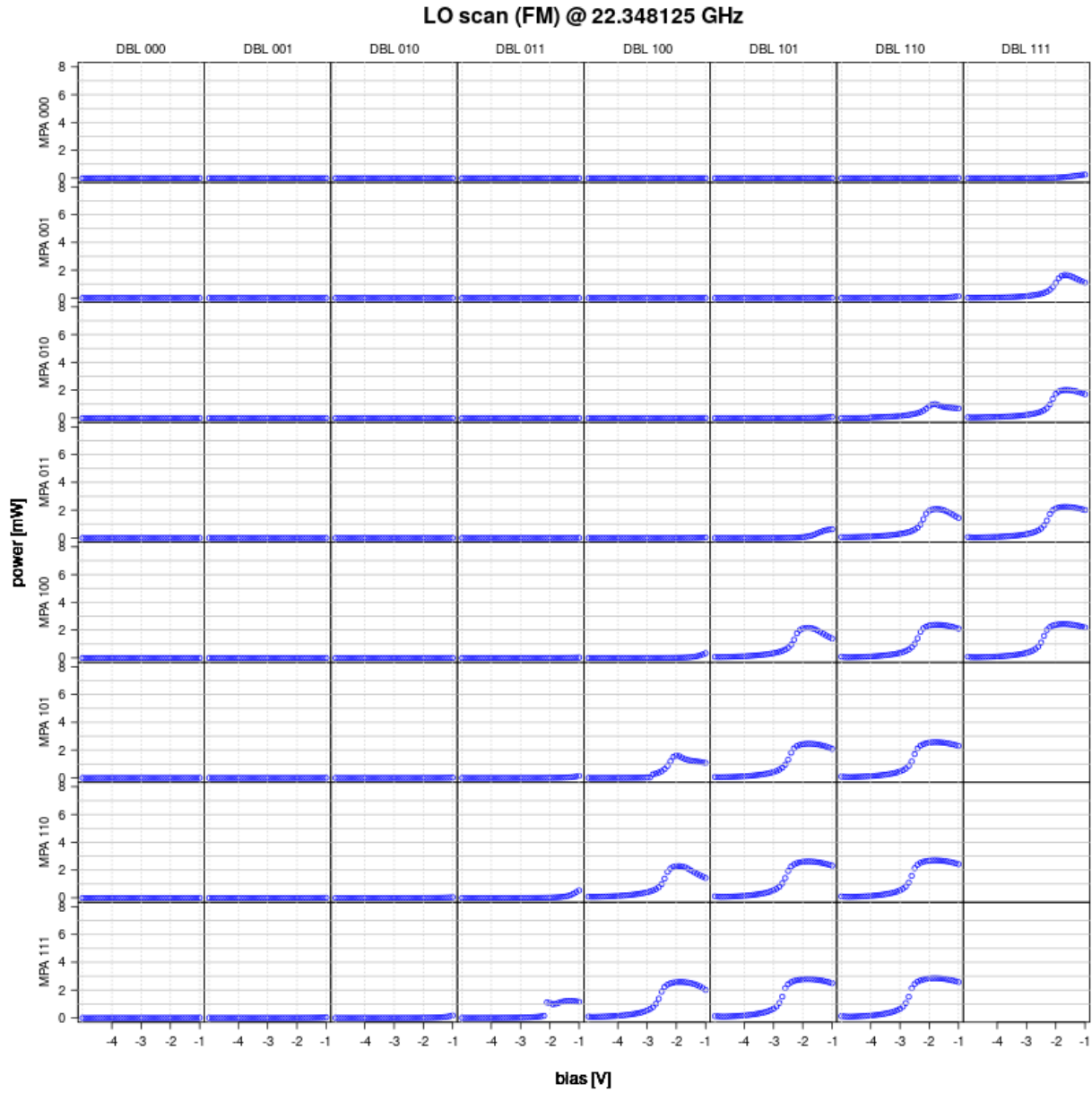


**LO scan (FM) @ 22.3425 GHz**

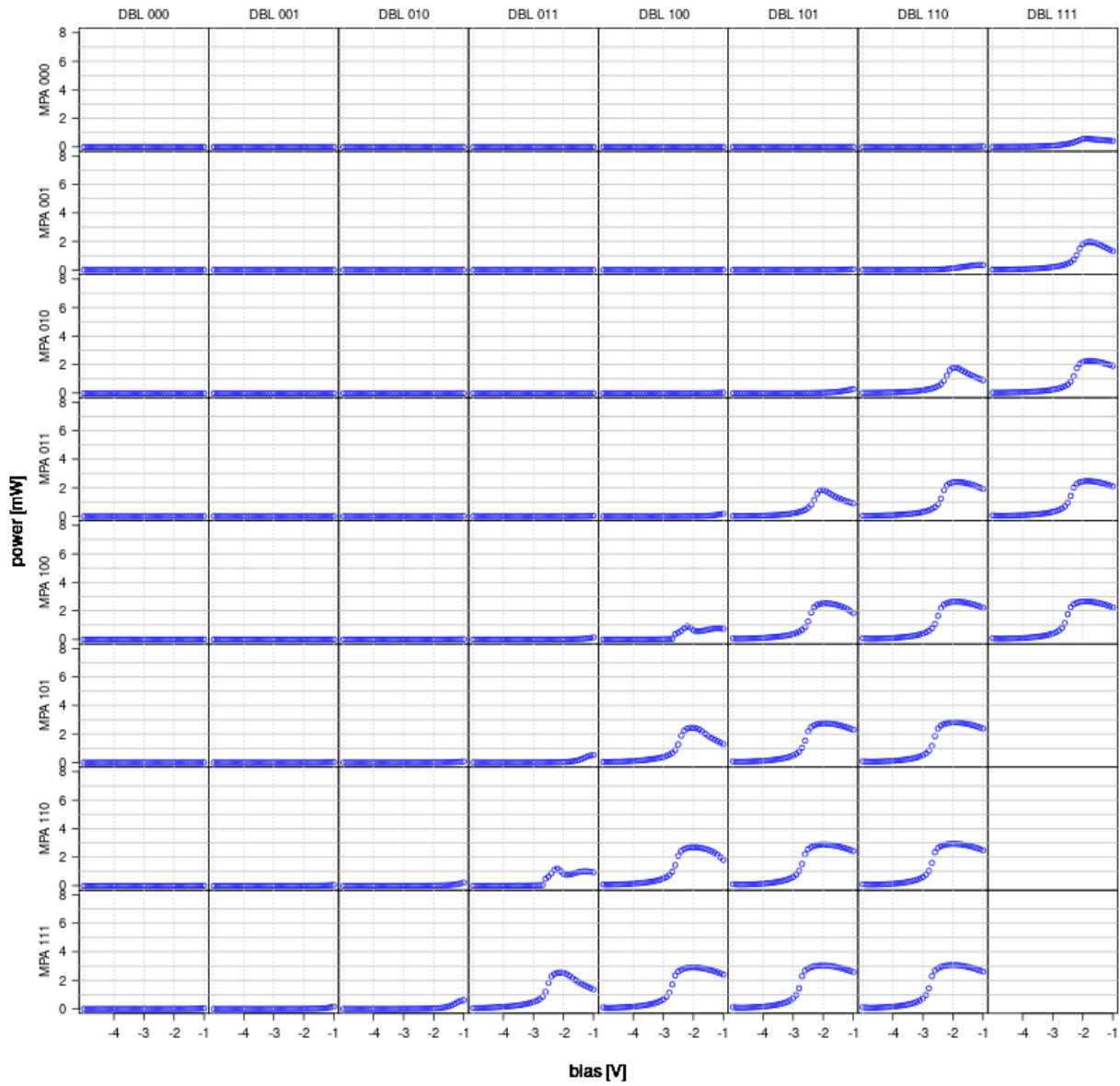


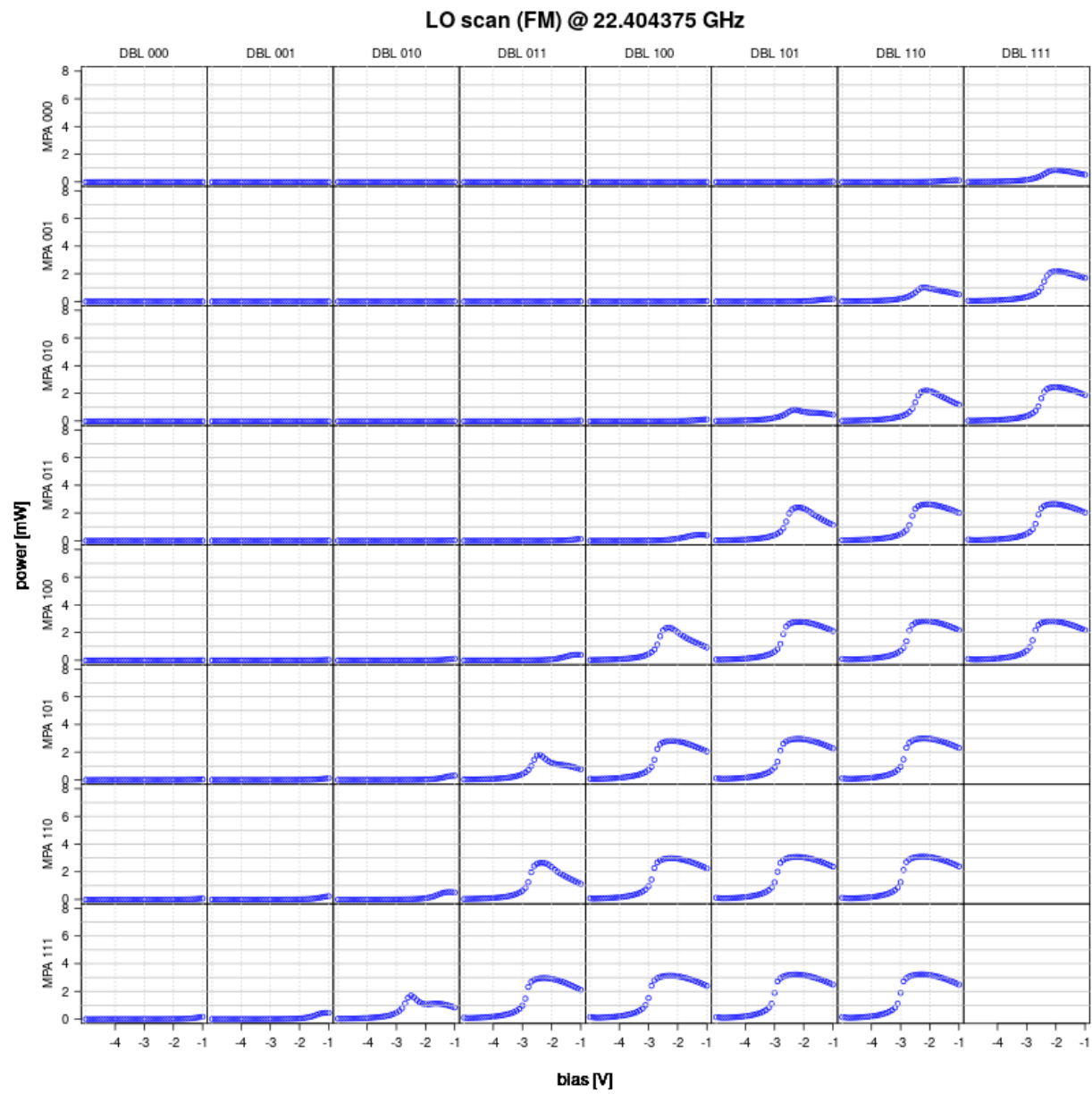




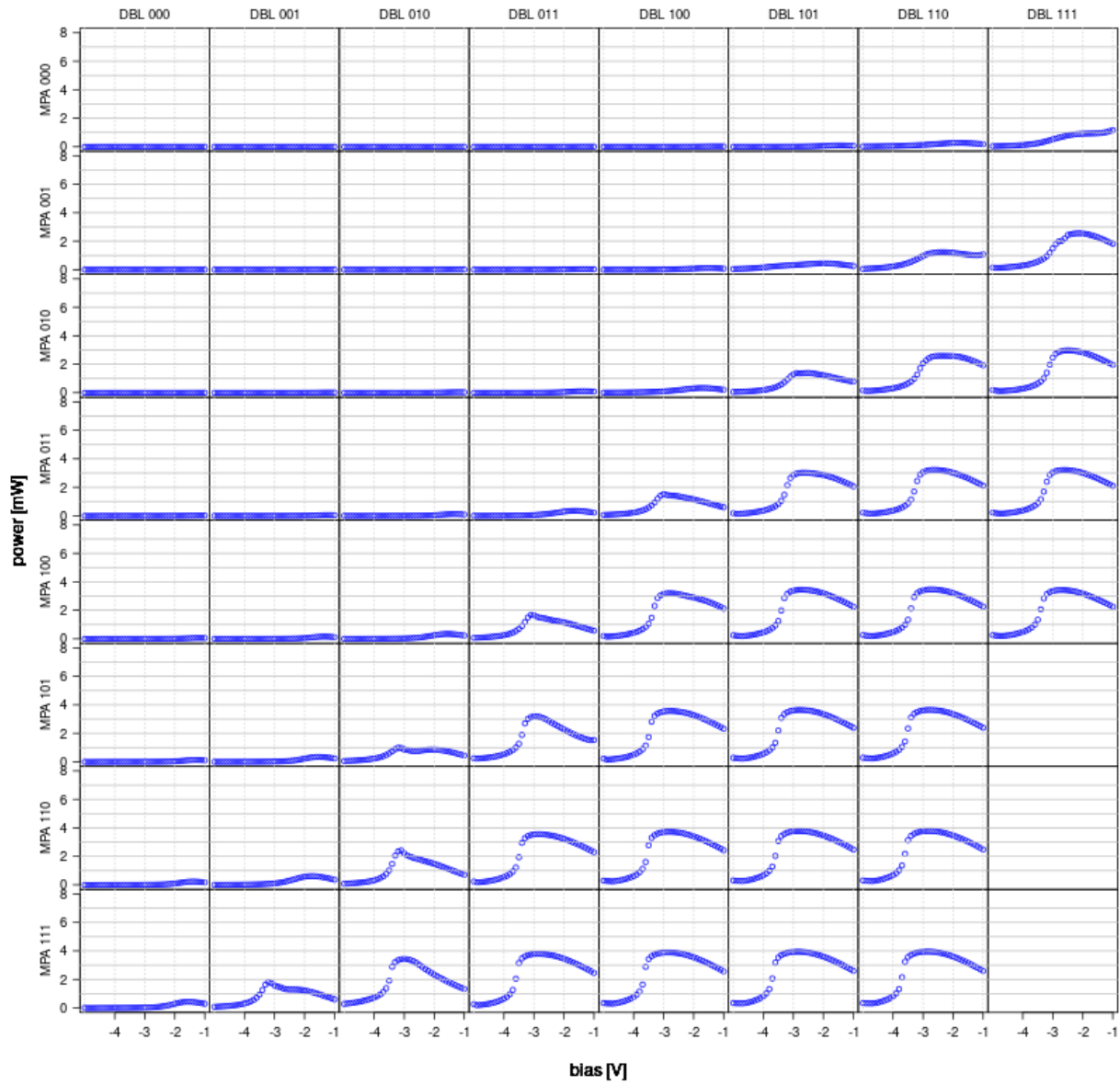


**LO scan (FM) @ 22.374375 GHz**

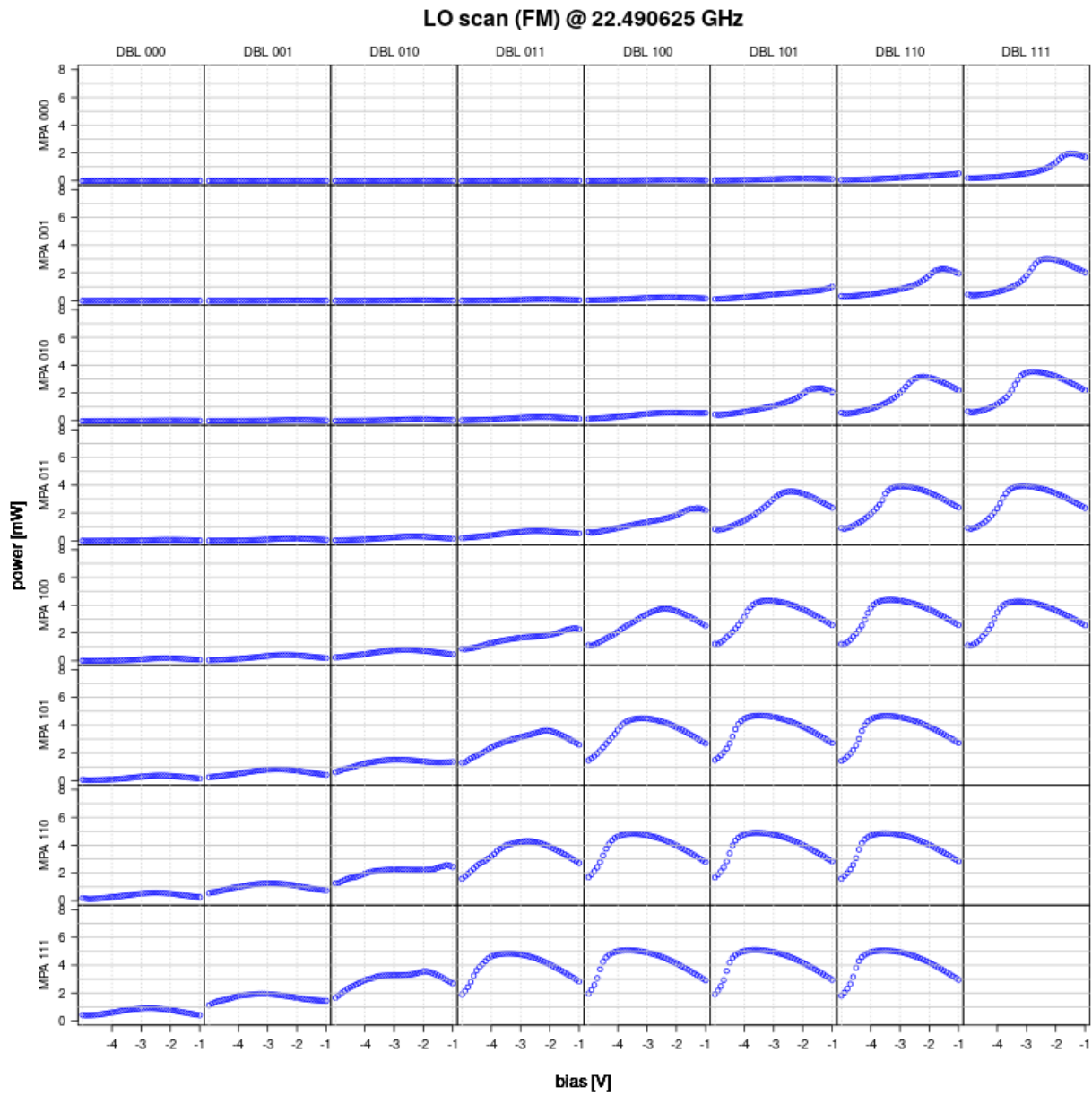




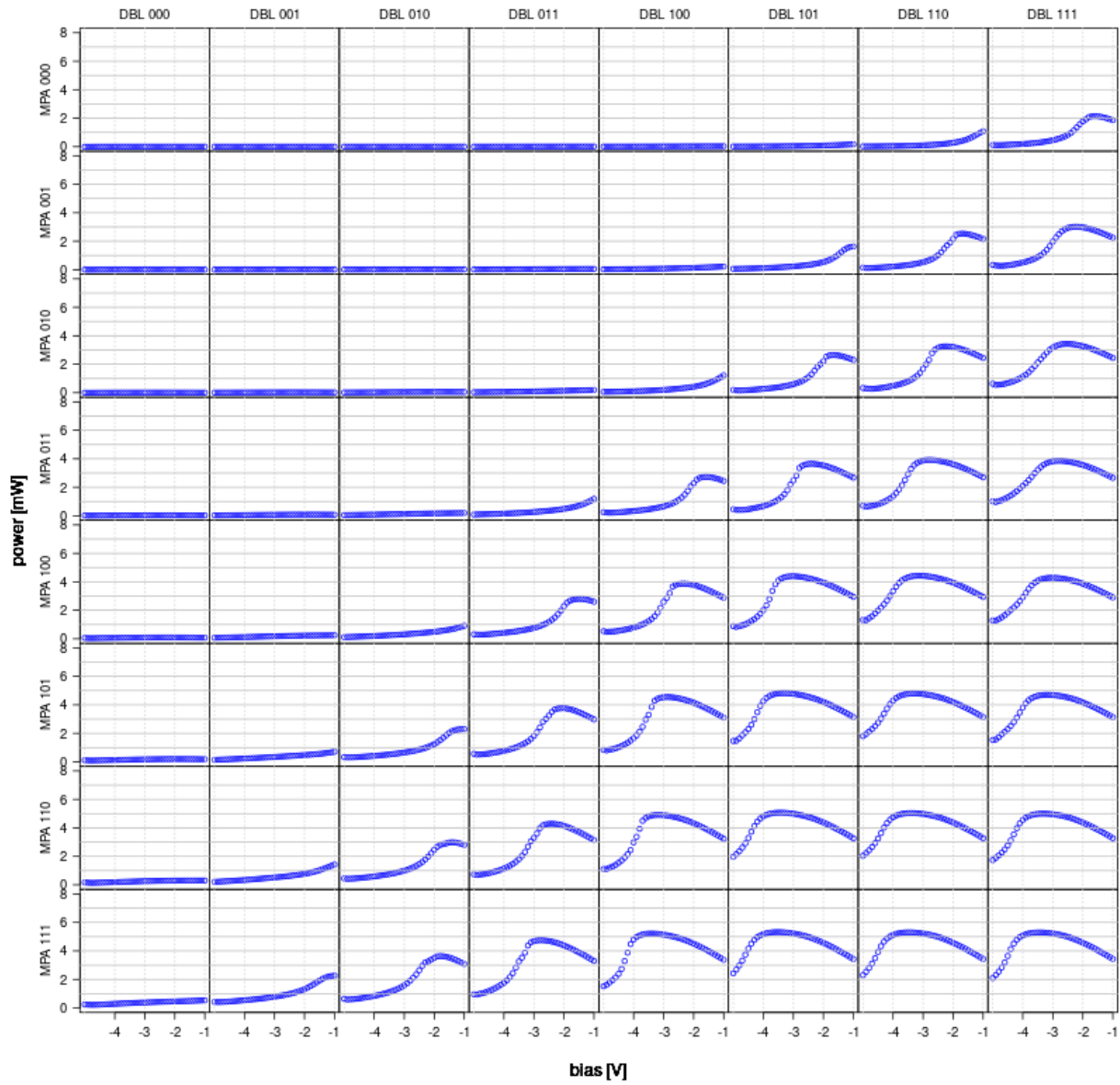
LO scan (FM) @ 22.4475 GHz



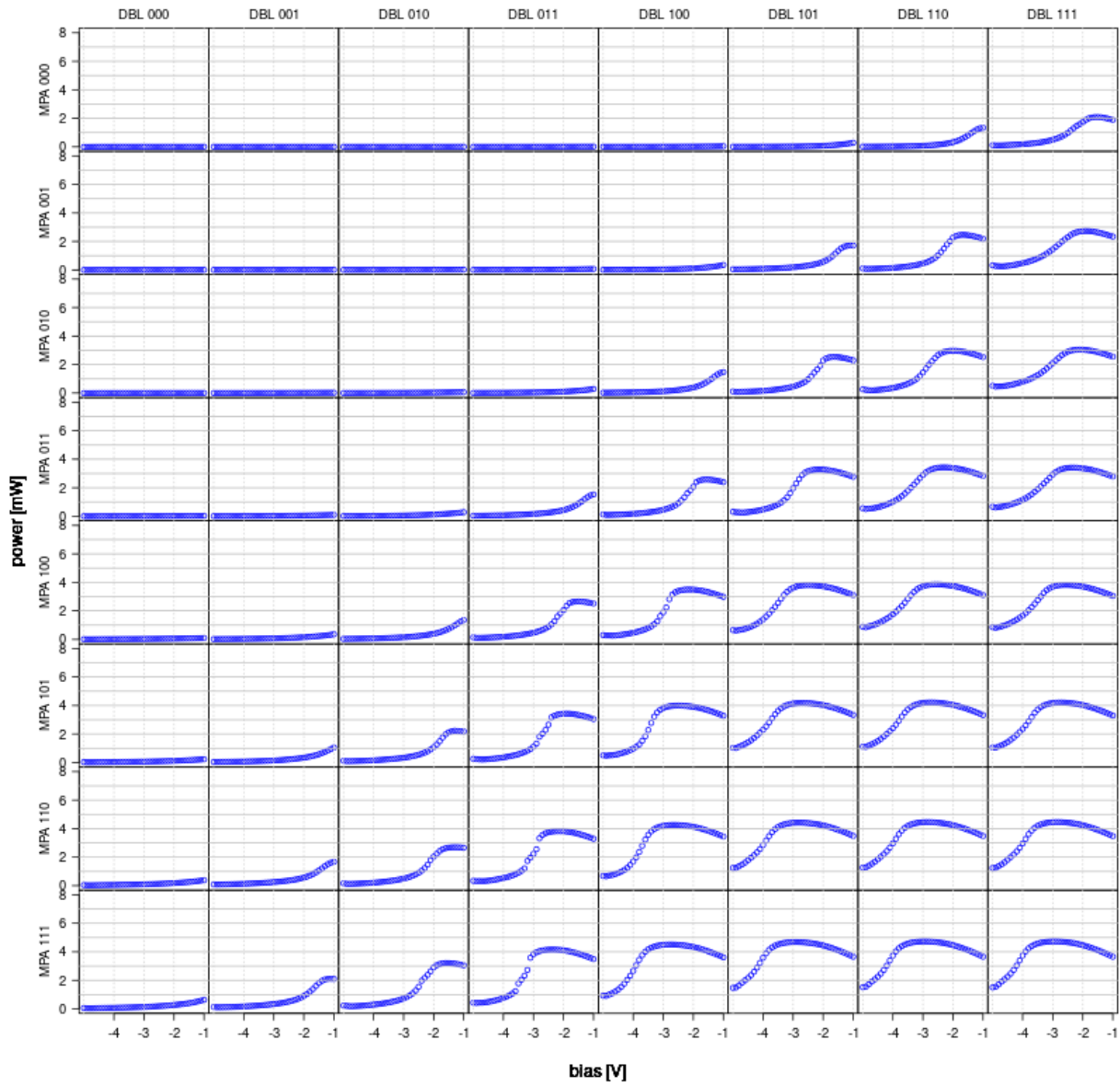




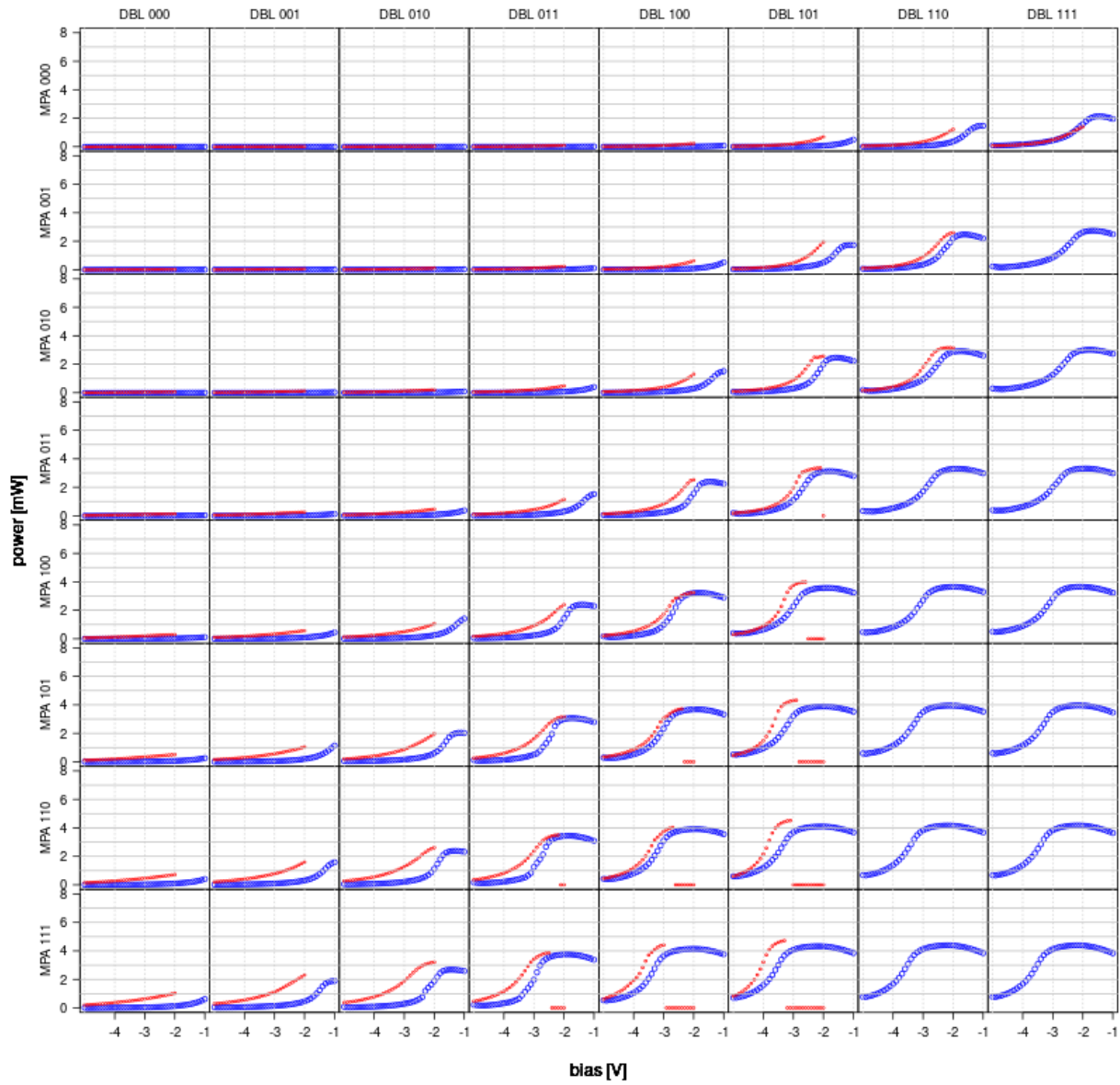
**LO scan (FM) @ 22.5225 GHz**

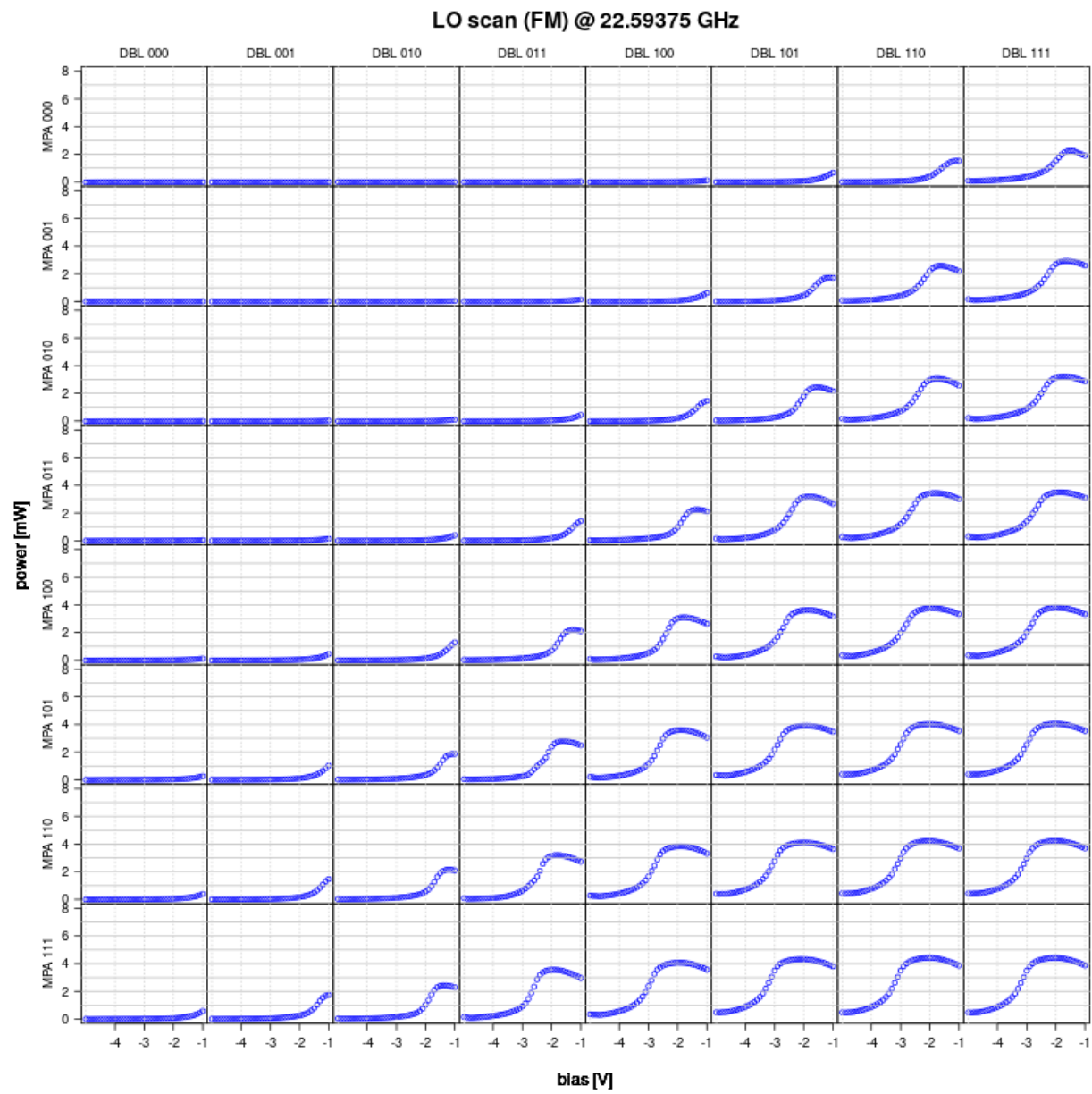


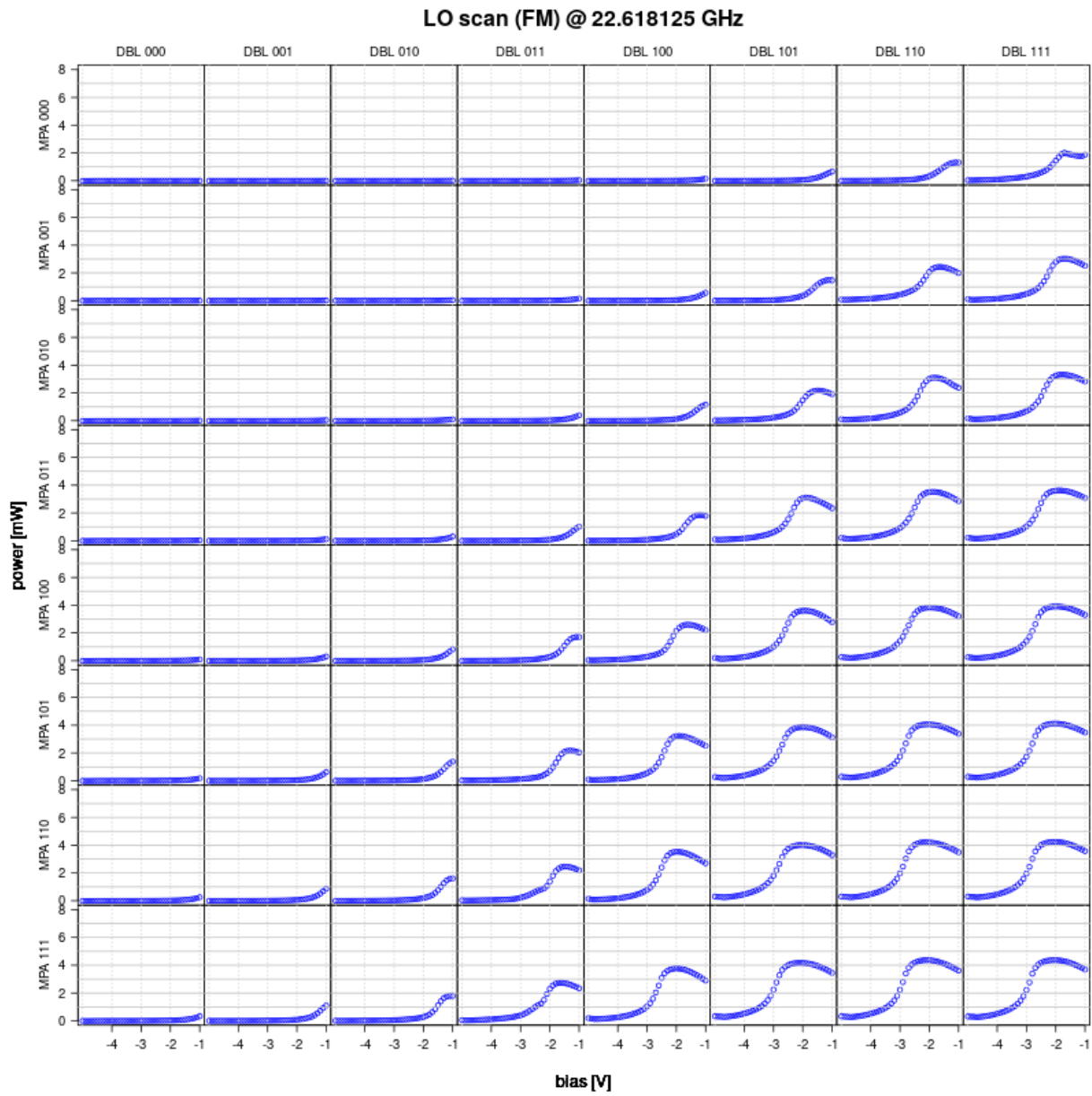
**LO scan (FM) @ 22.545 GHz**



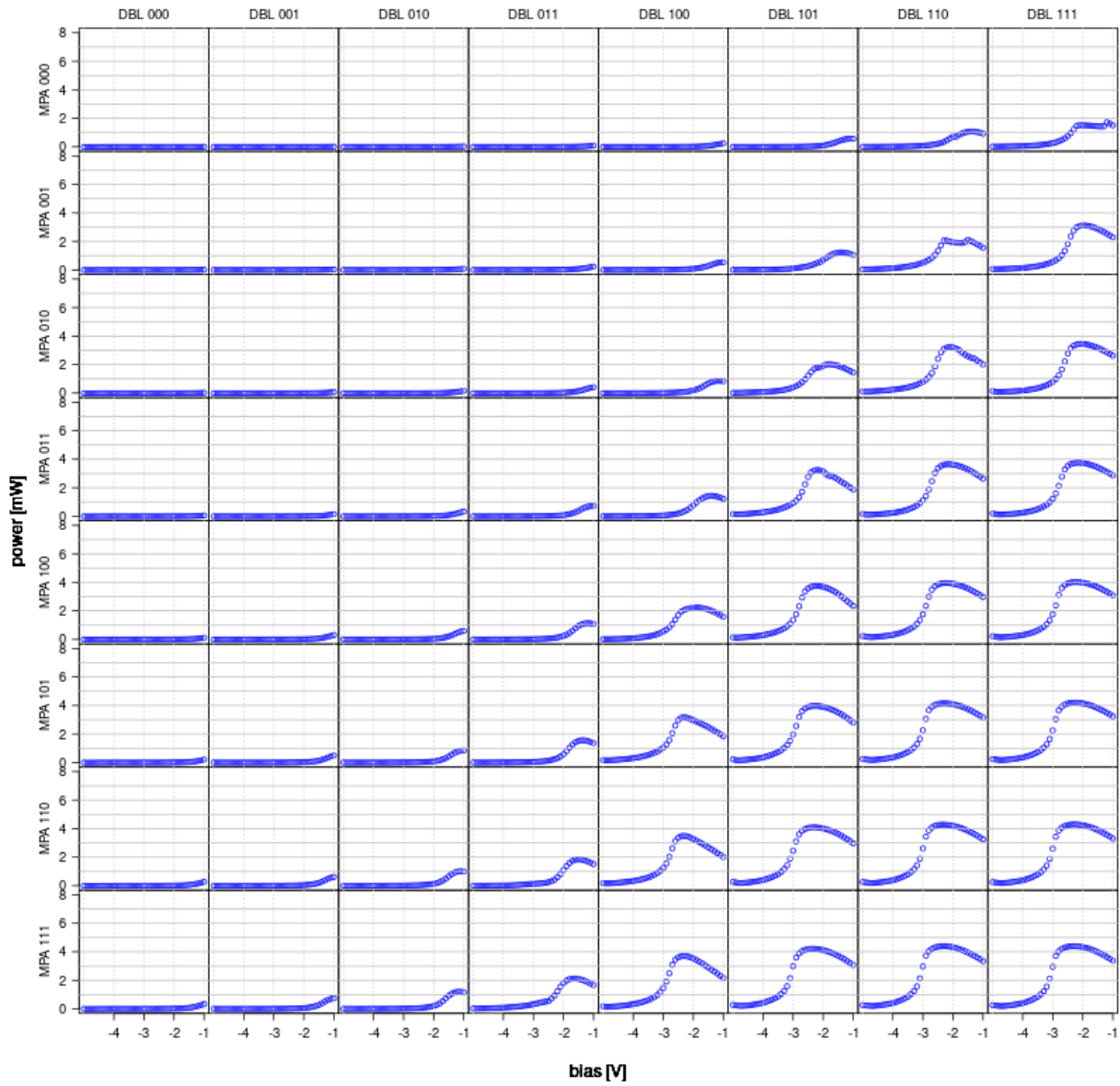
**LO scan (FM) @ 22.569375 GHz**

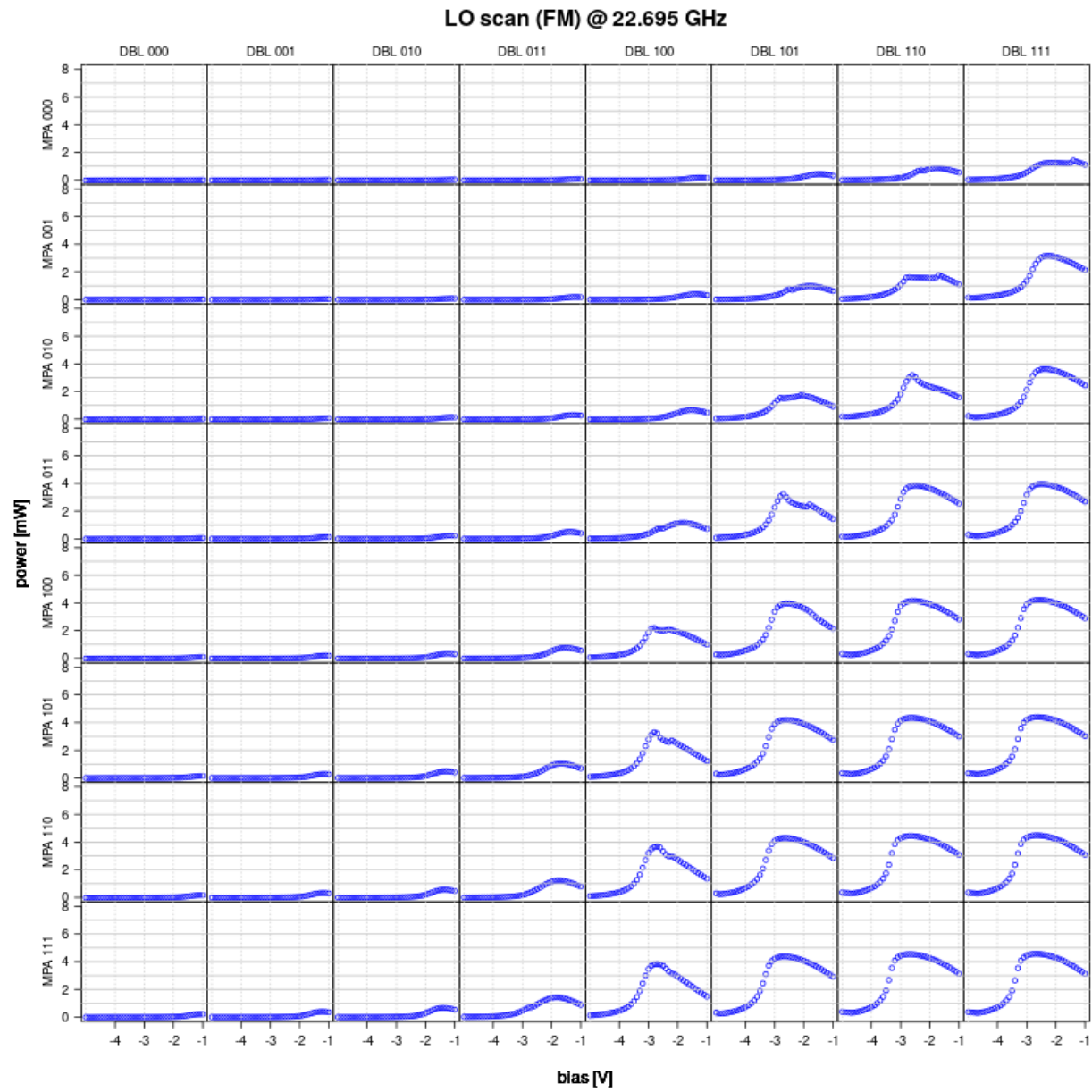






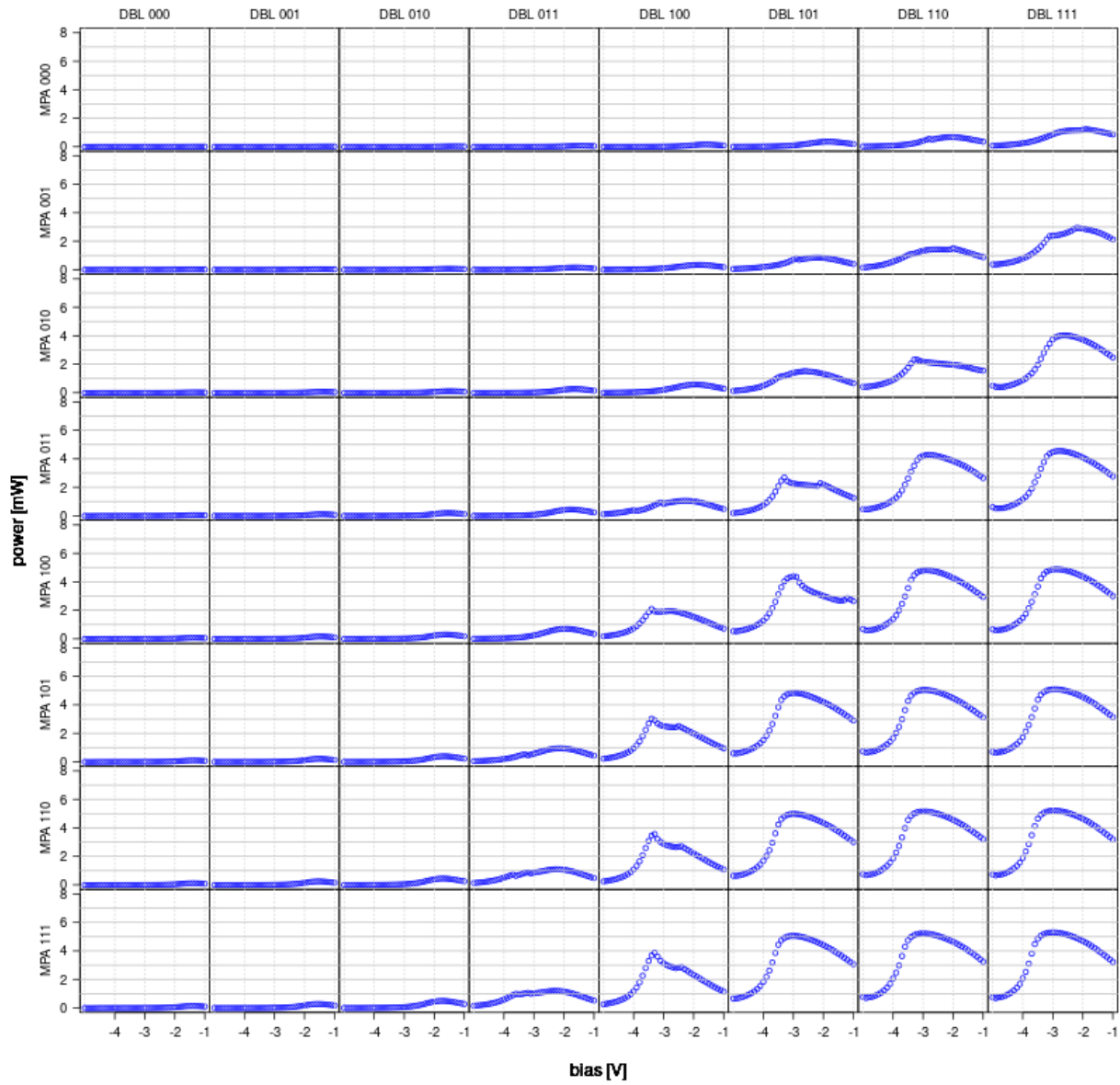
LO scan (FM) @ 22.6575 GHz

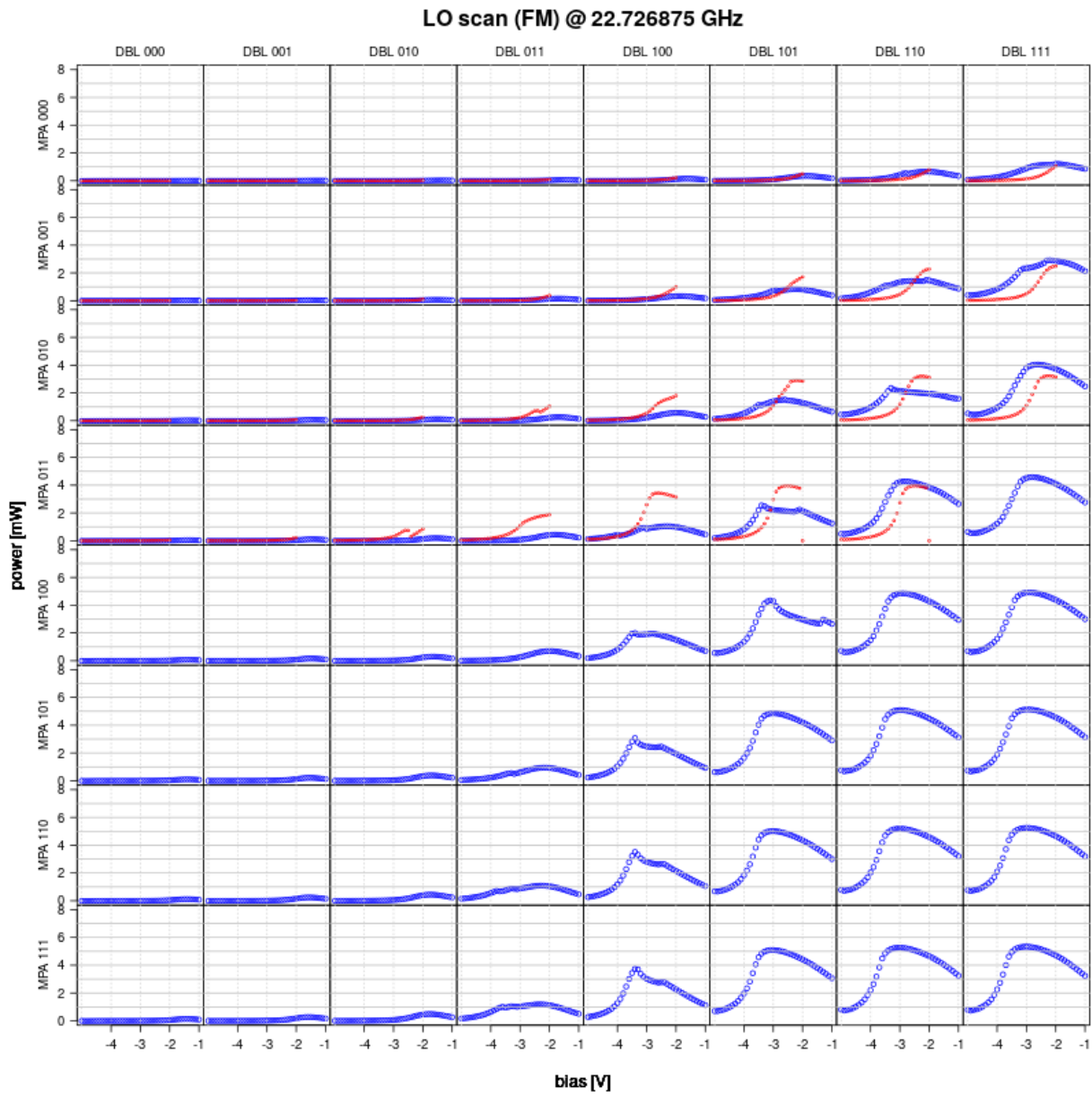


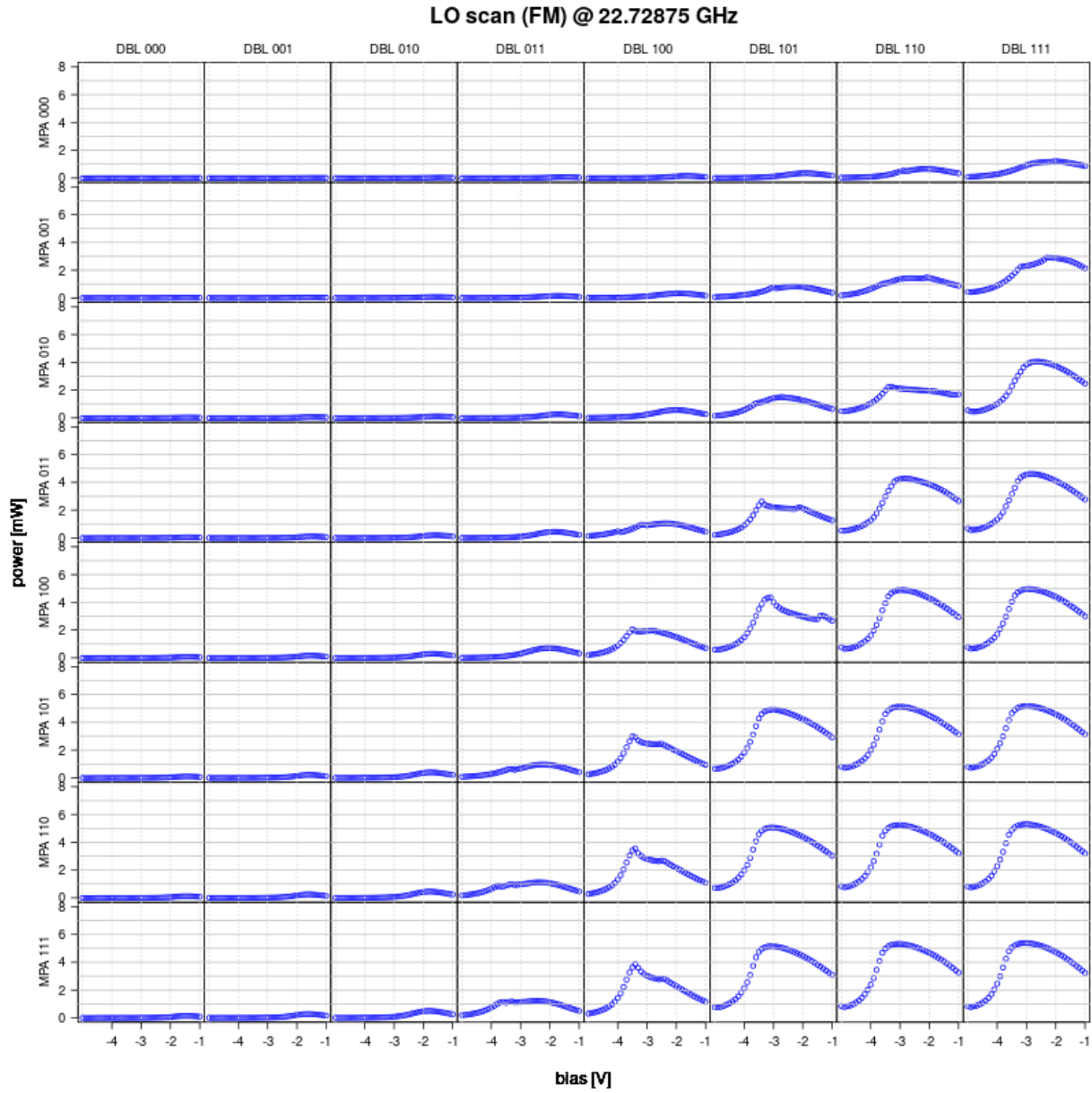


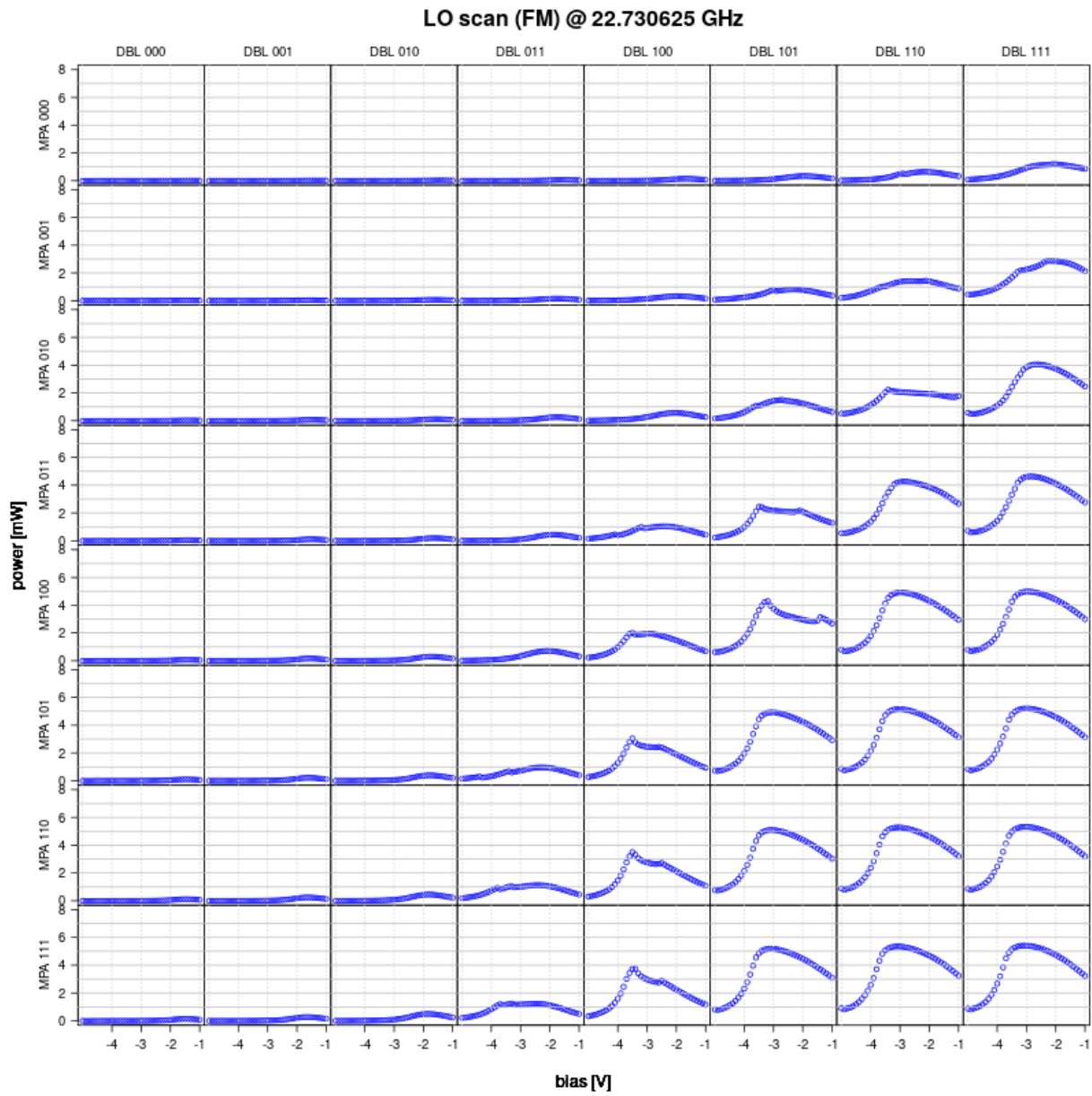


**LO scan (FM) @ 22.725 GHz**

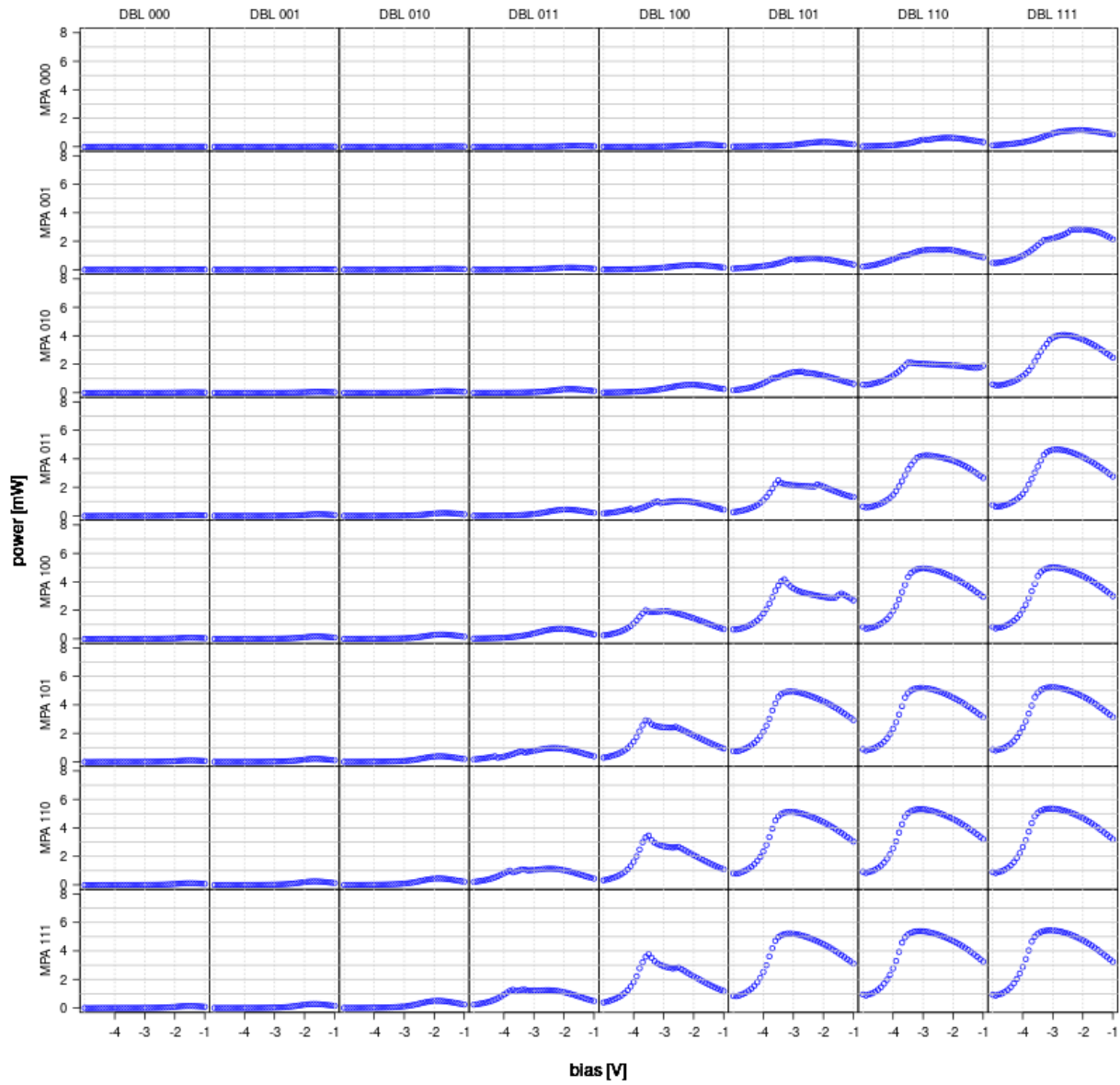




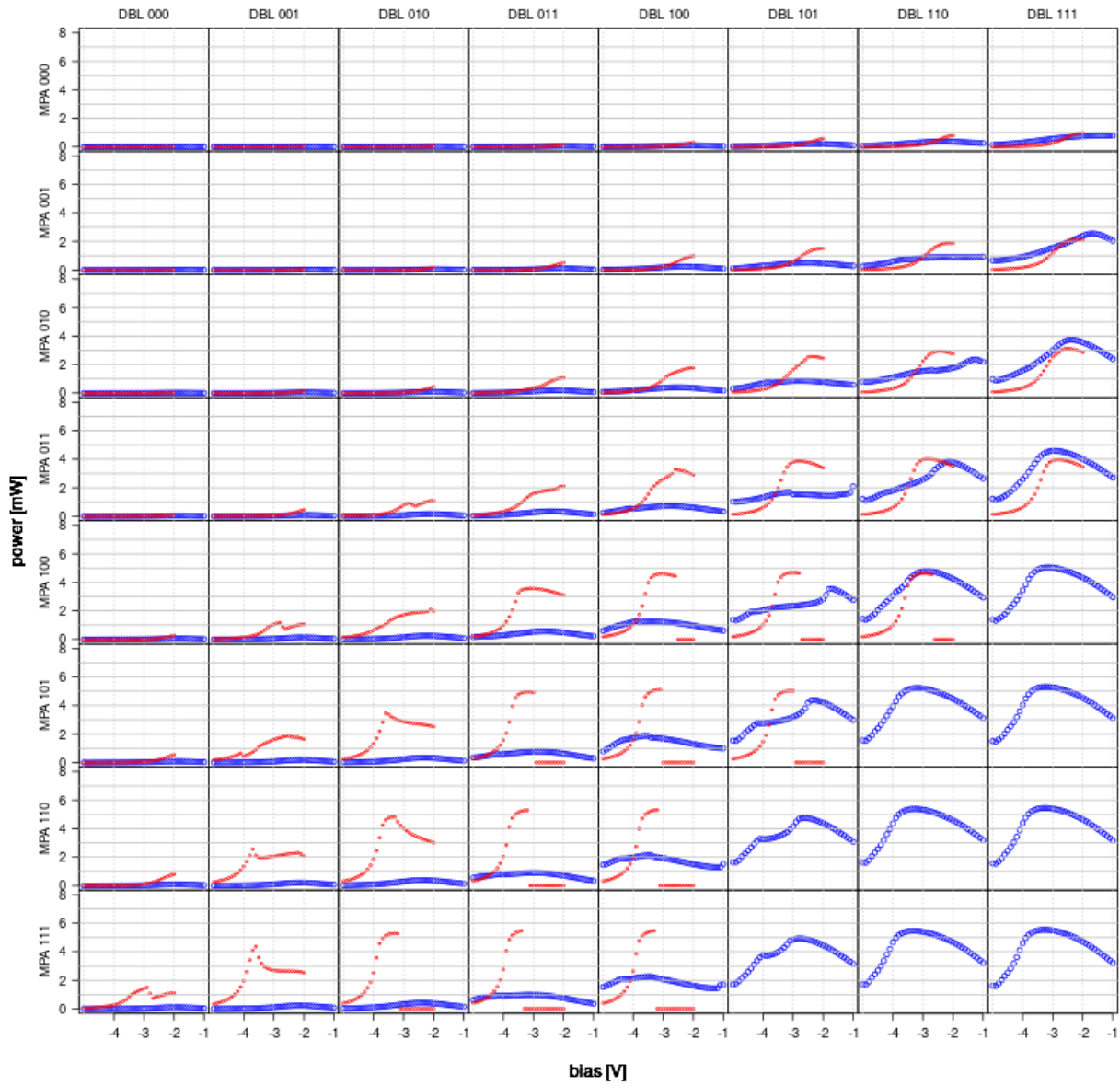




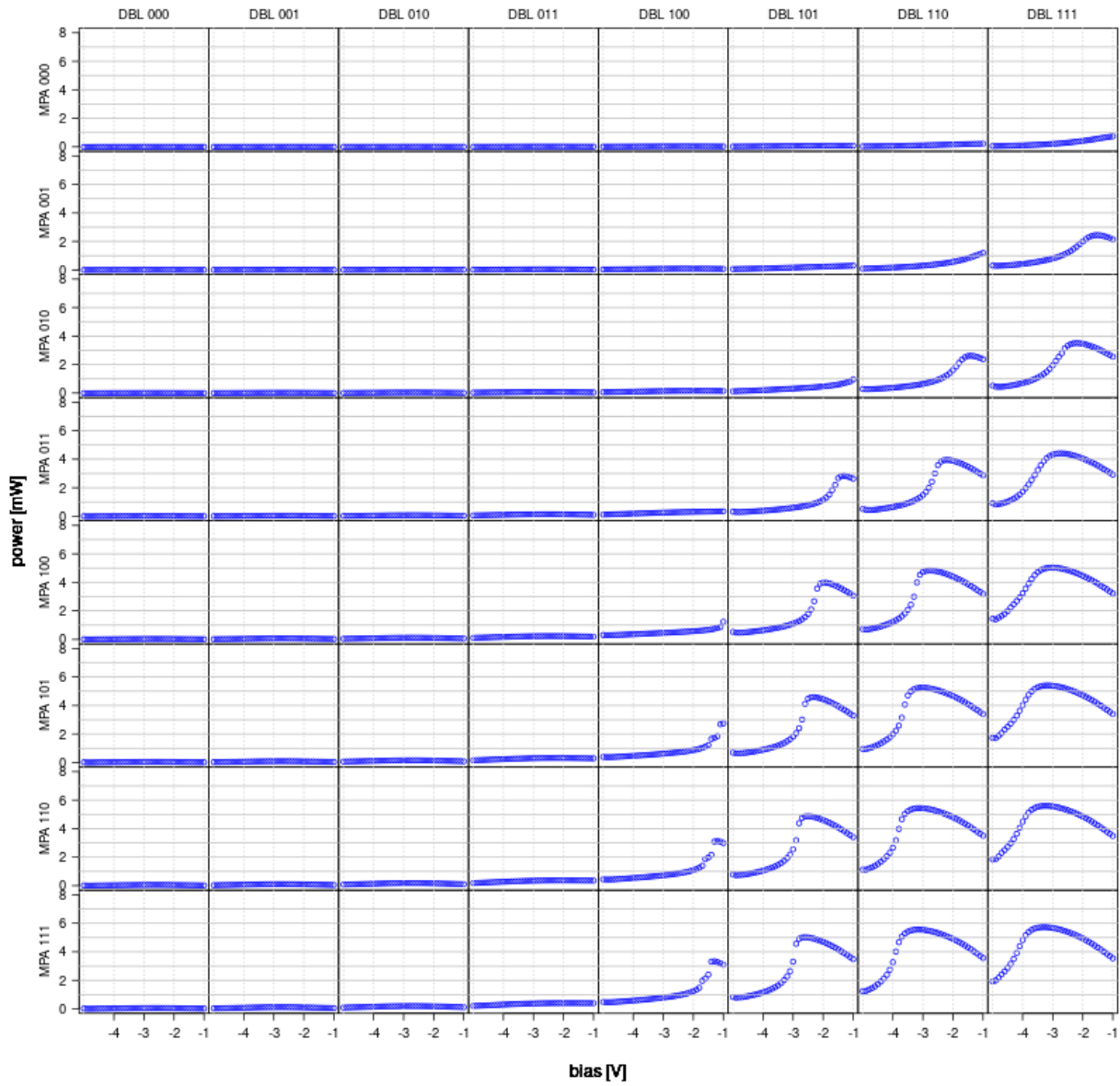
**LO scan (FM) @ 22.7325 GHz**



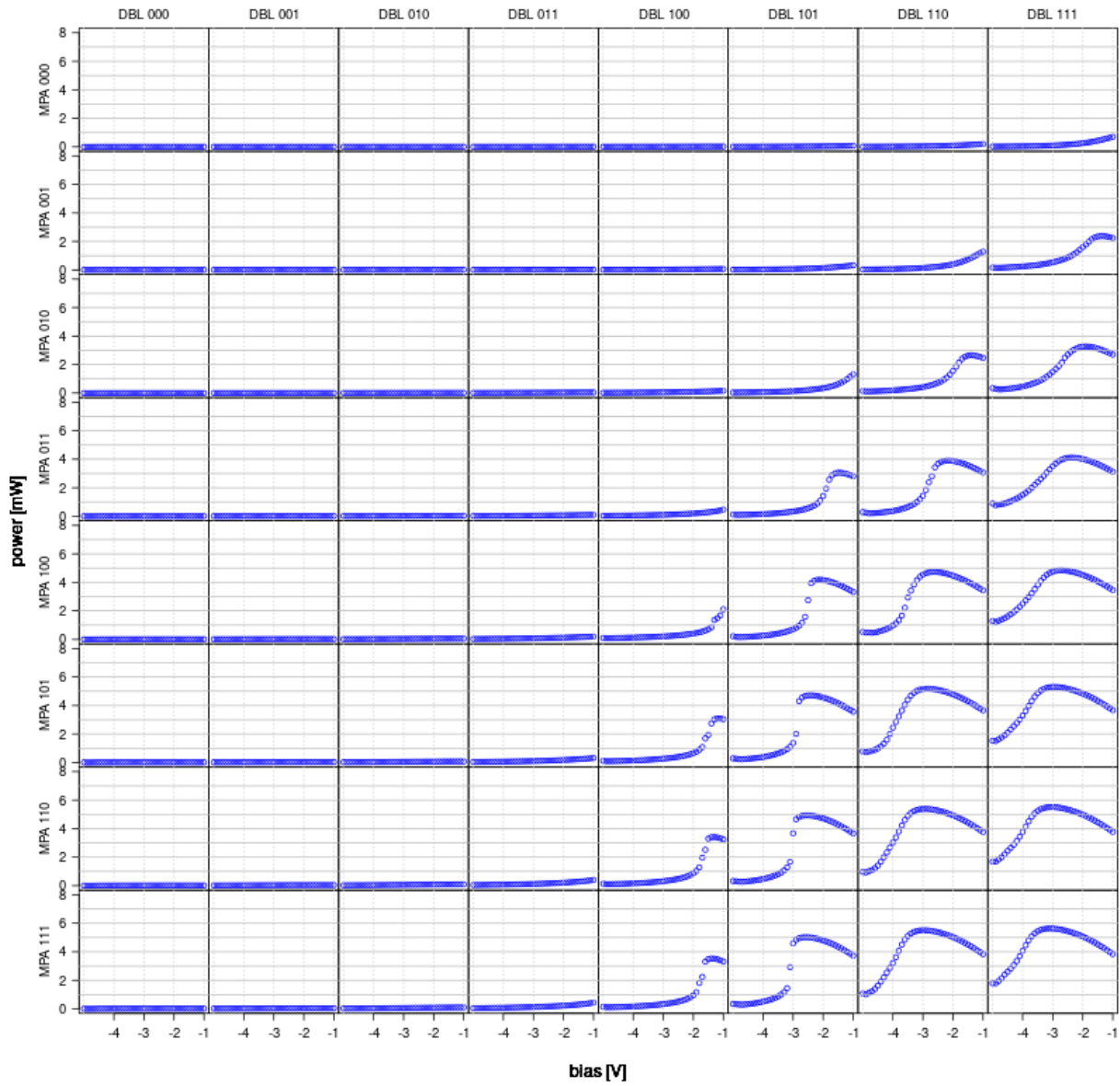
**LO scan (FM) @ 22.756875 GHz**



**LO scan (FM) @ 22.786875 GHz**

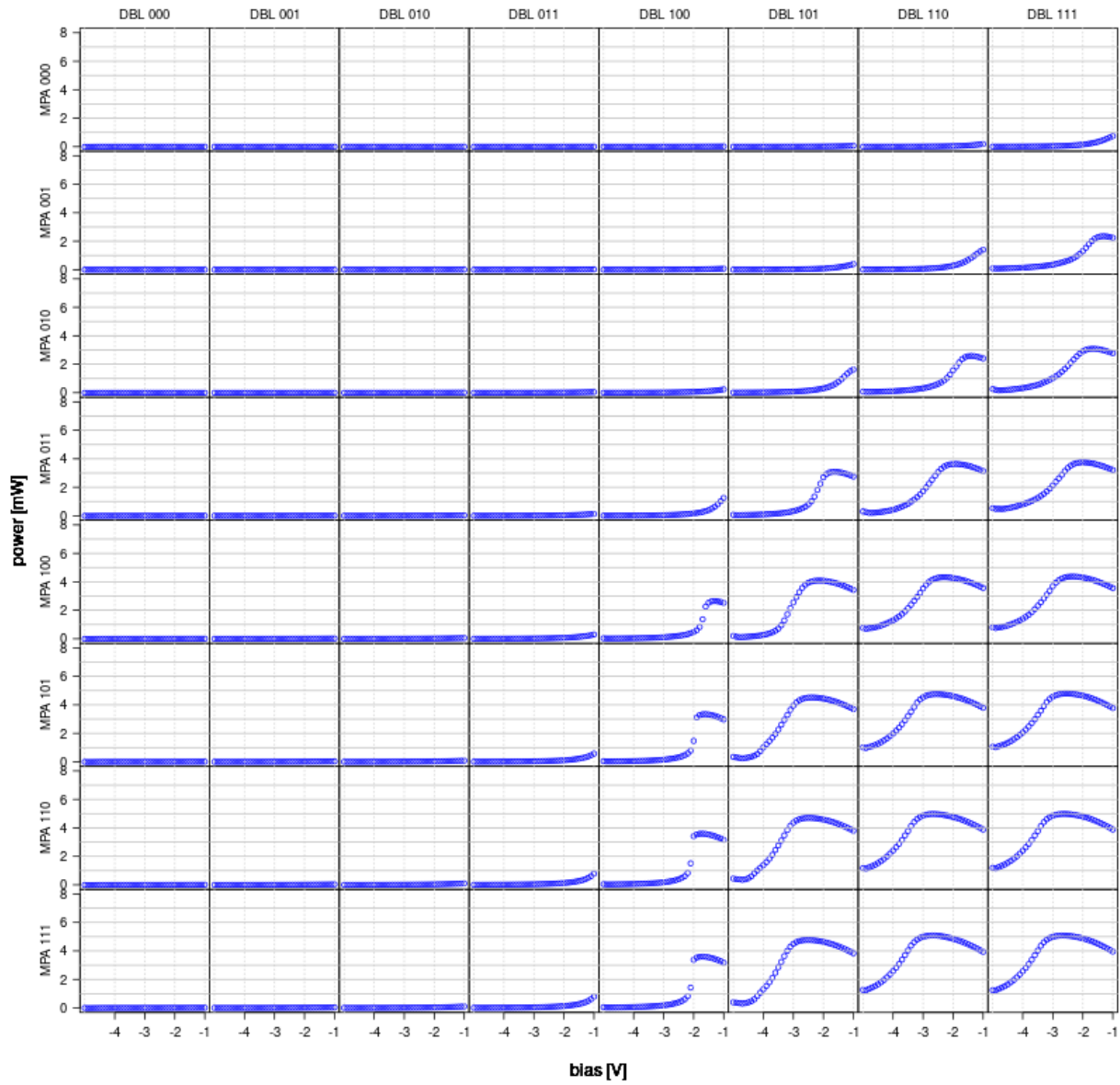


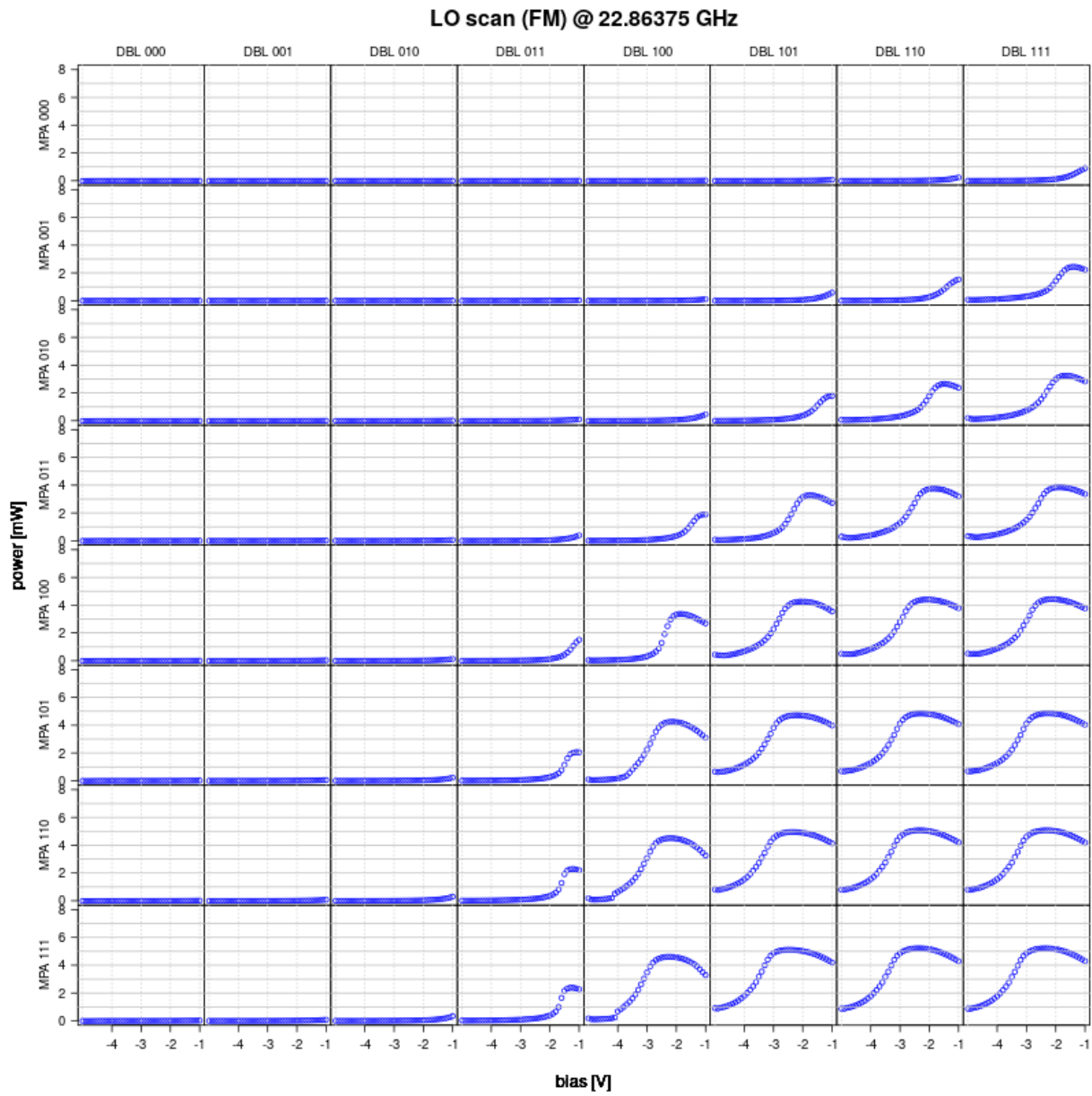
**LO scan (FM) @ 22.809375 GHz**



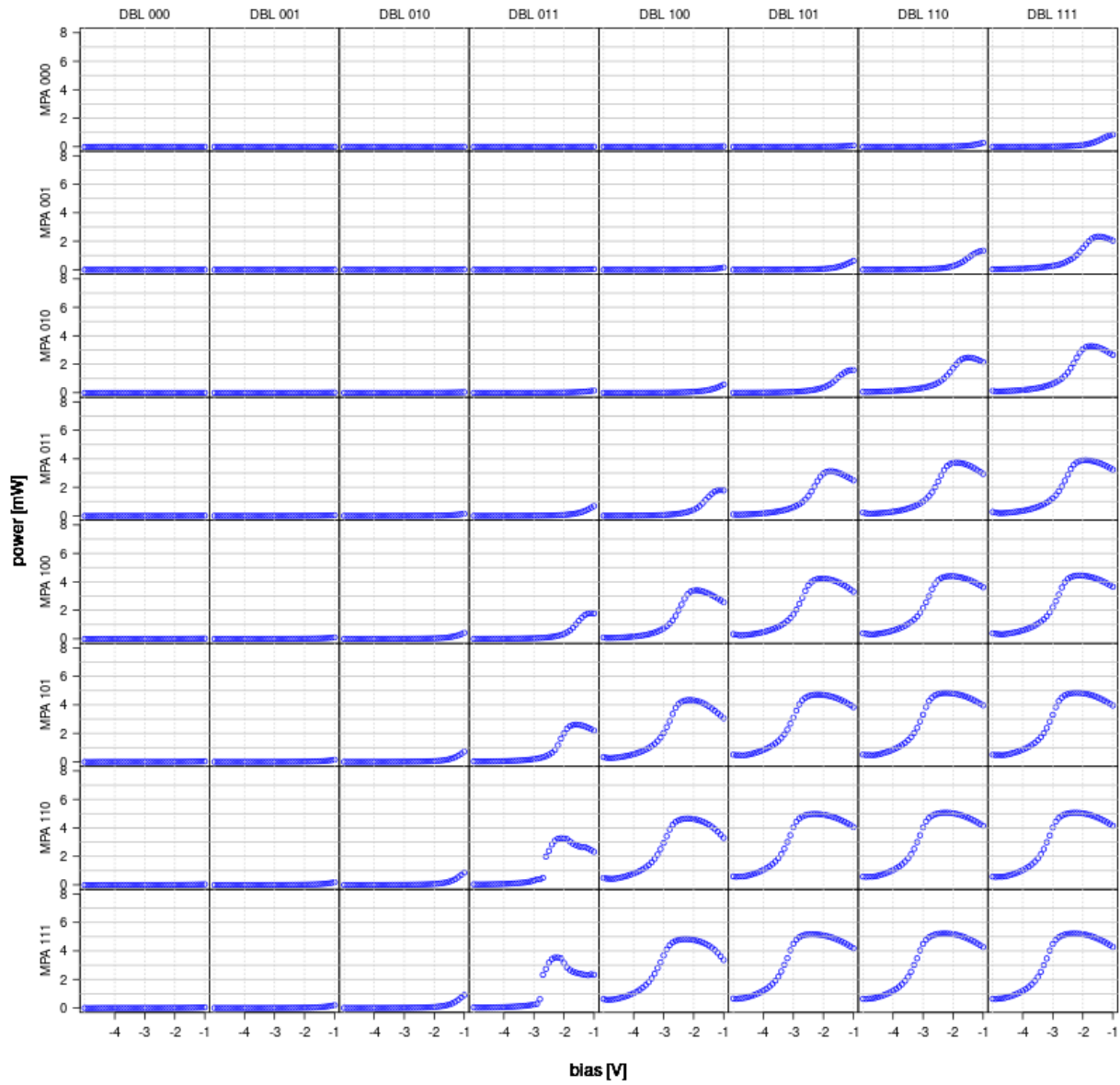


**LO scan (FM) @ 22.8375 GHz**

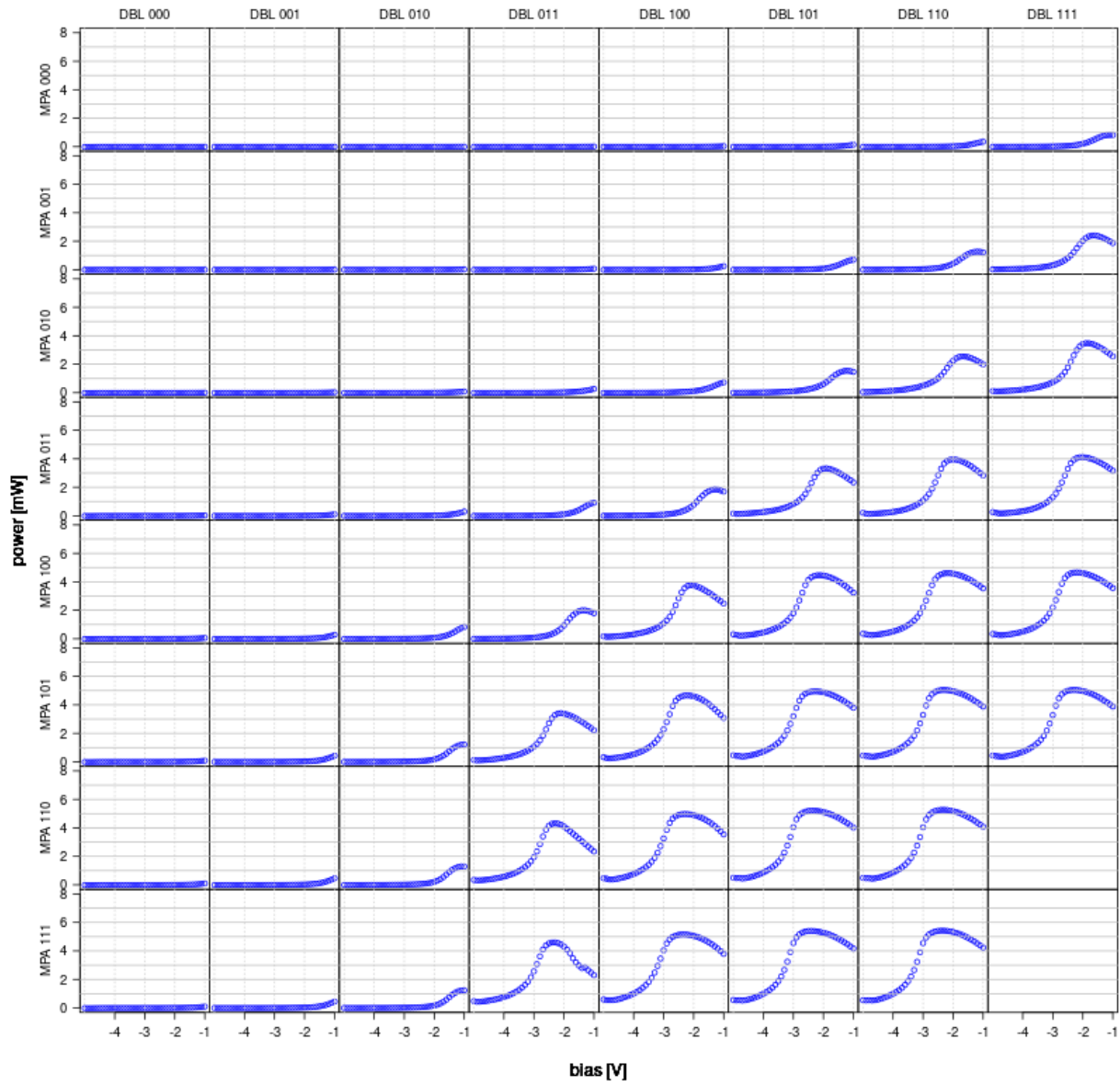




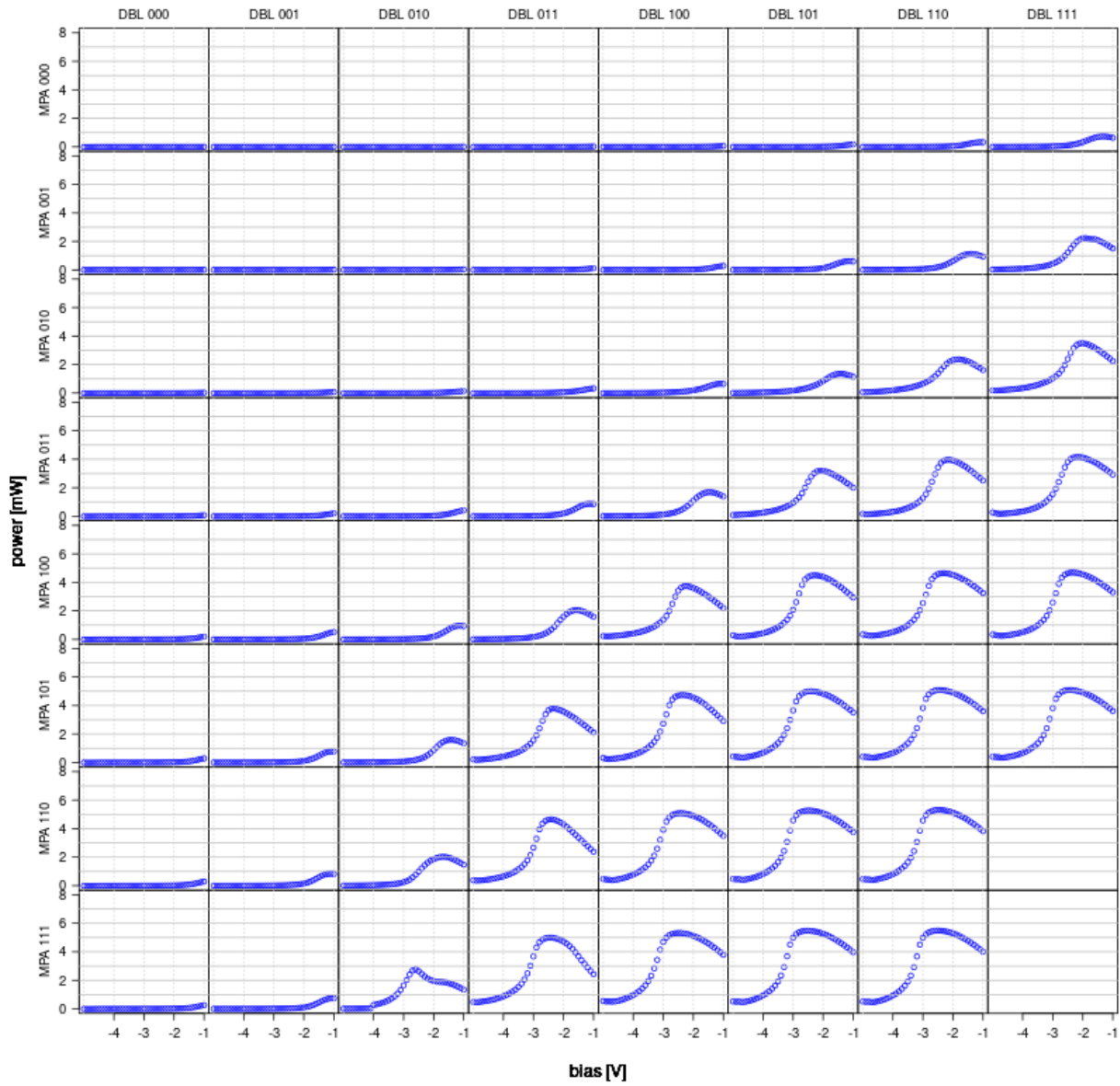
**LO scan (FM) @ 22.888125 GHz**



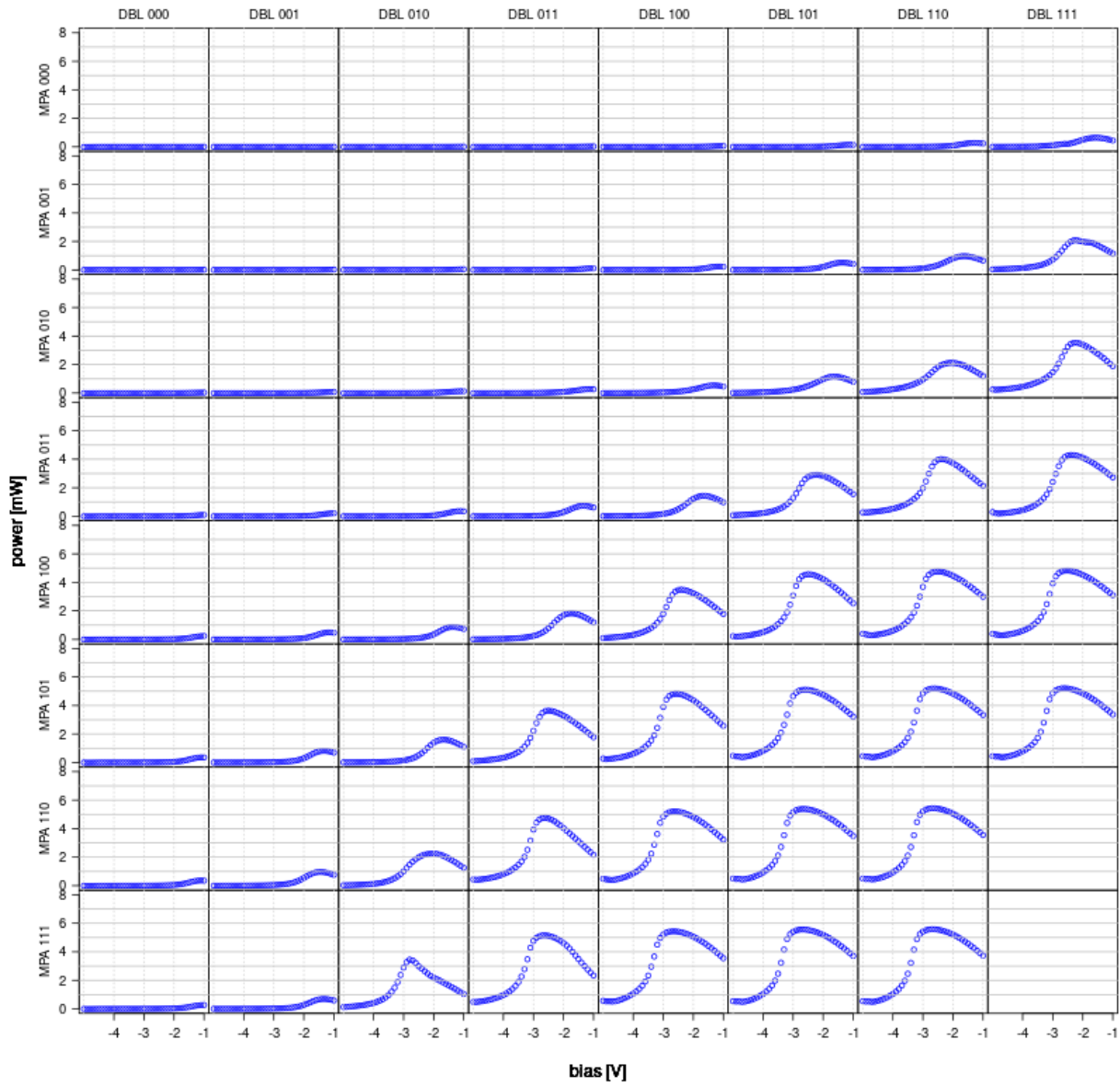
**LO scan (FM) @ 22.9125 GHz**

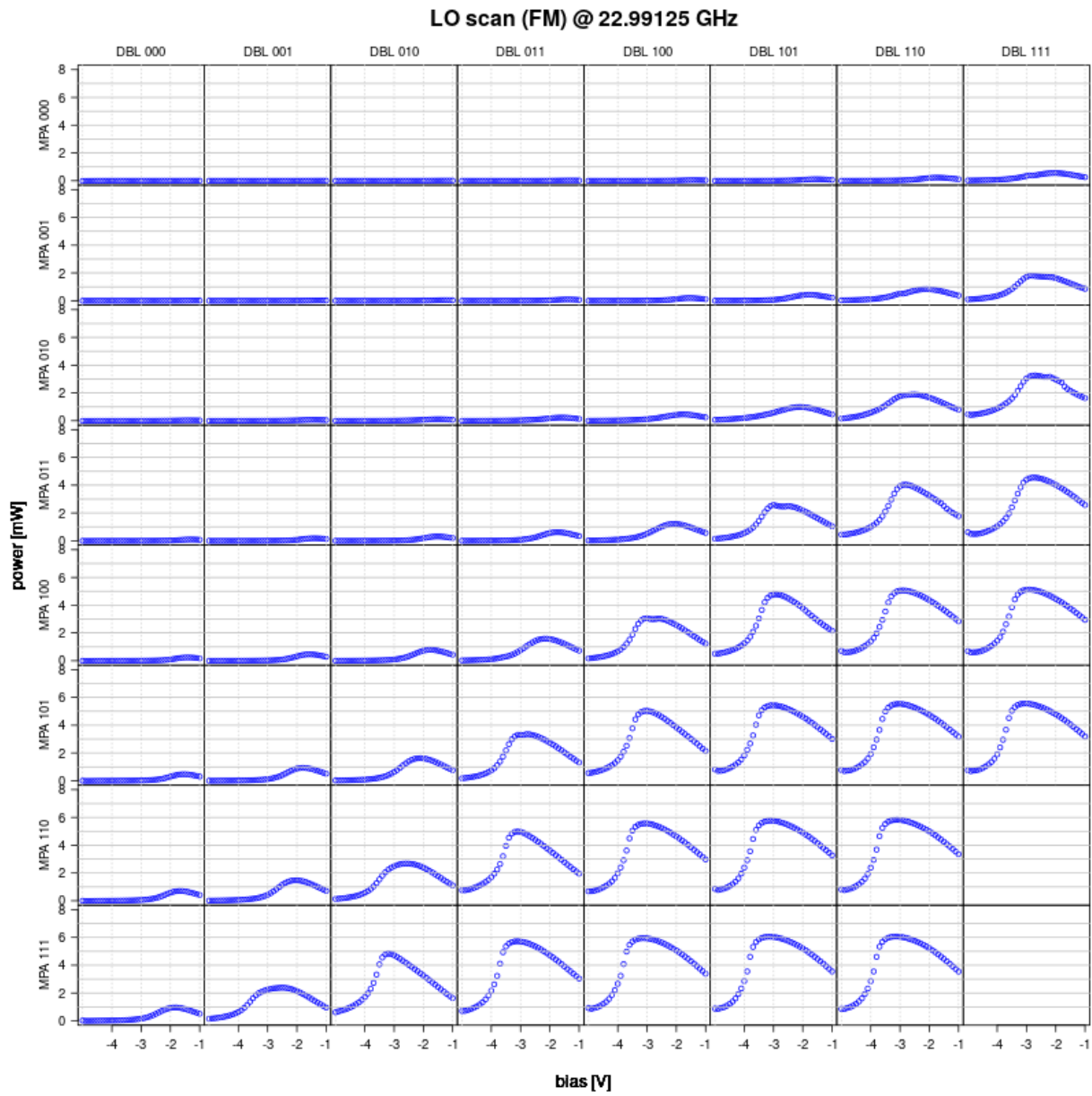


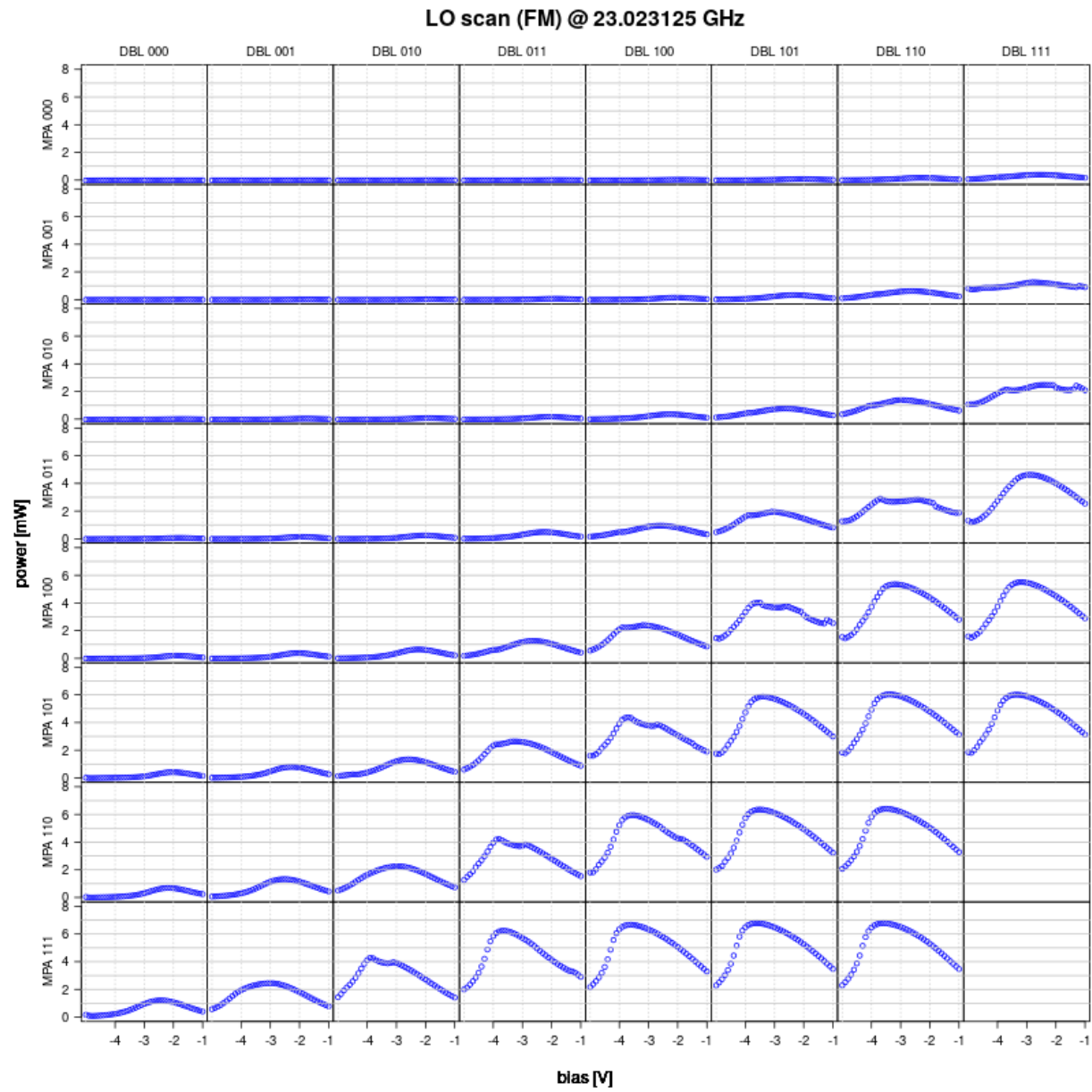
LO scan (FM) @ 22.935 GHz



**LO scan (FM) @ 22.9575 GHz**

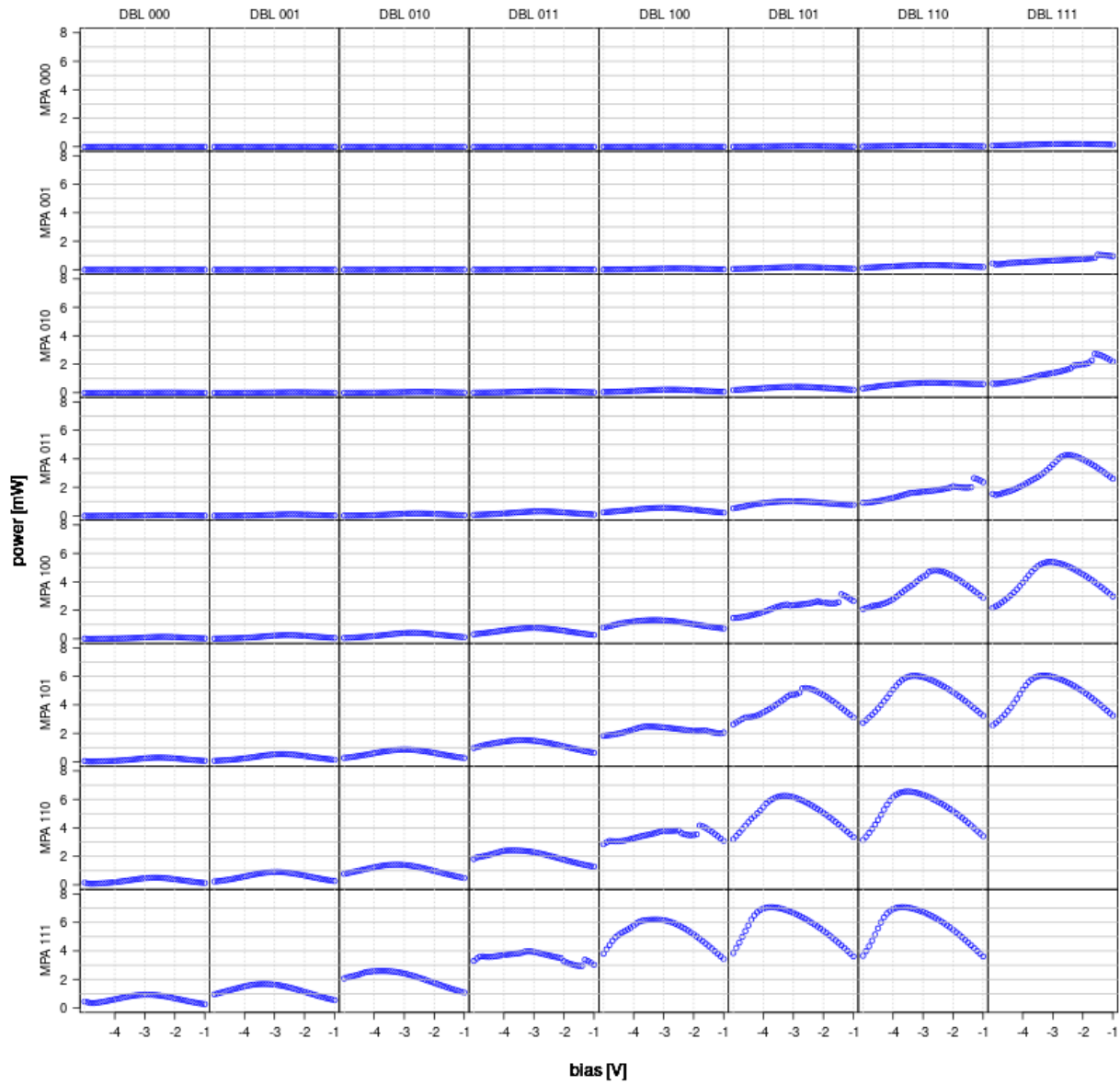


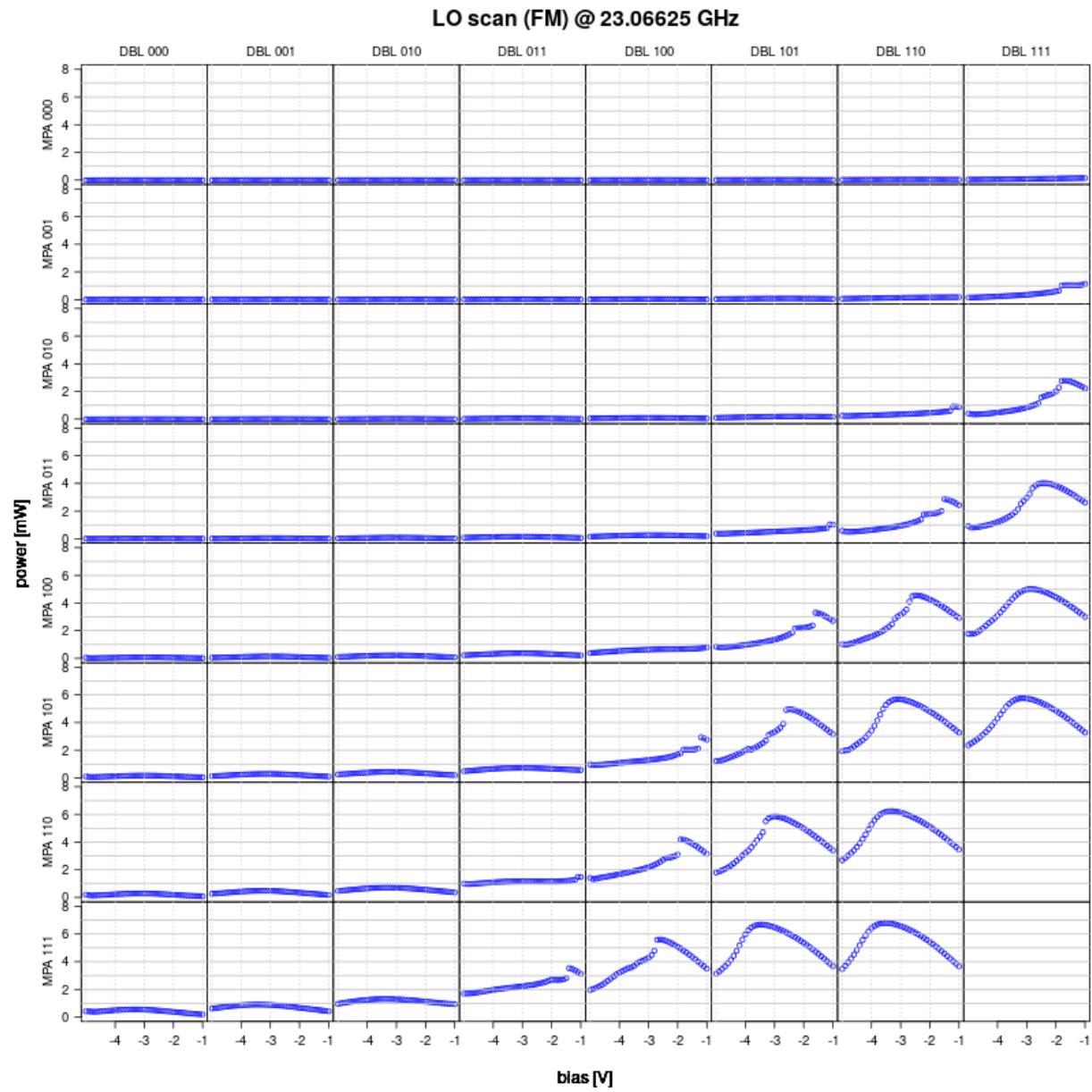


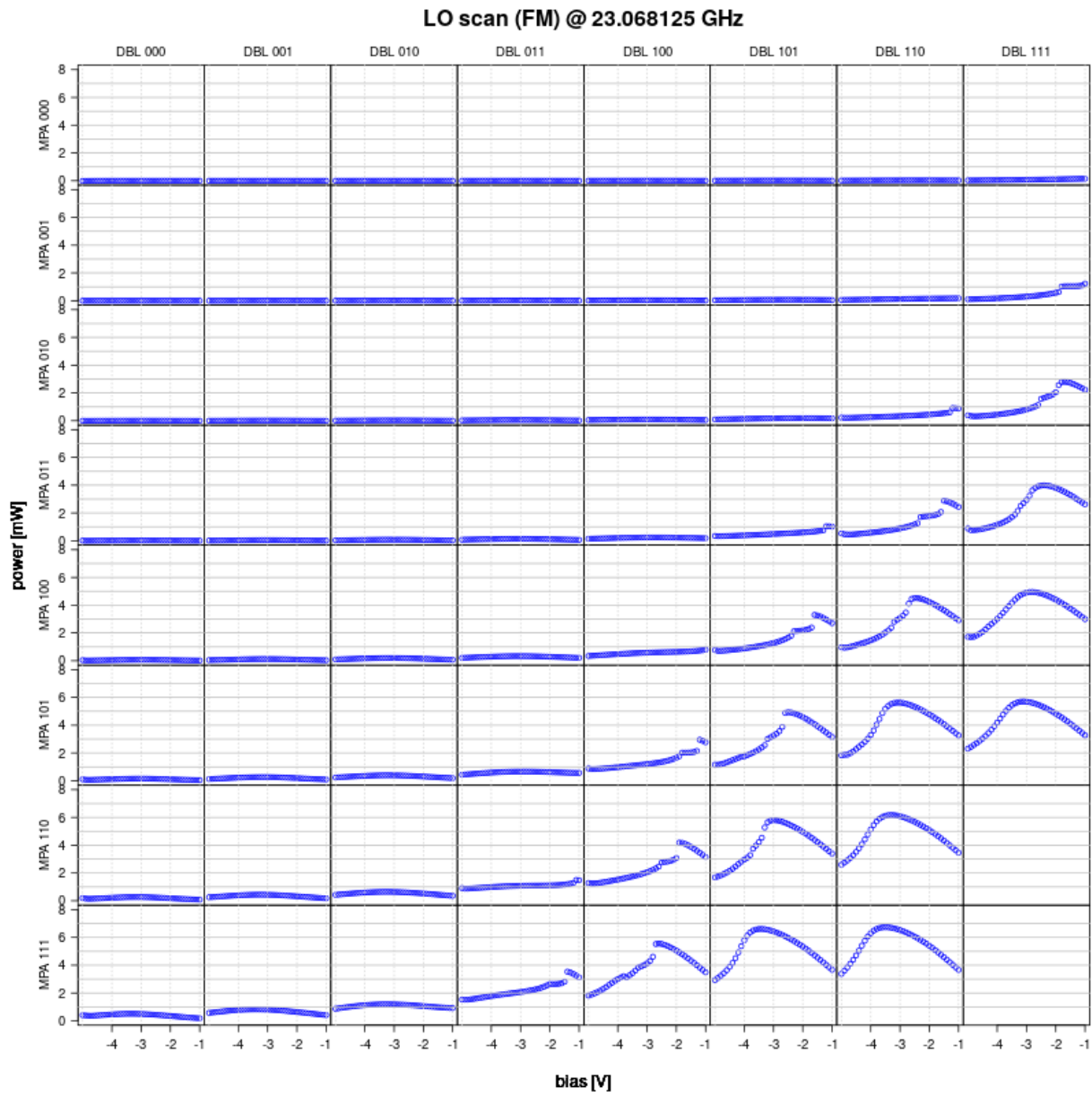




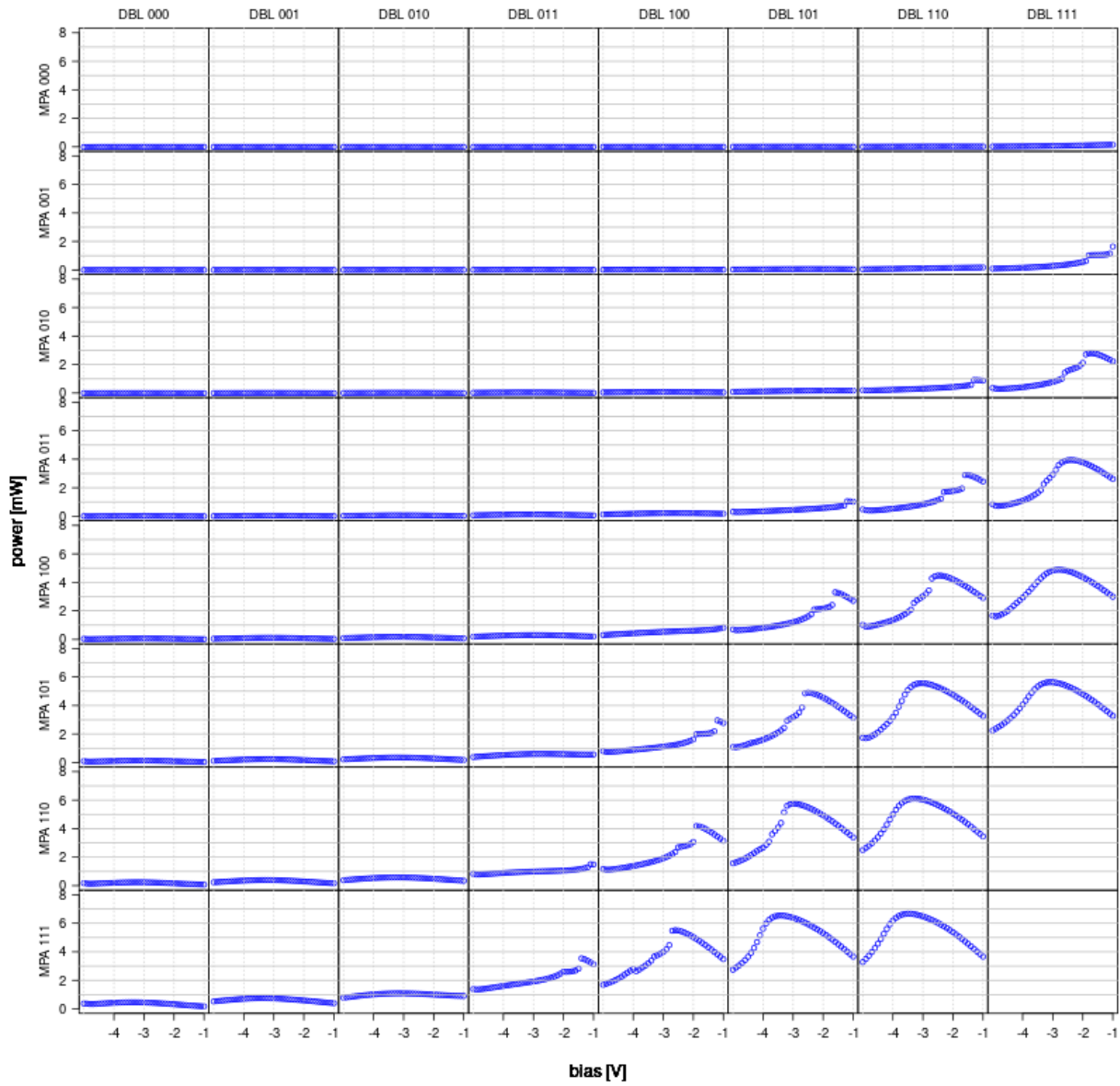
LO scan (FM) @ 23.0475 GHz



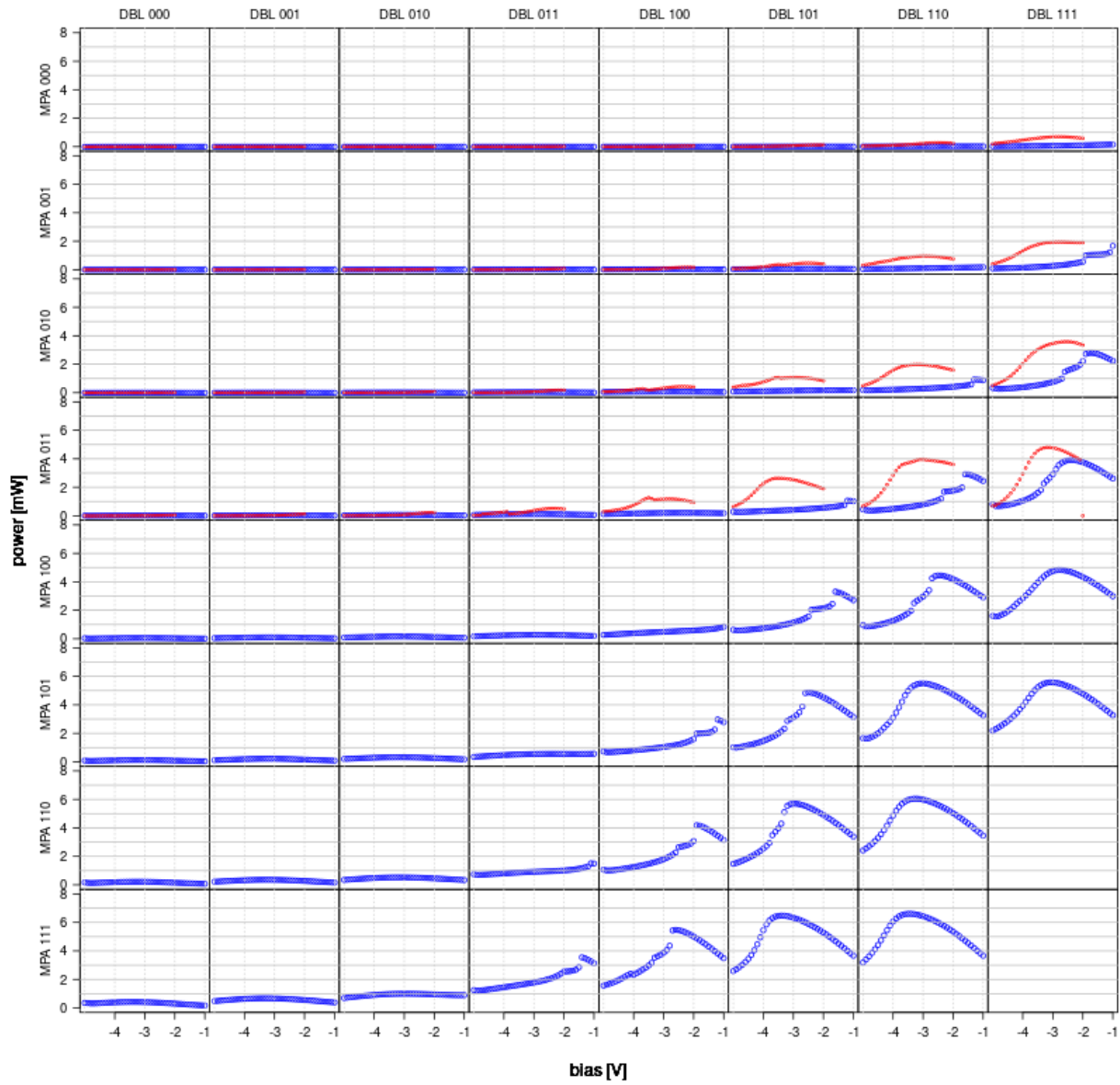


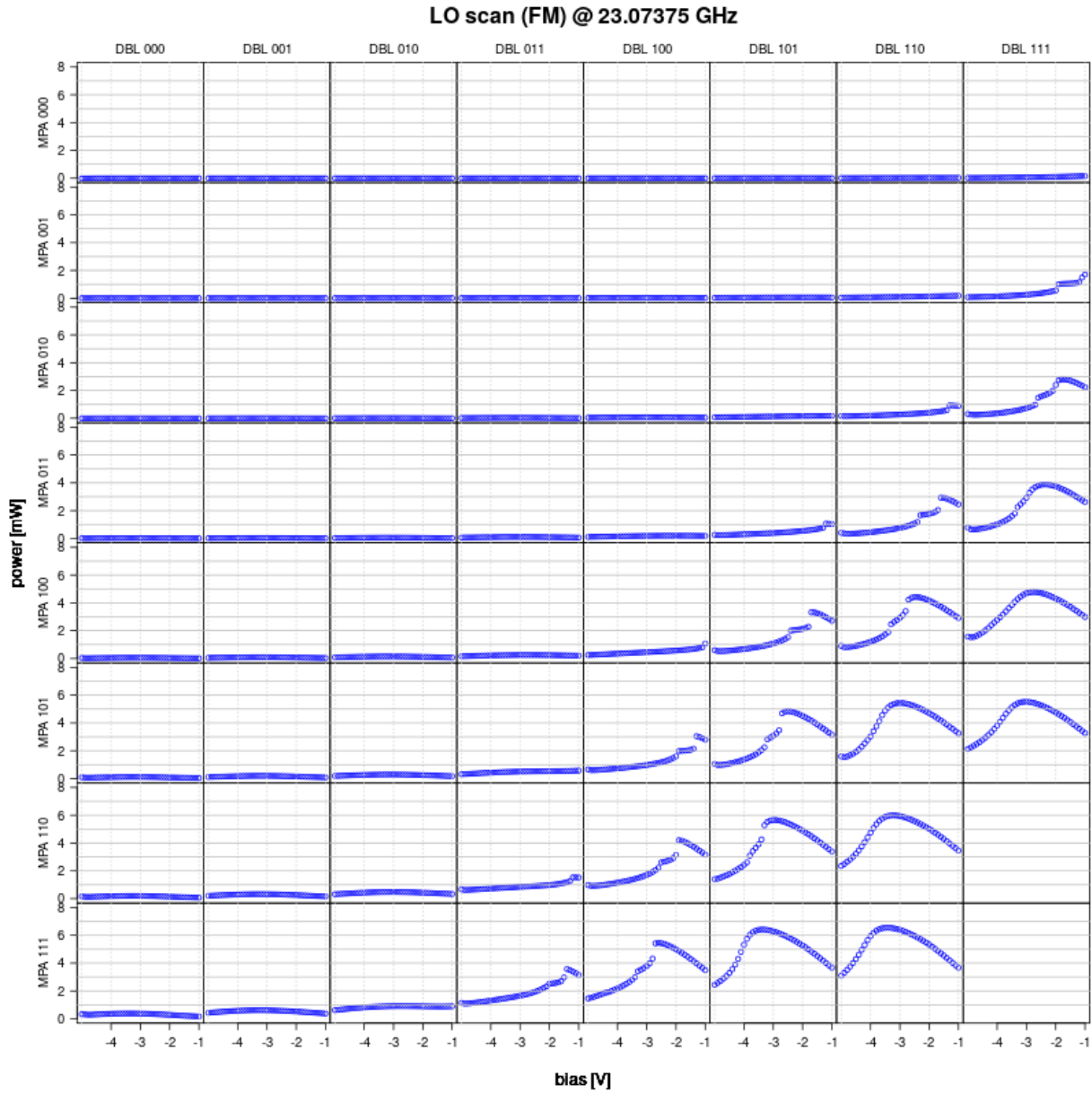


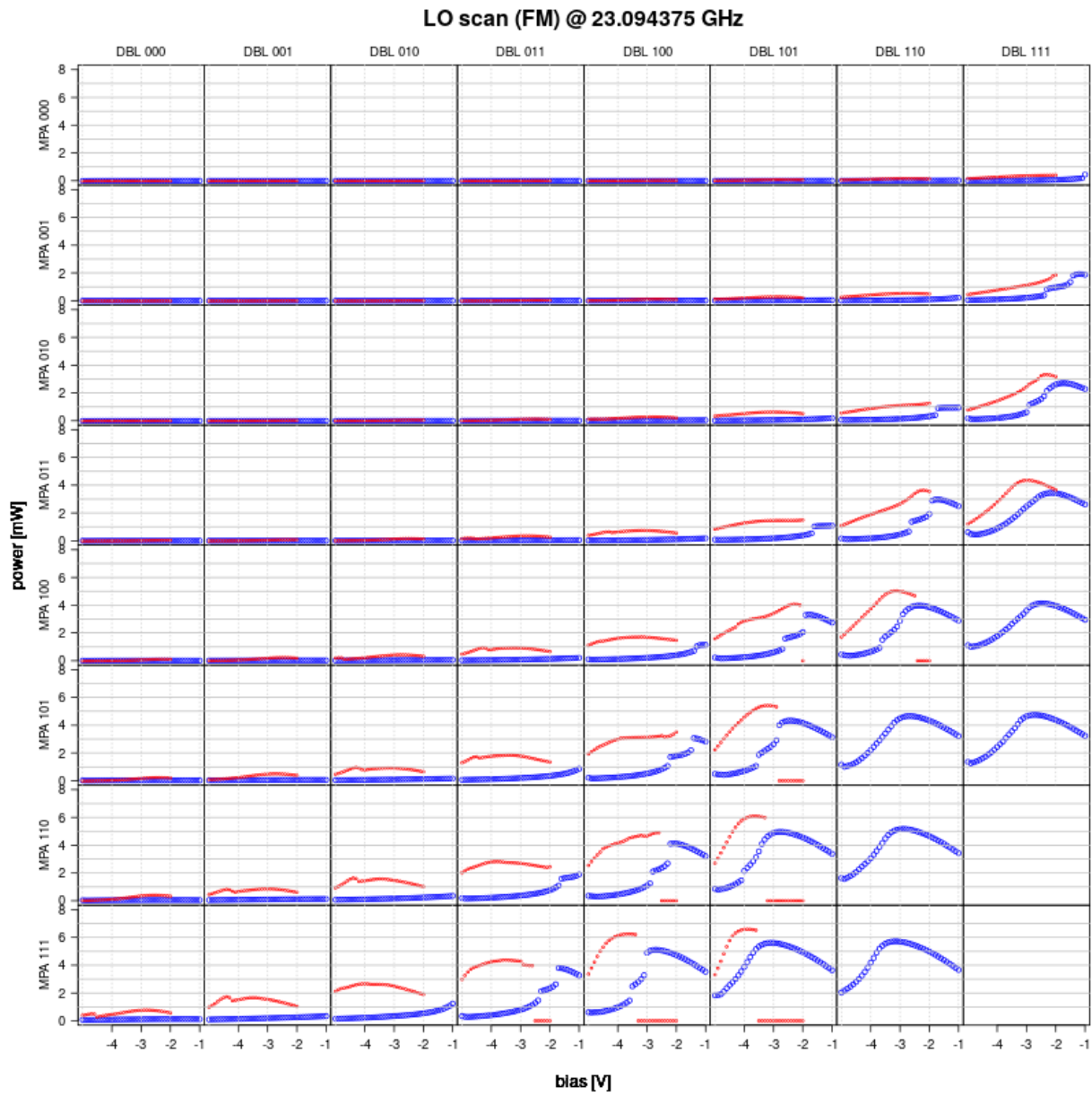
LO scan (FM) @ 23.07 GHz

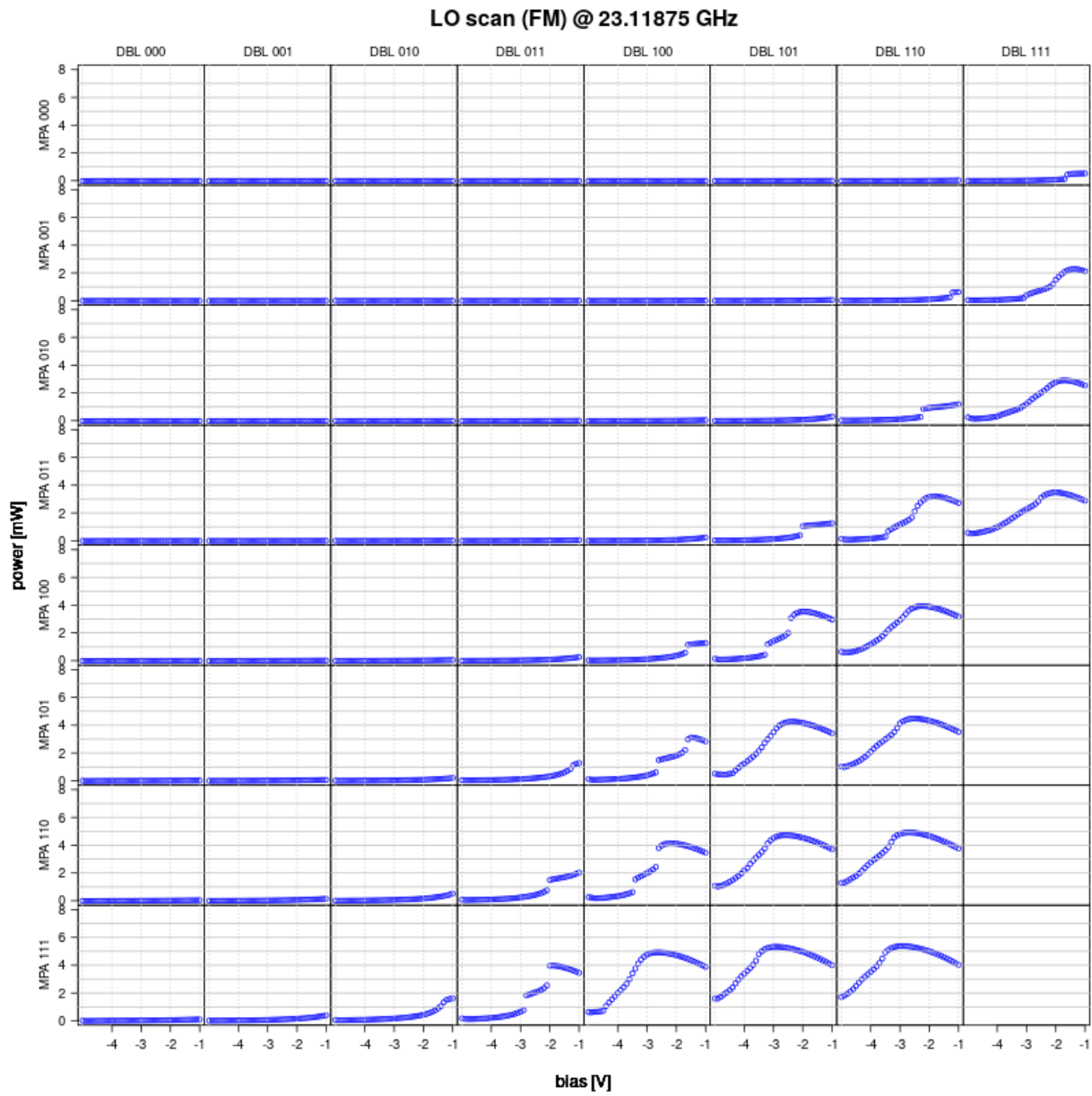


LO scan (FM) @ 23.071875 GHz

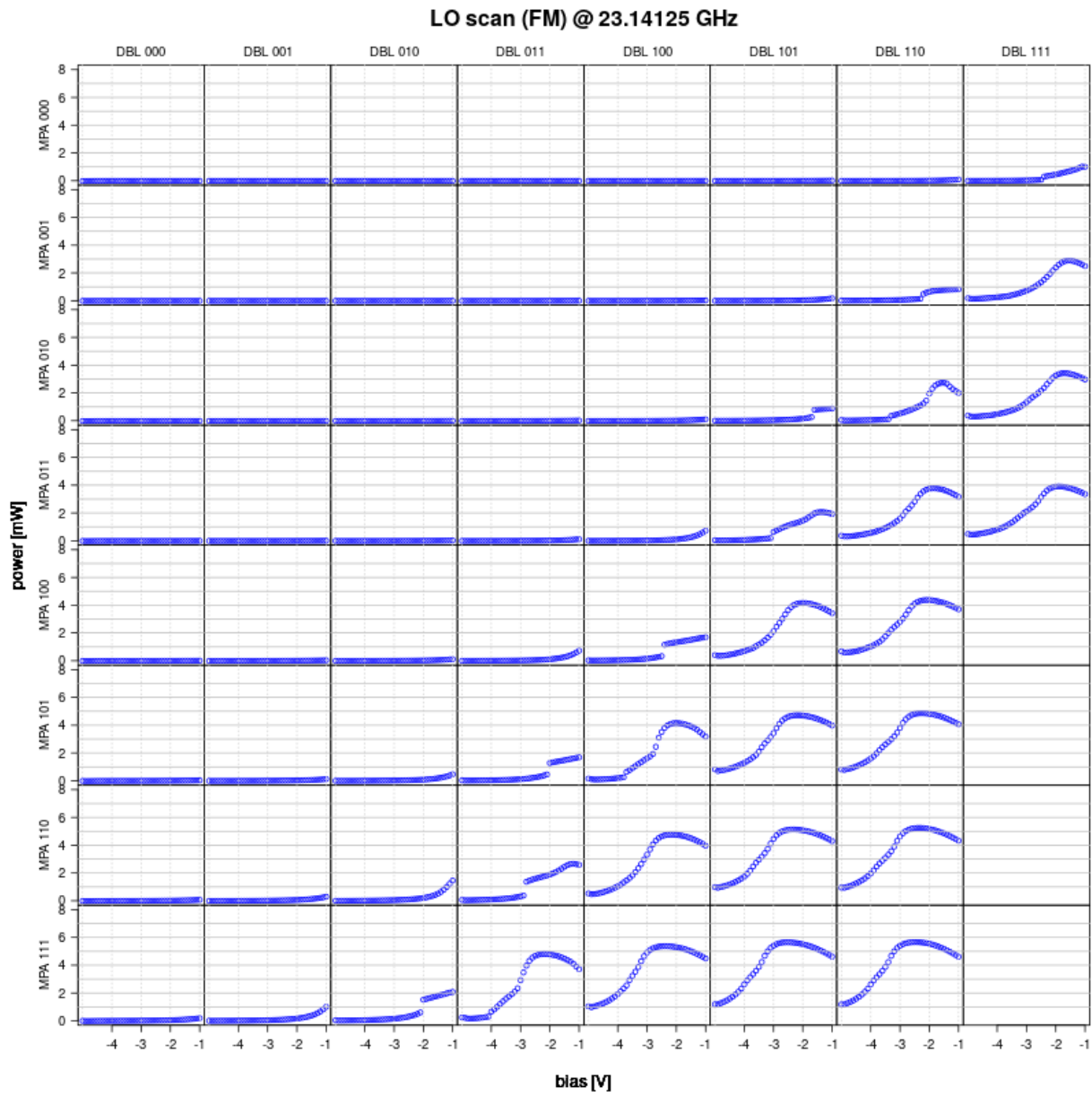


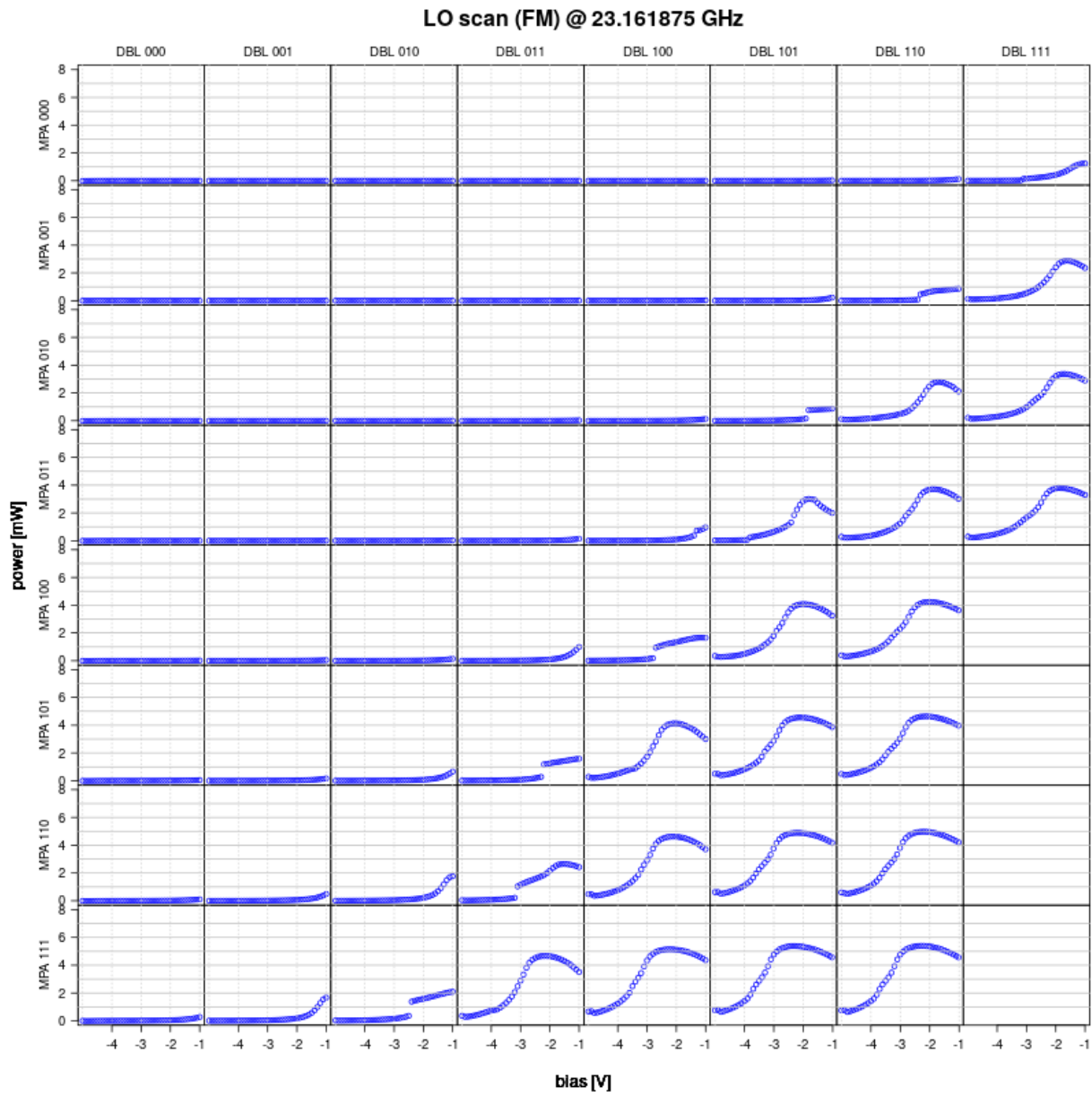




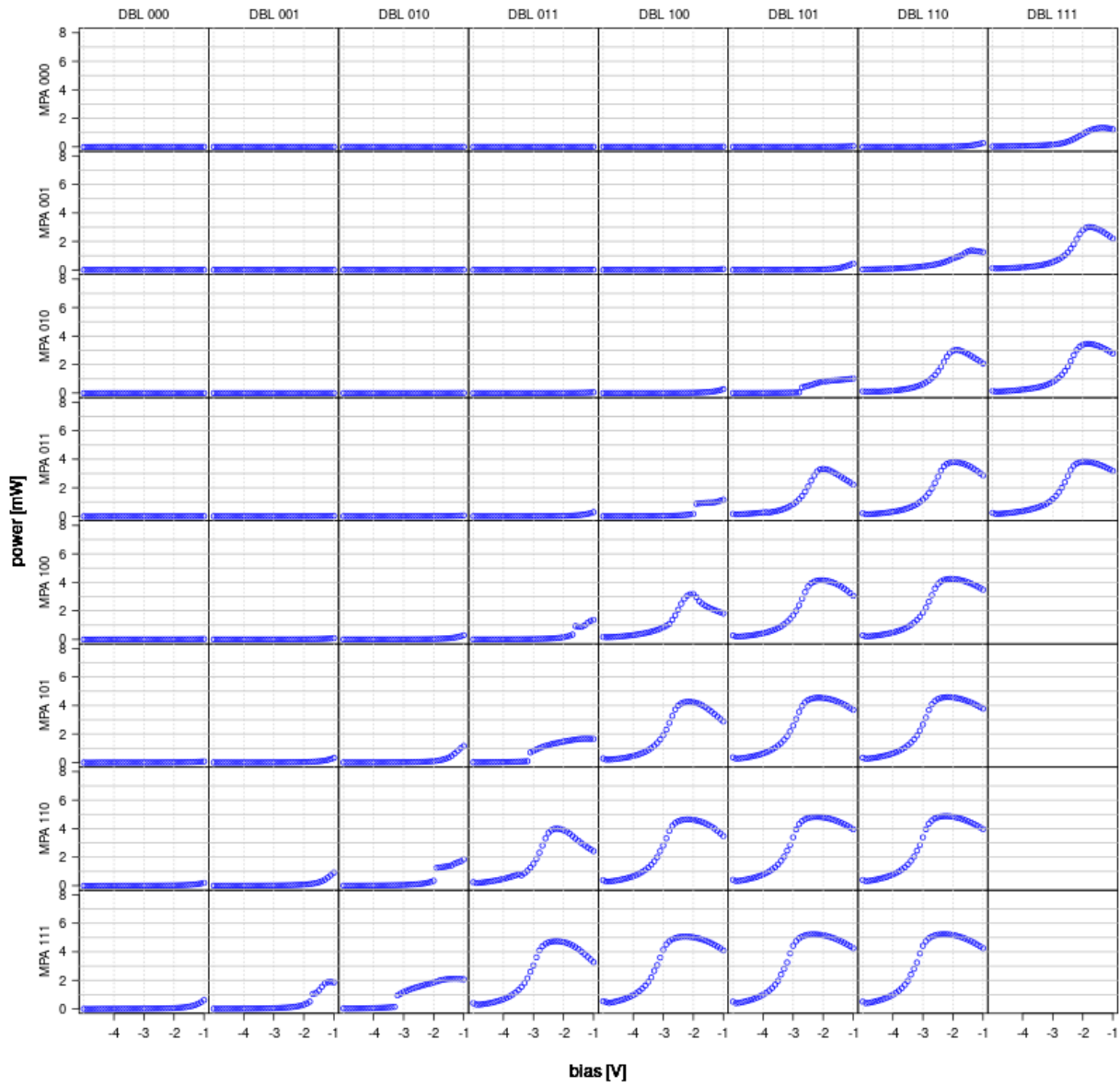


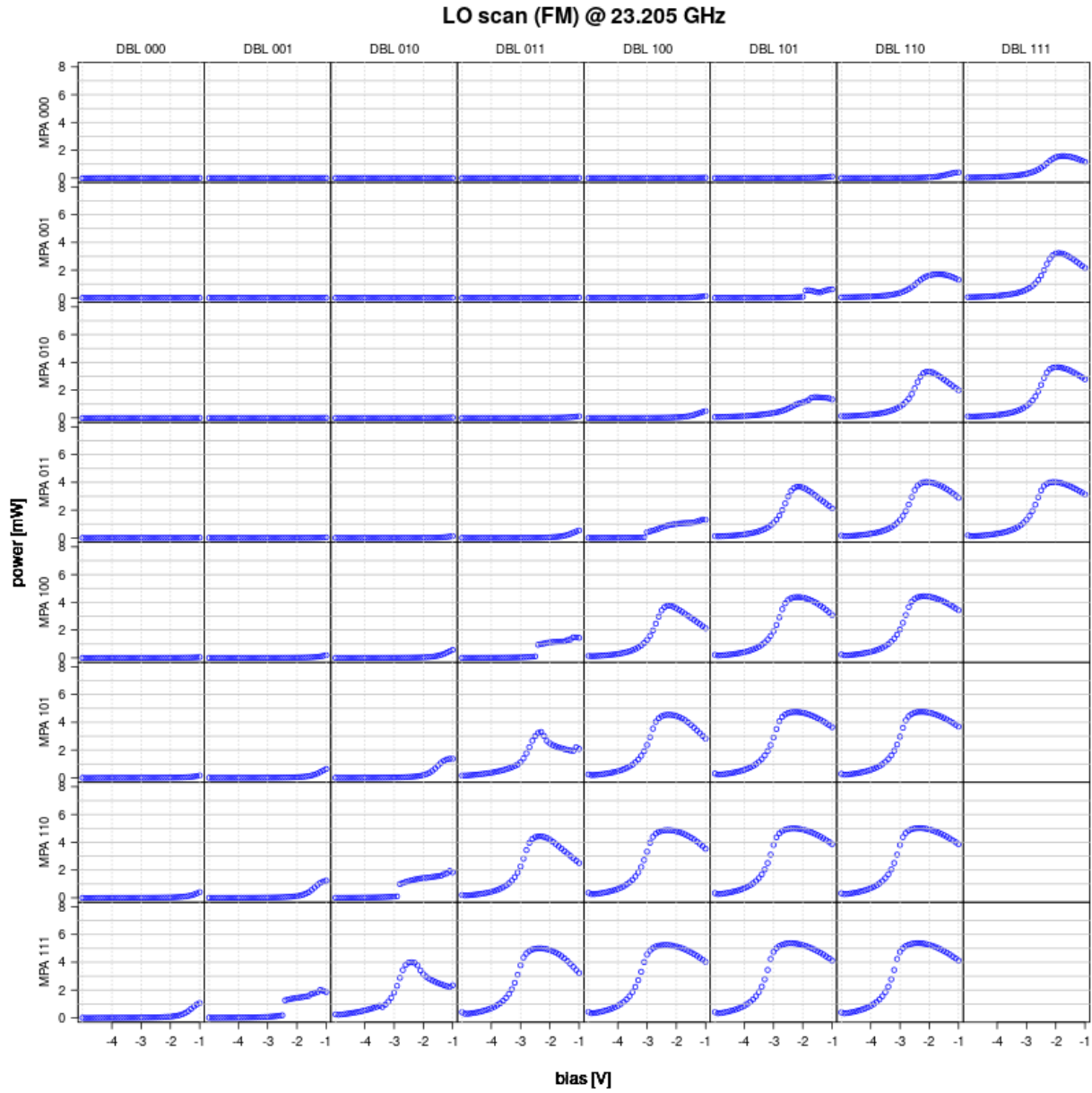




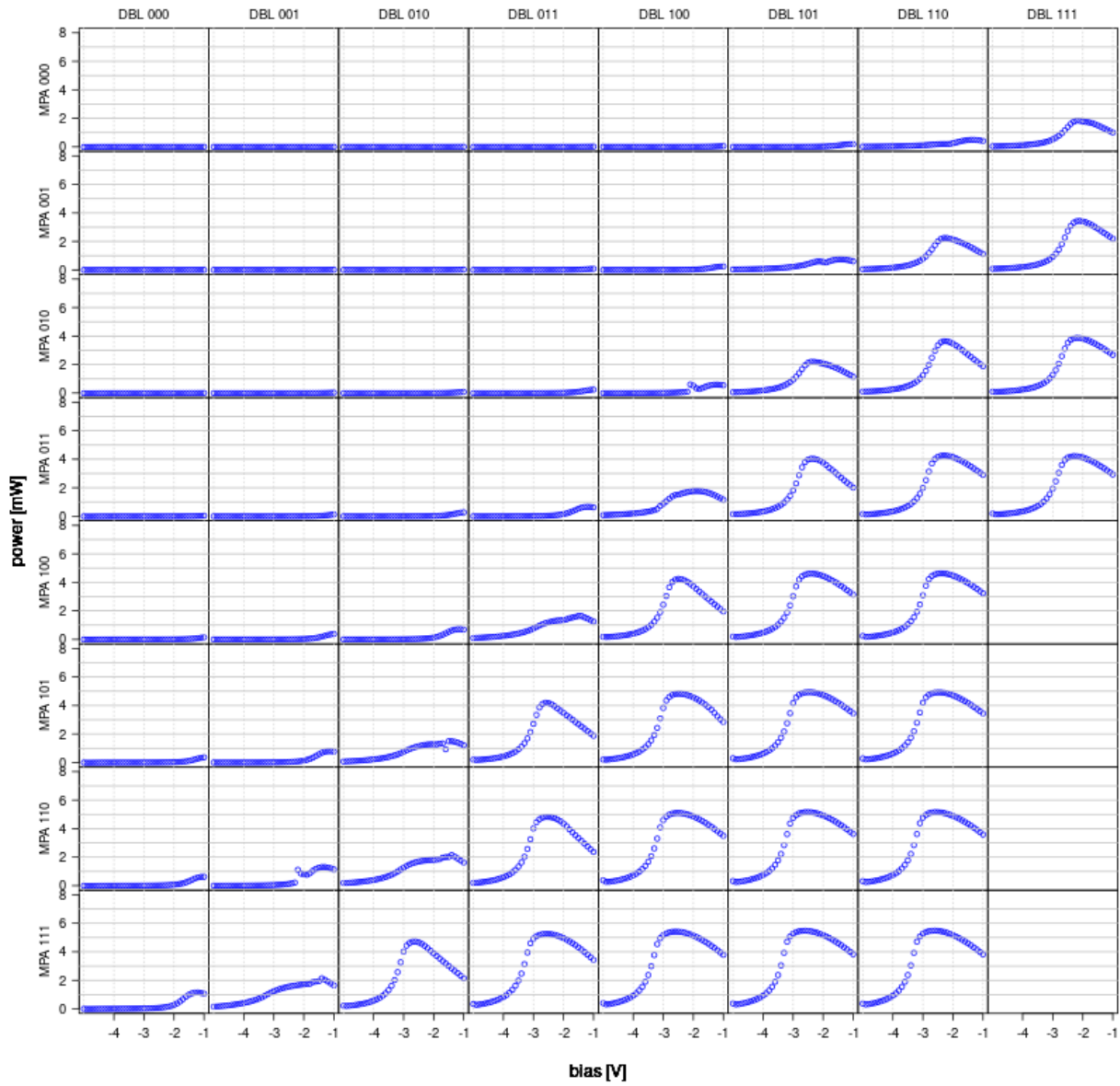


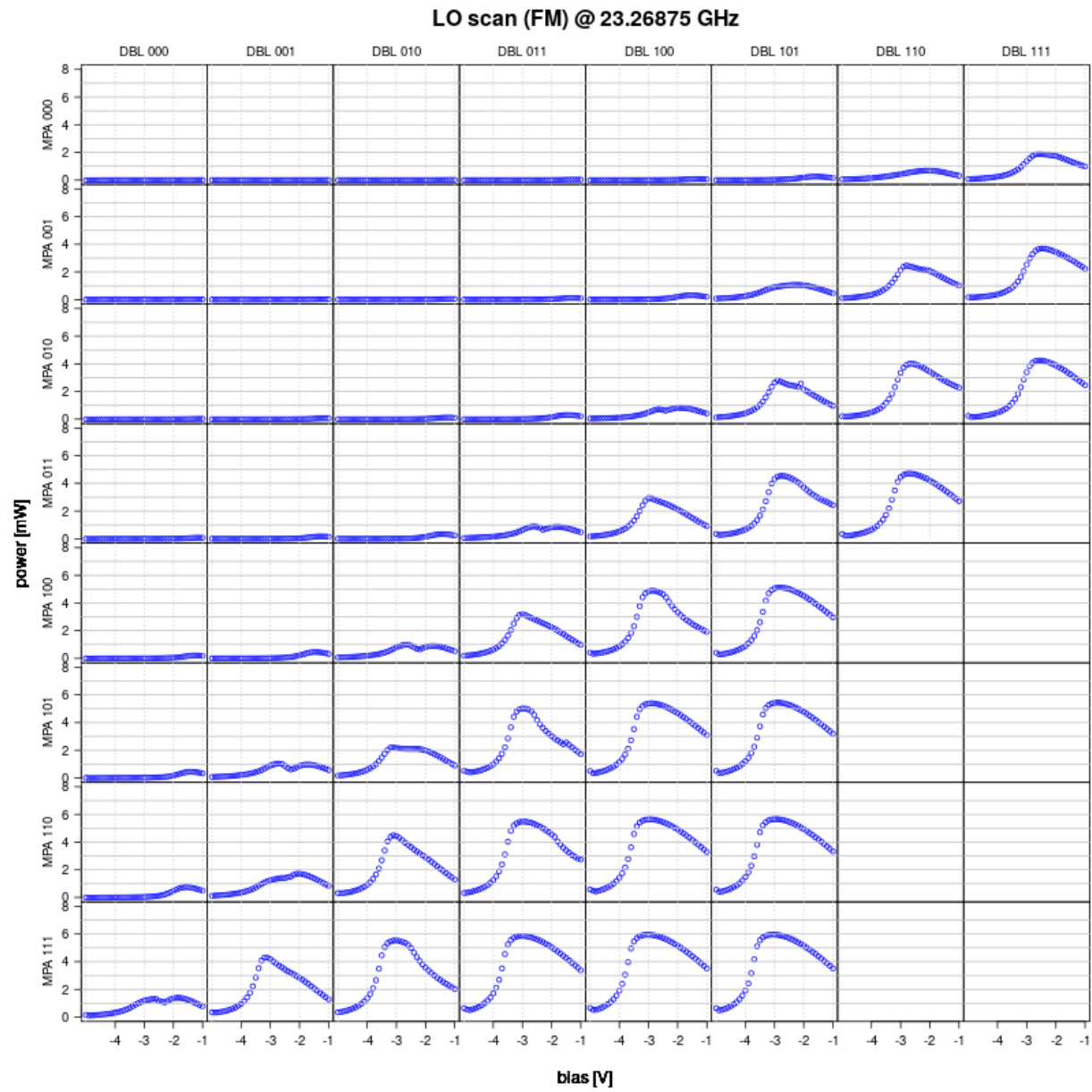
**LO scan (FM) @ 23.184375 GHz**

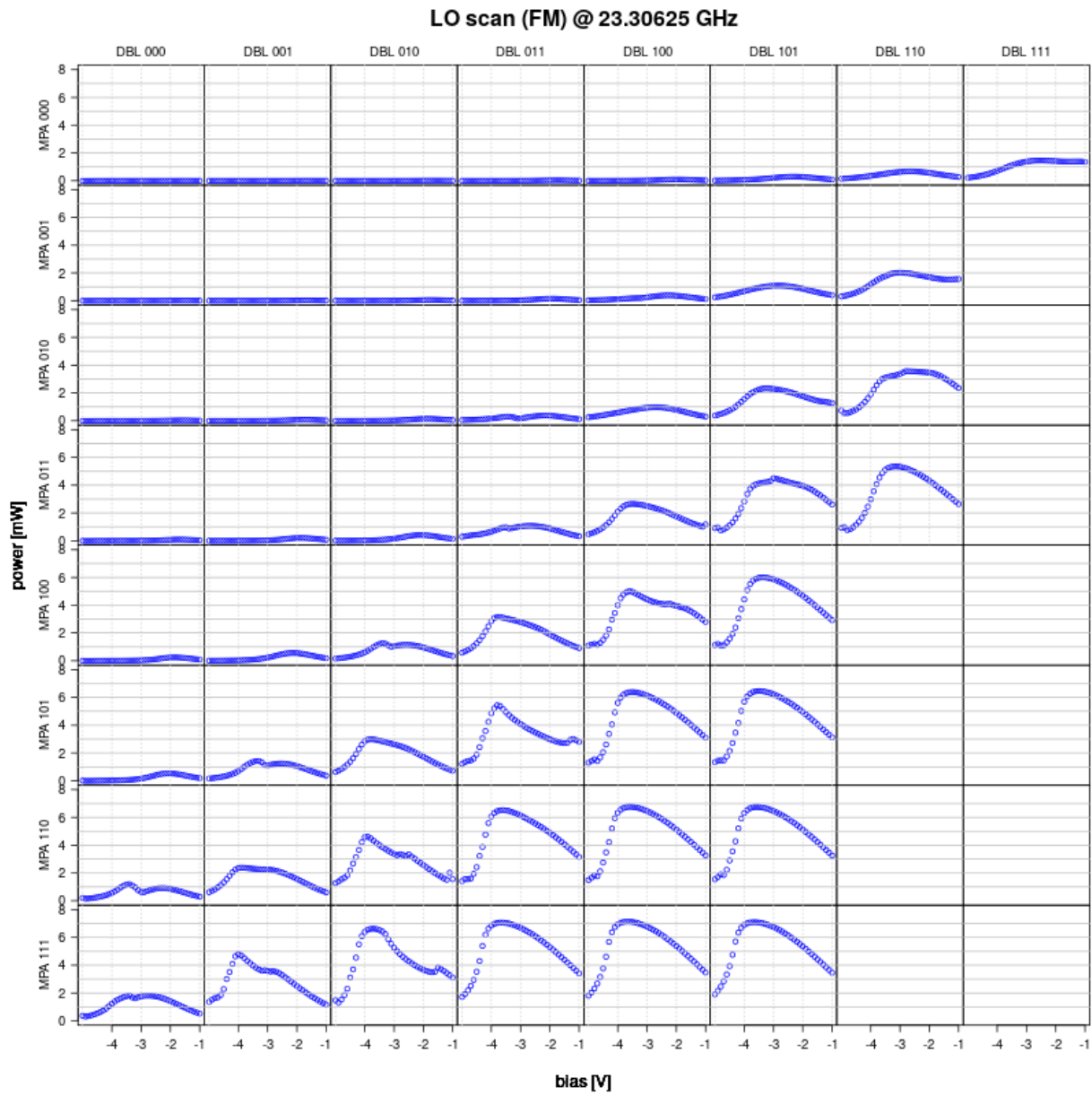


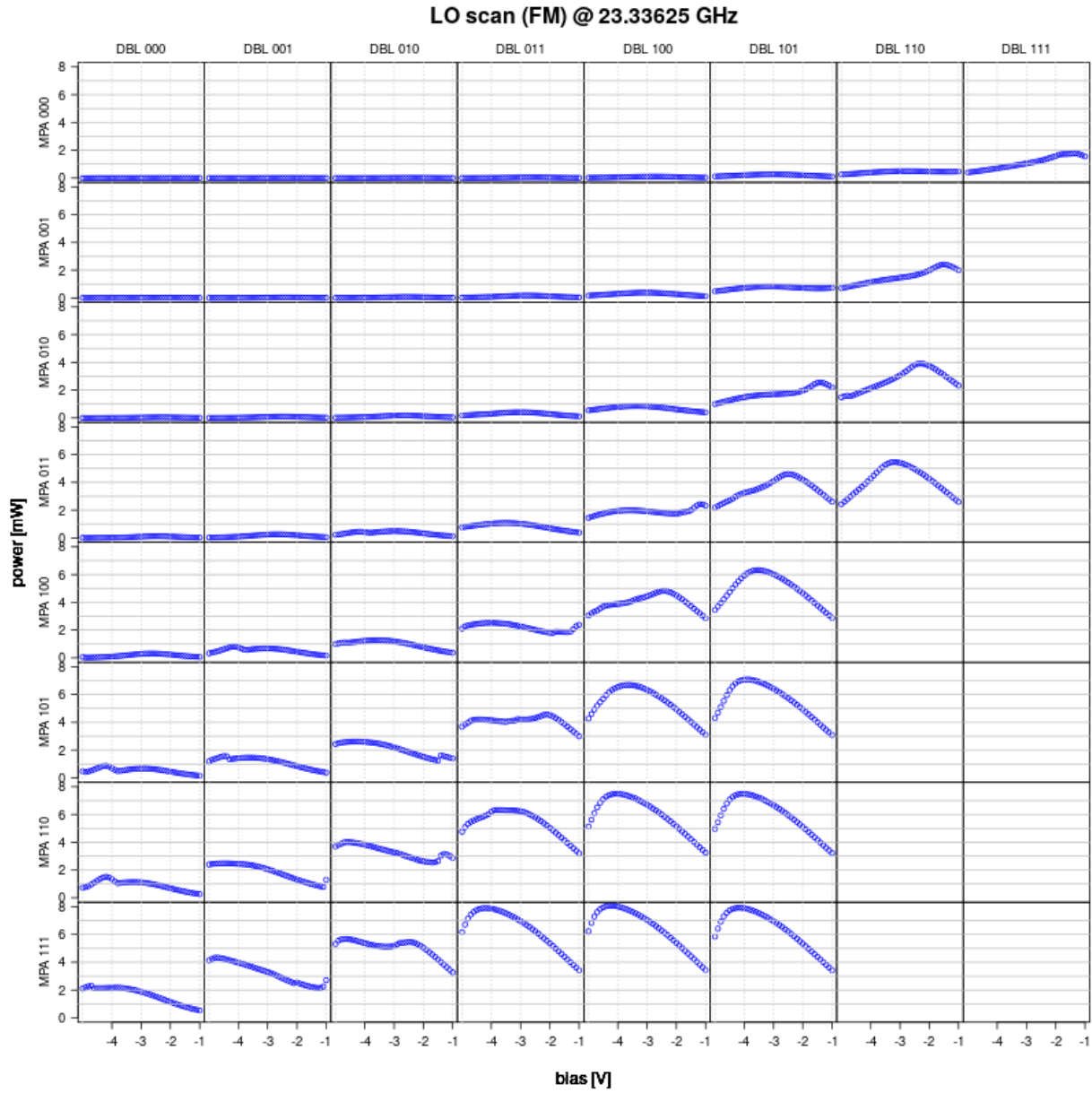


**LO scan (FM) @ 23.233125 GHz**

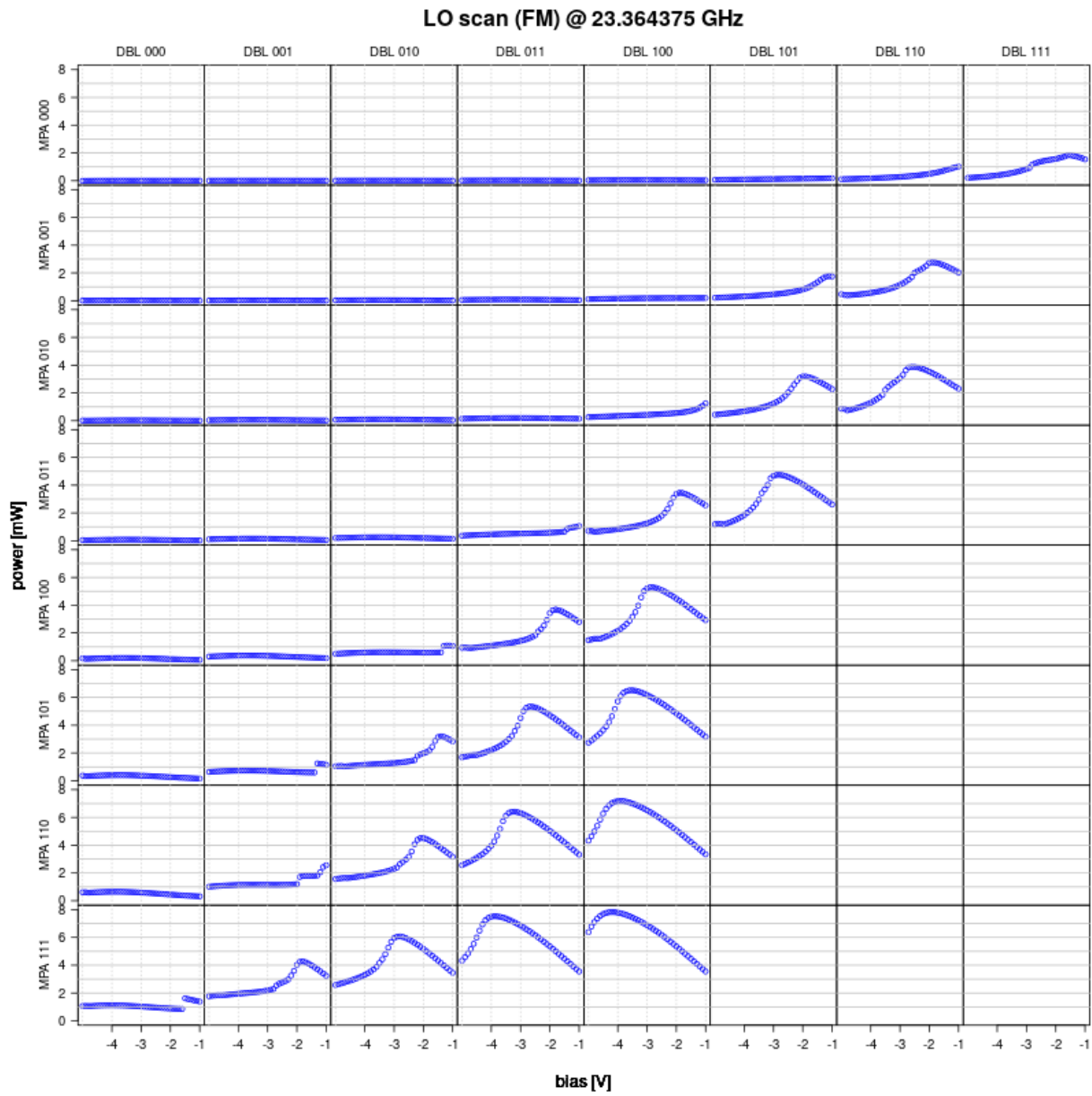




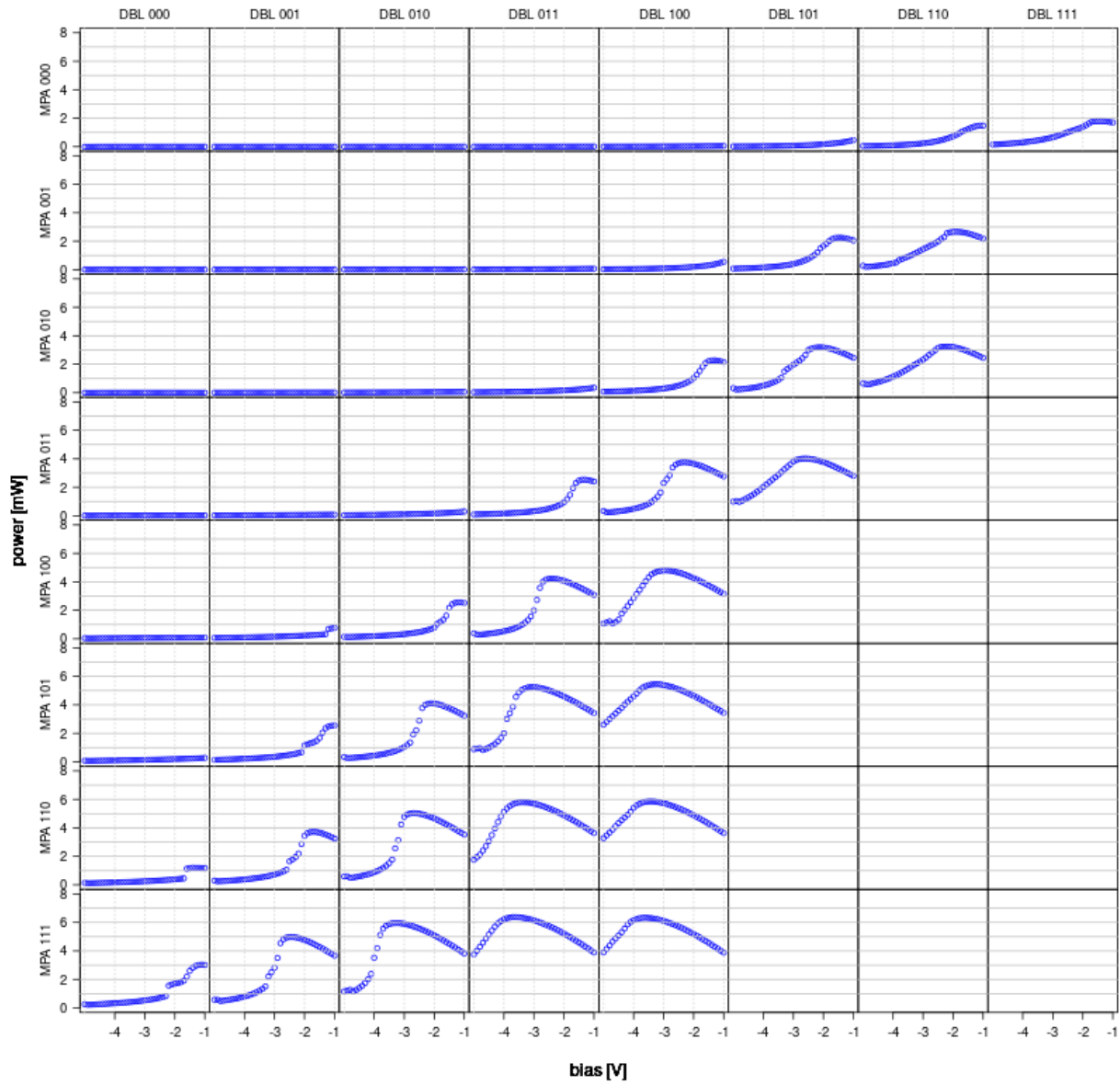


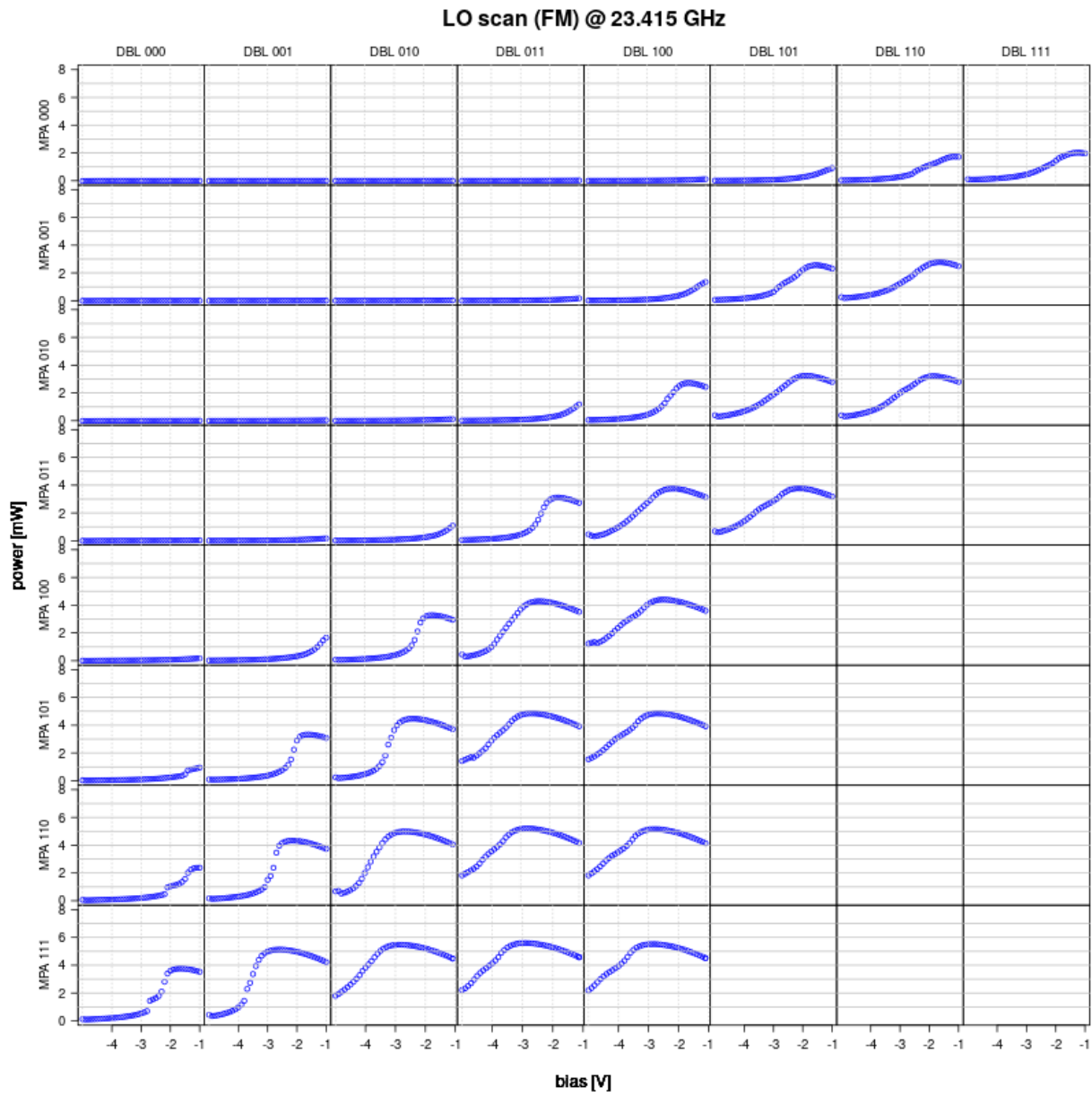




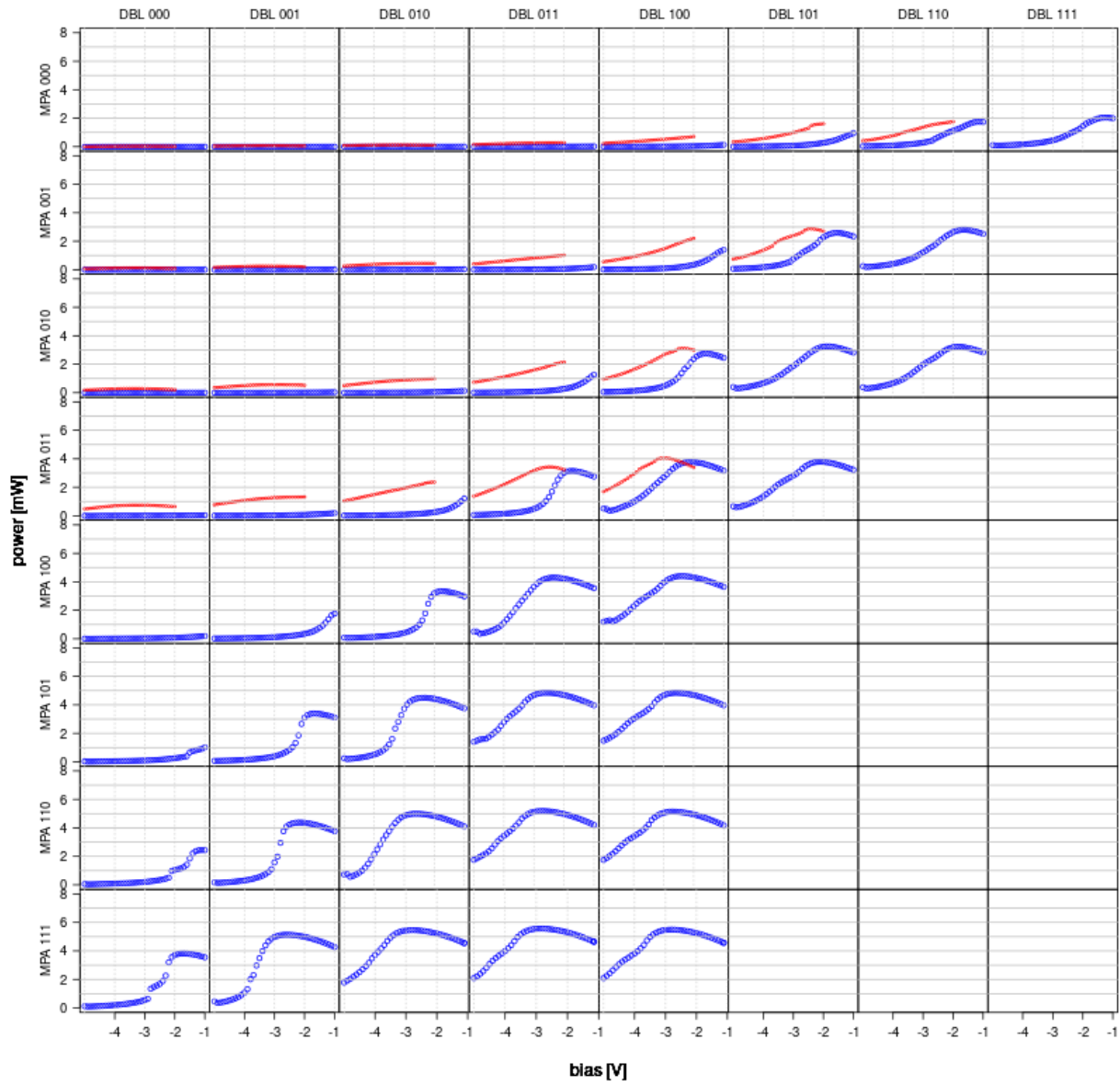


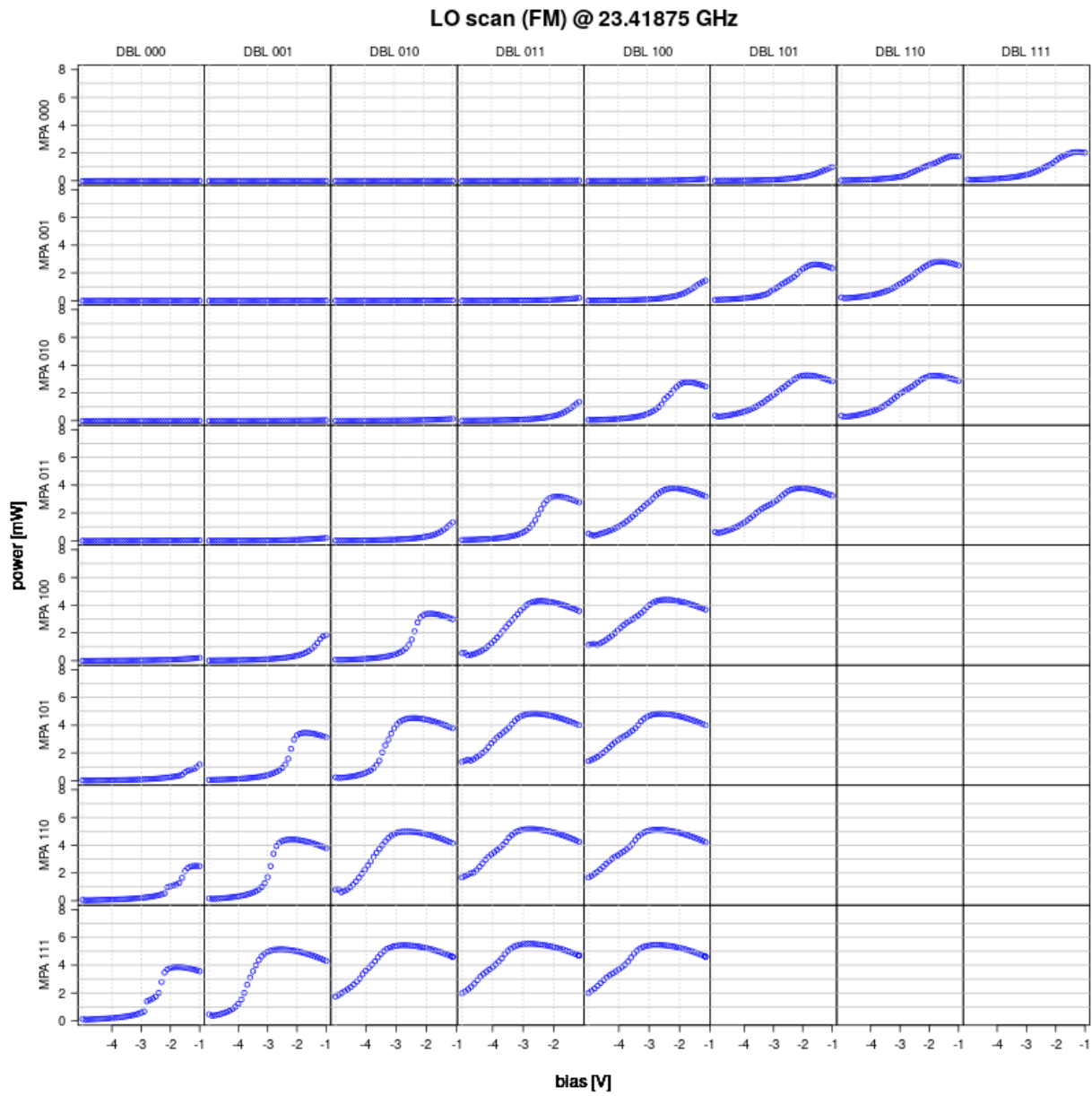
**LO scan (FM) @ 23.3925 GHz**

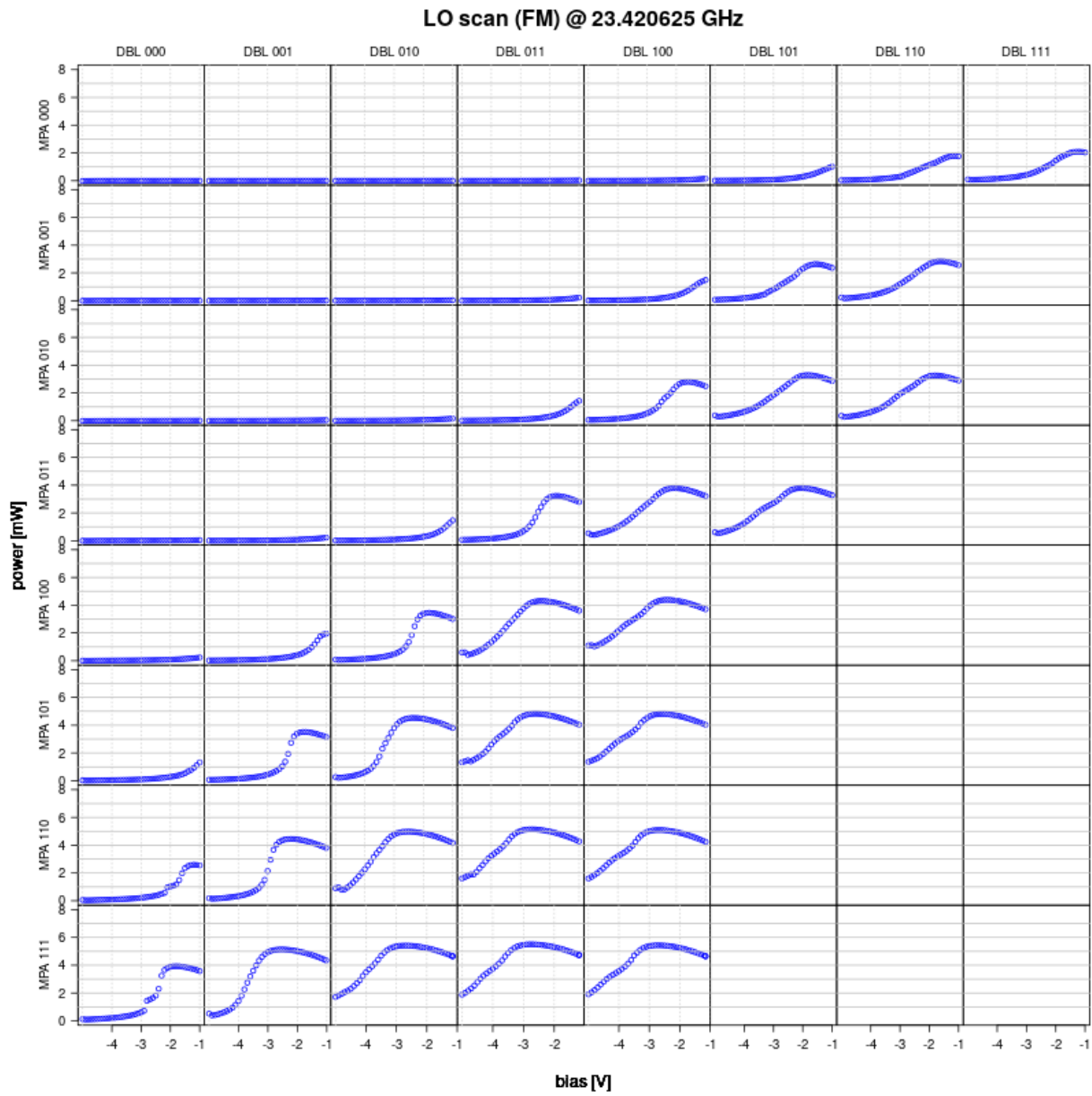




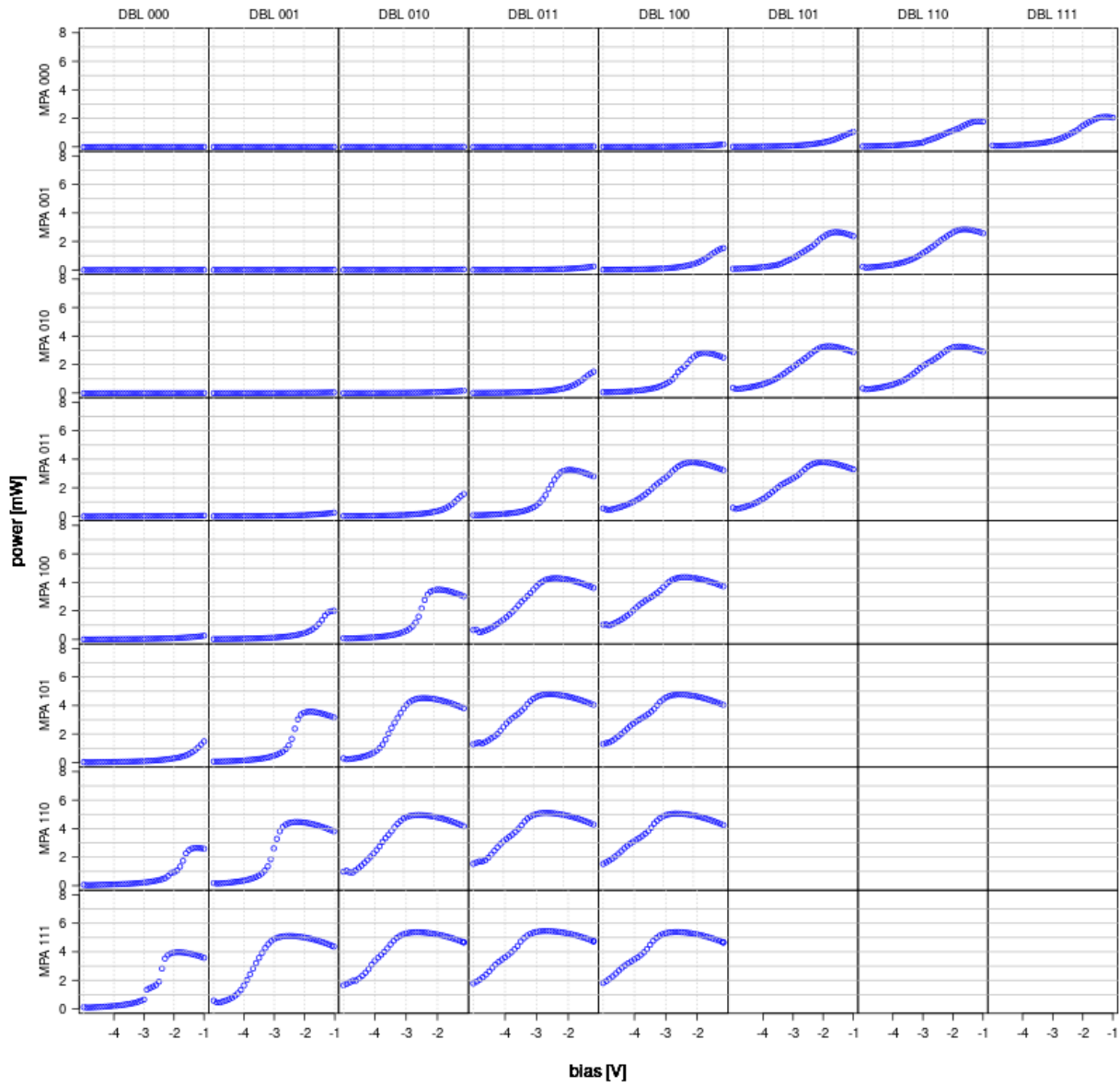
**LO scan (FM) @ 23.416875 GHz**

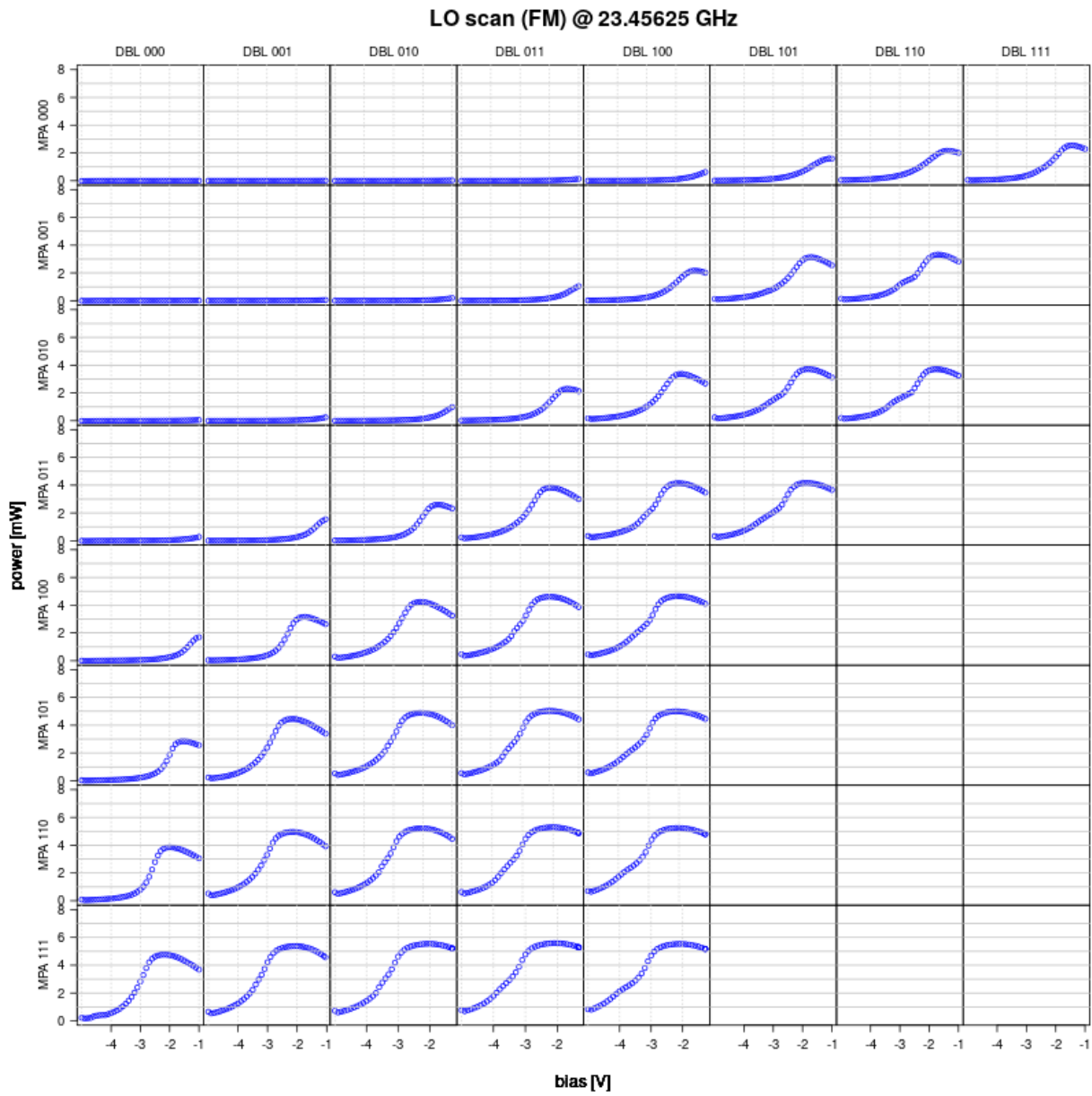






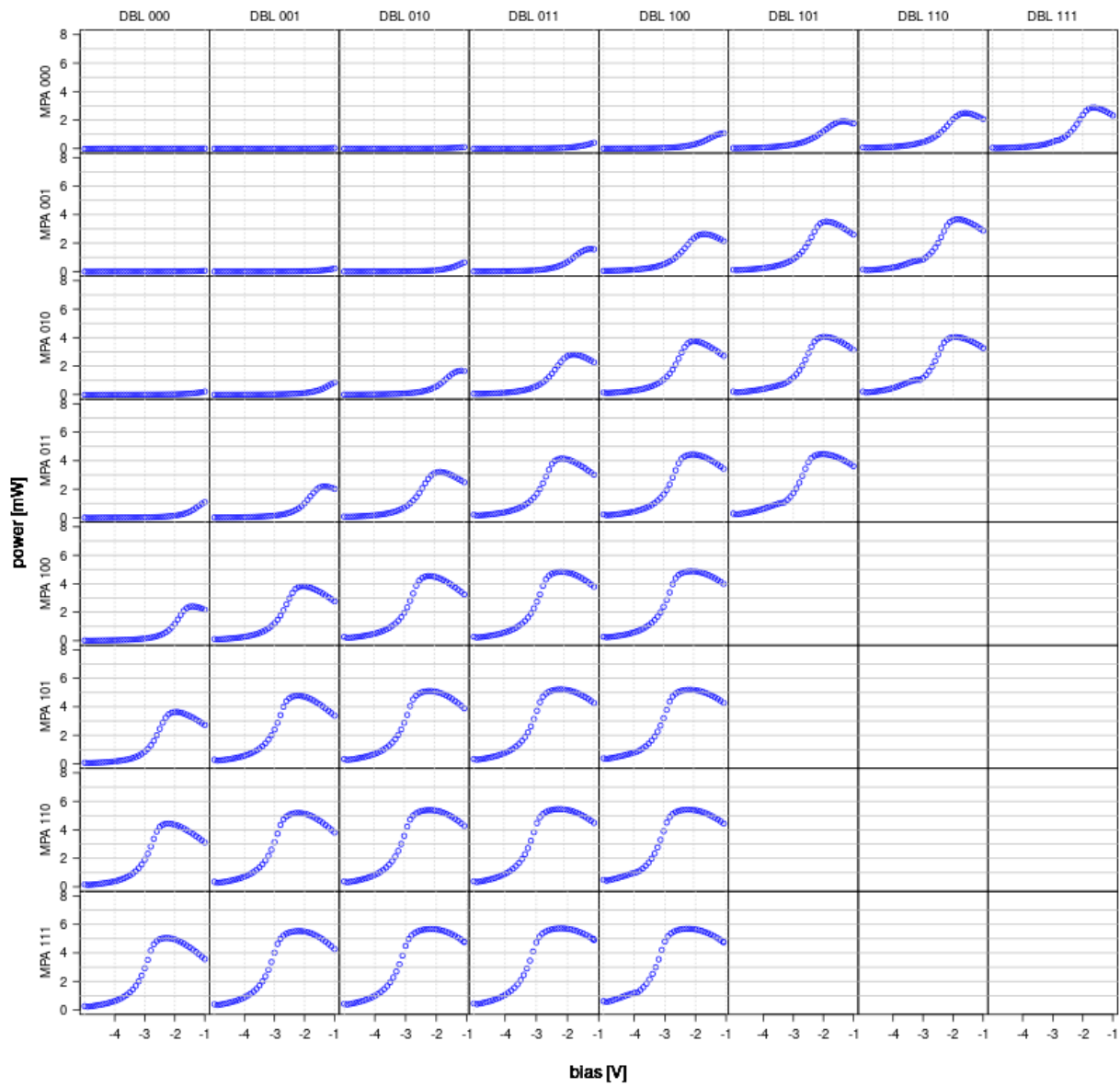
**LO scan (FM) @ 23.4225 GHz**

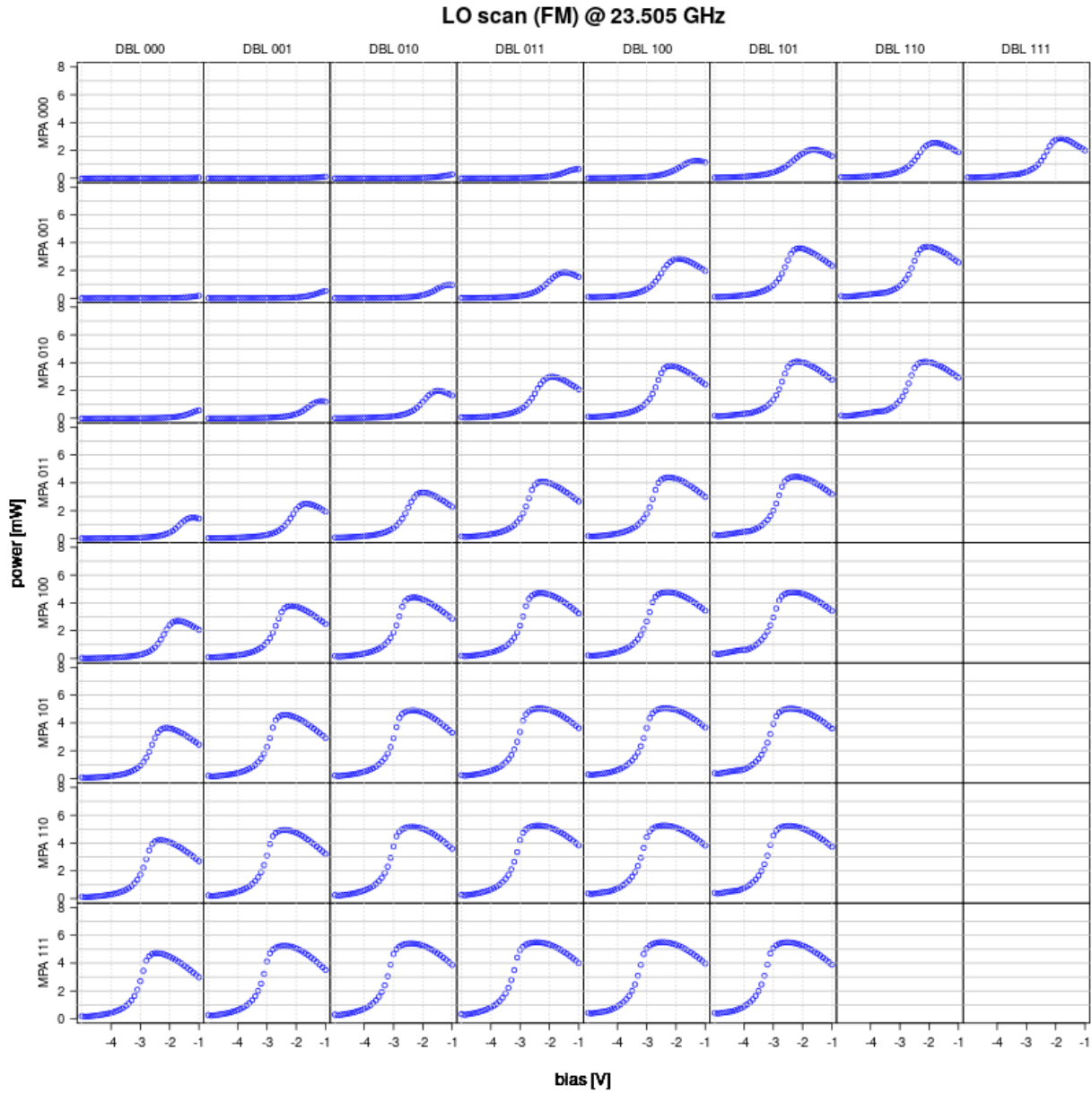


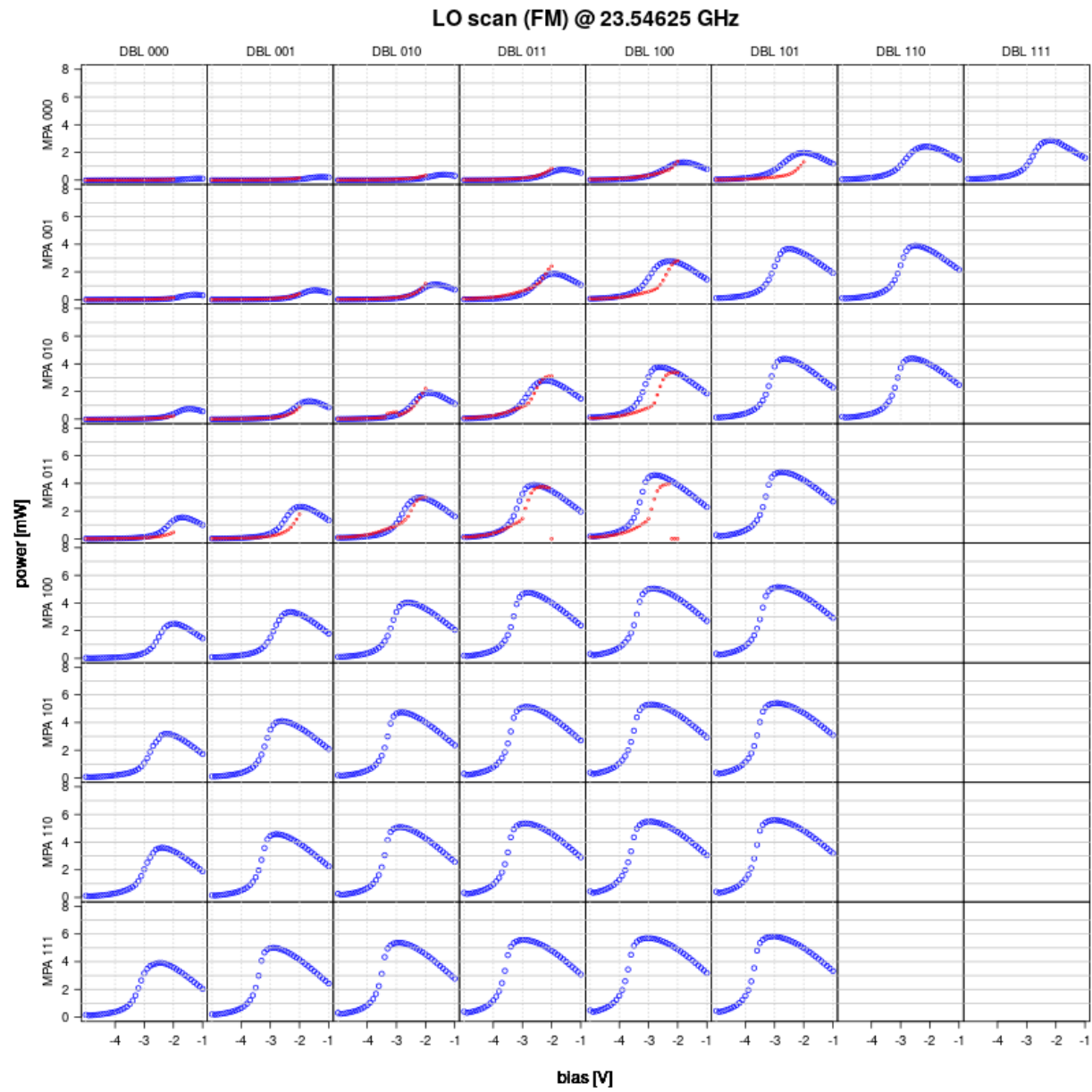


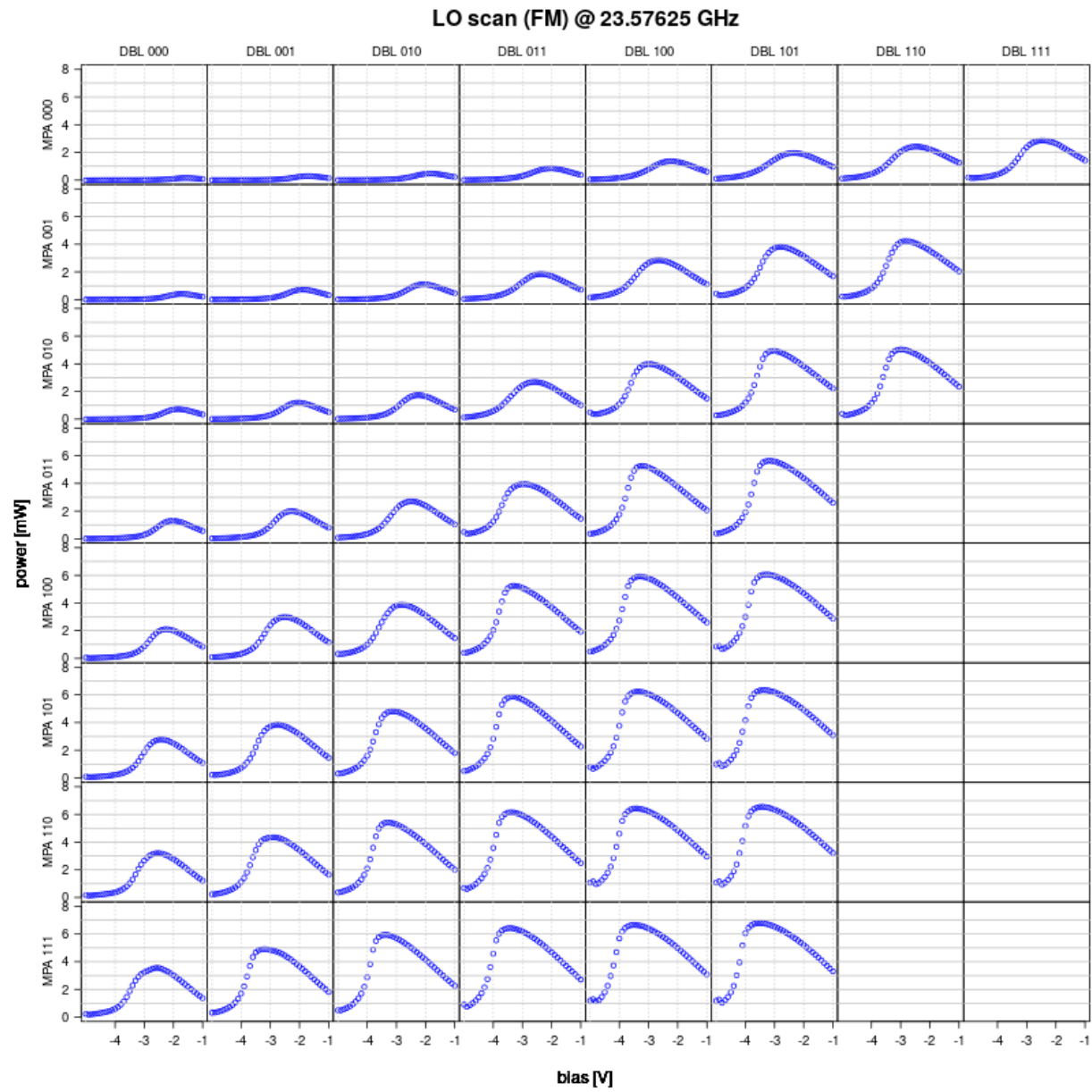


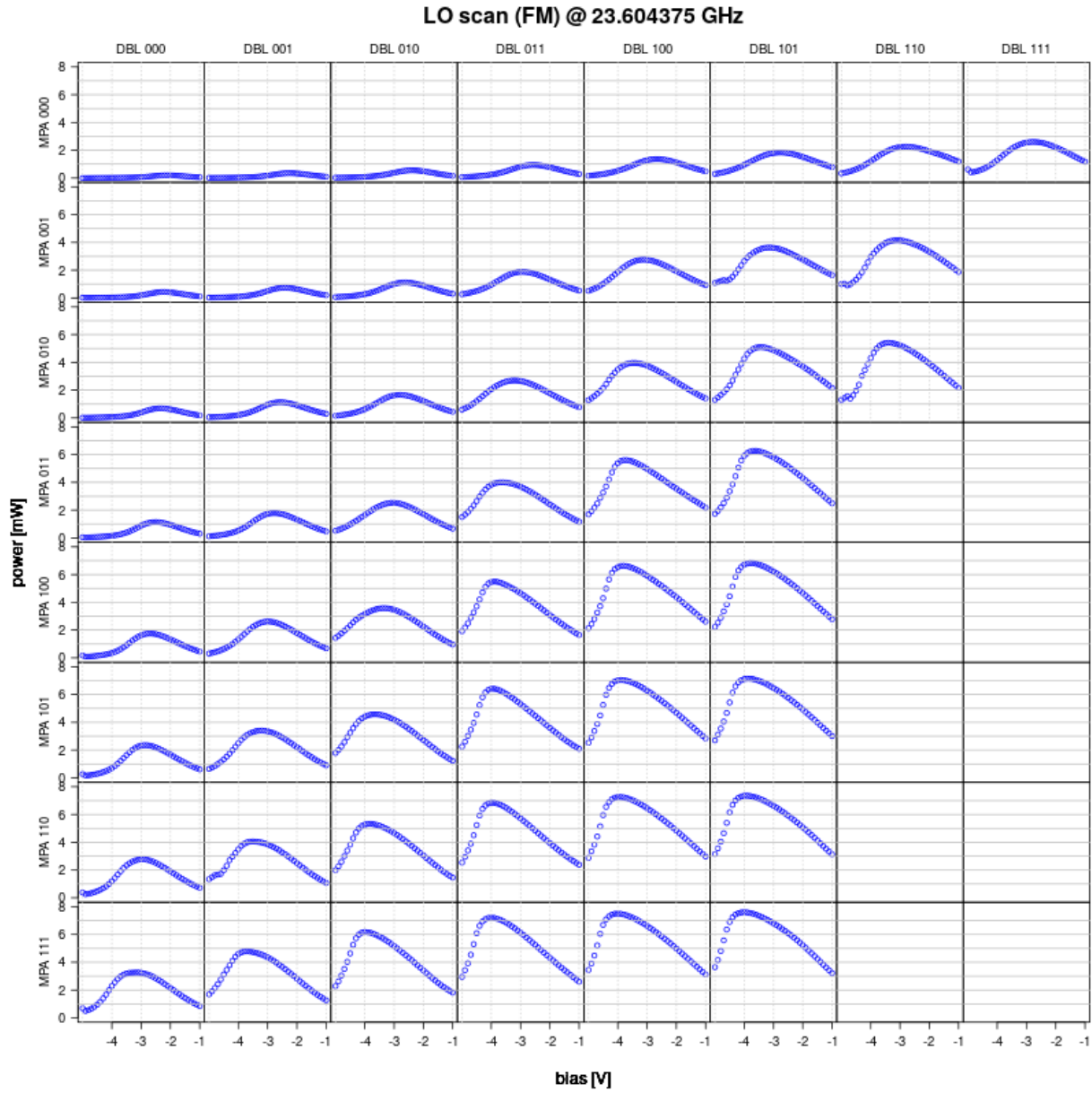
**LO scan (FM) @ 23.480625 GHz**

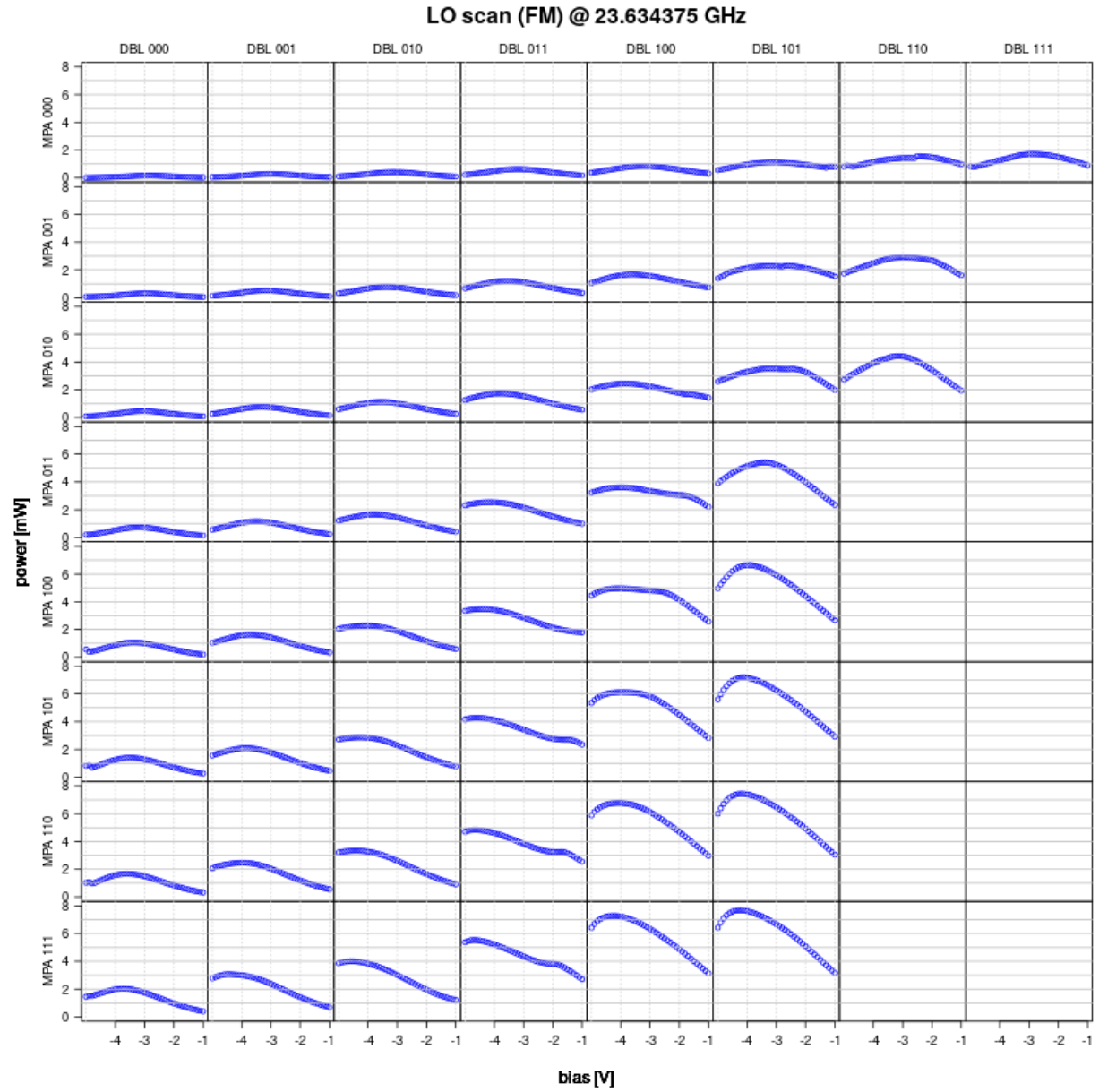




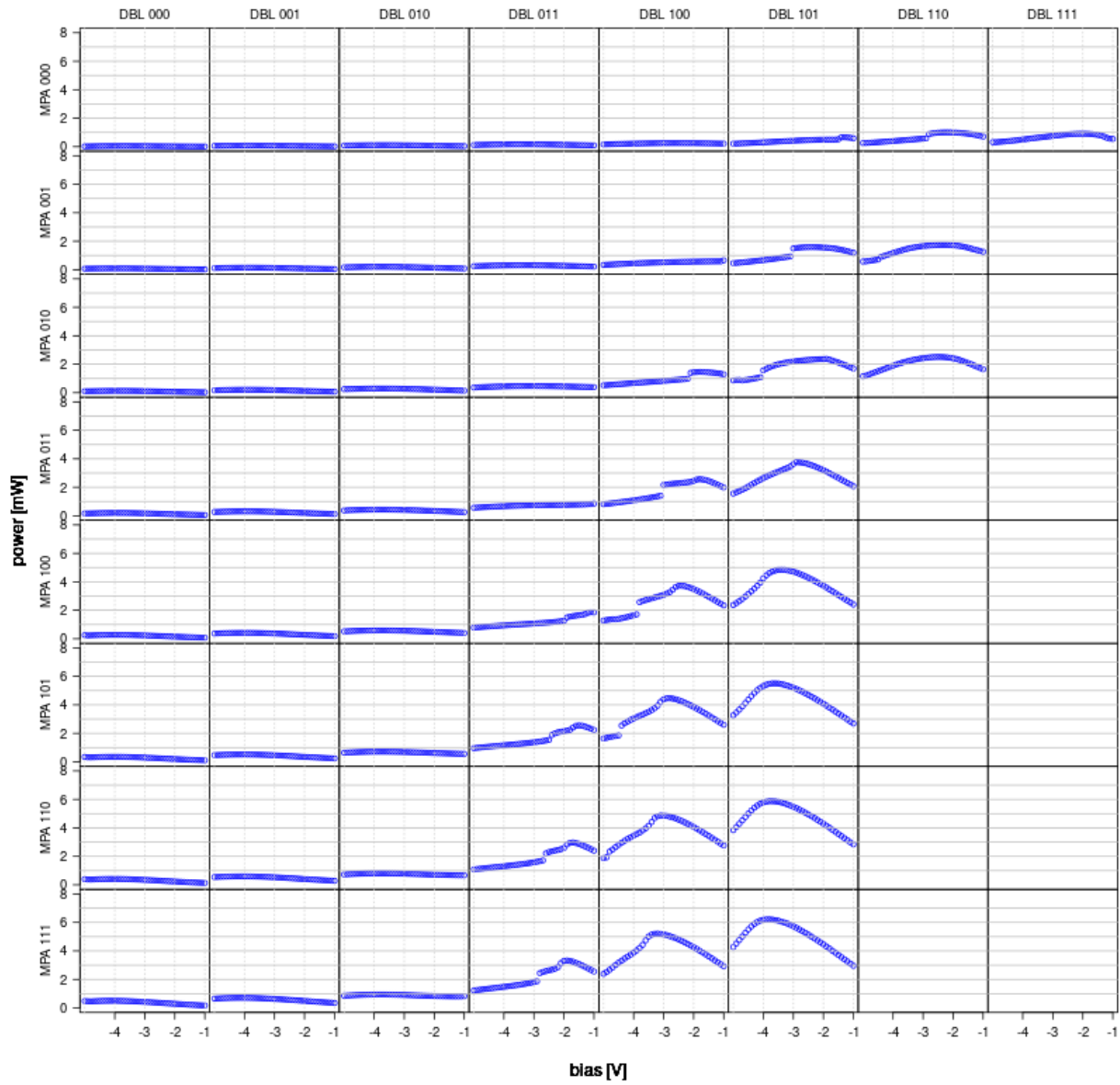


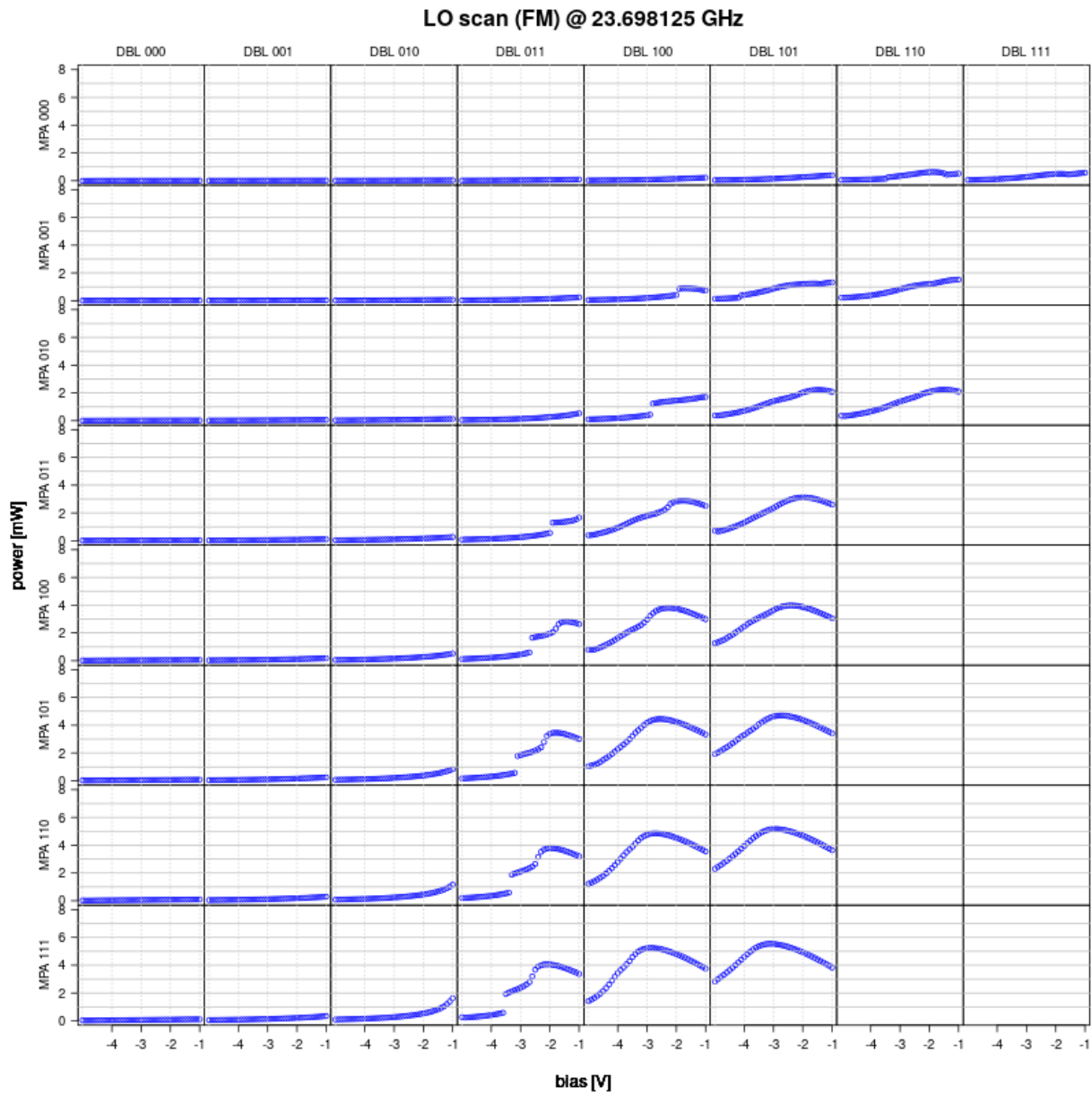






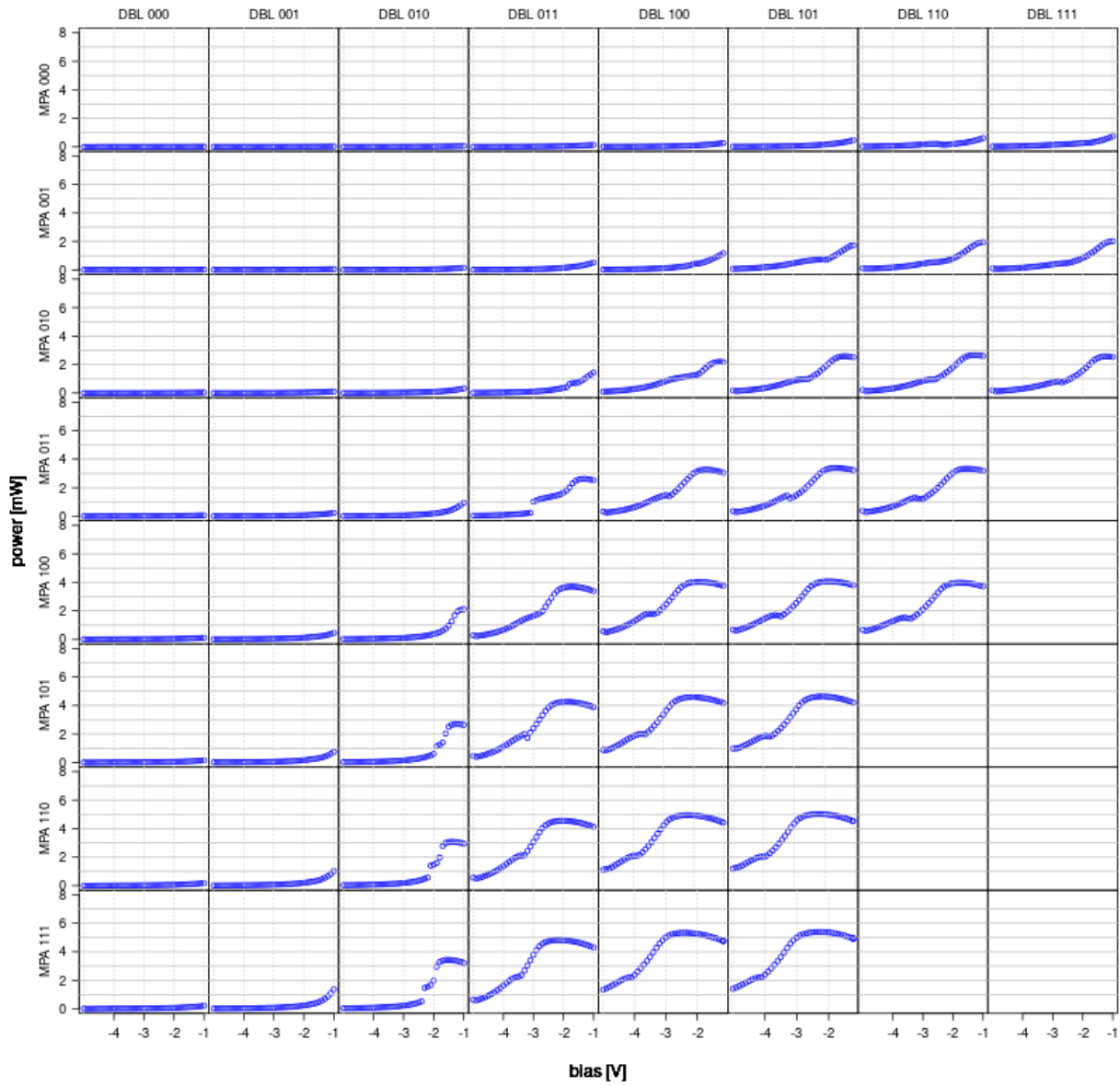
**LO scan (FM) @ 23.6625 GHz**

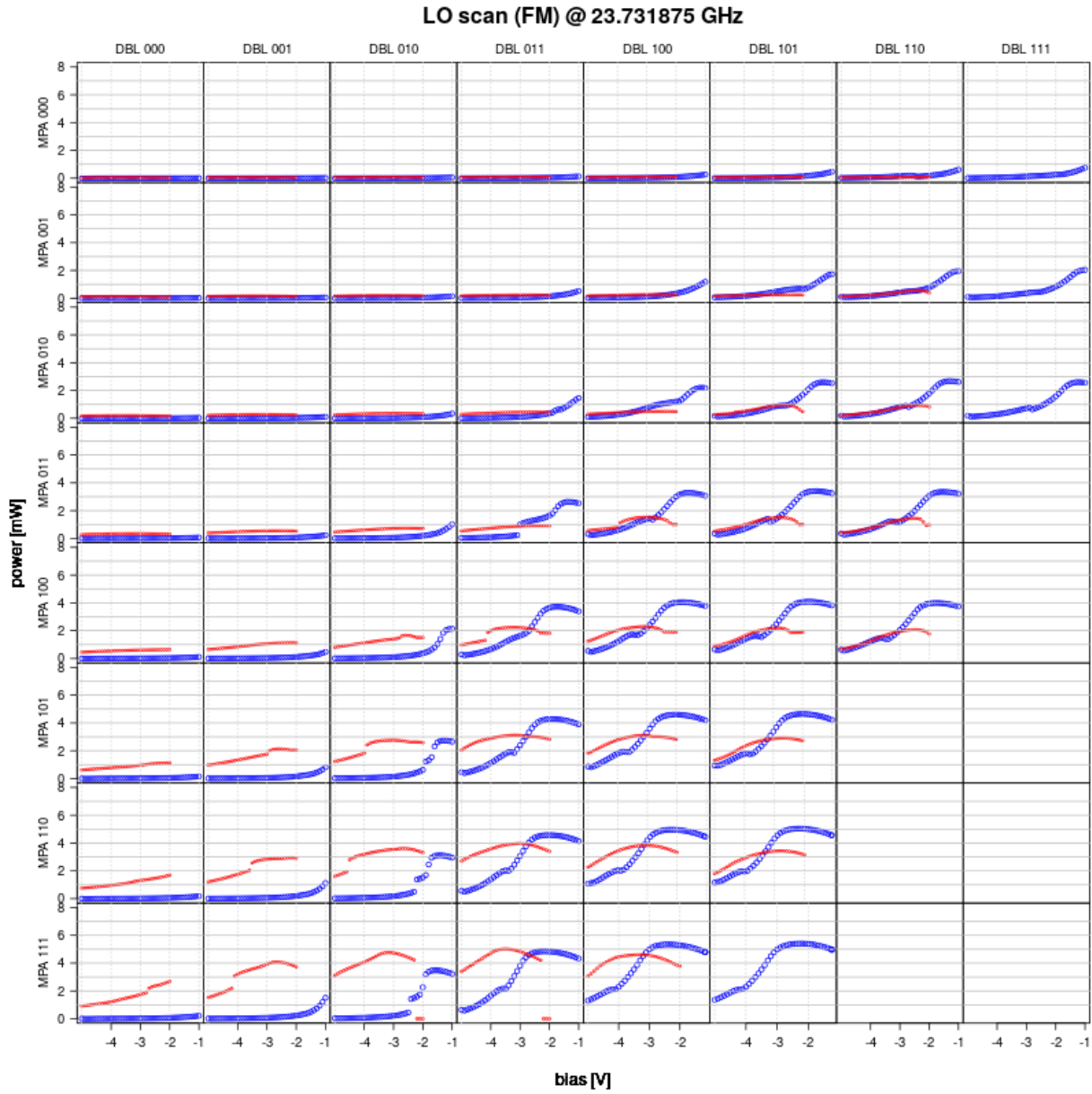


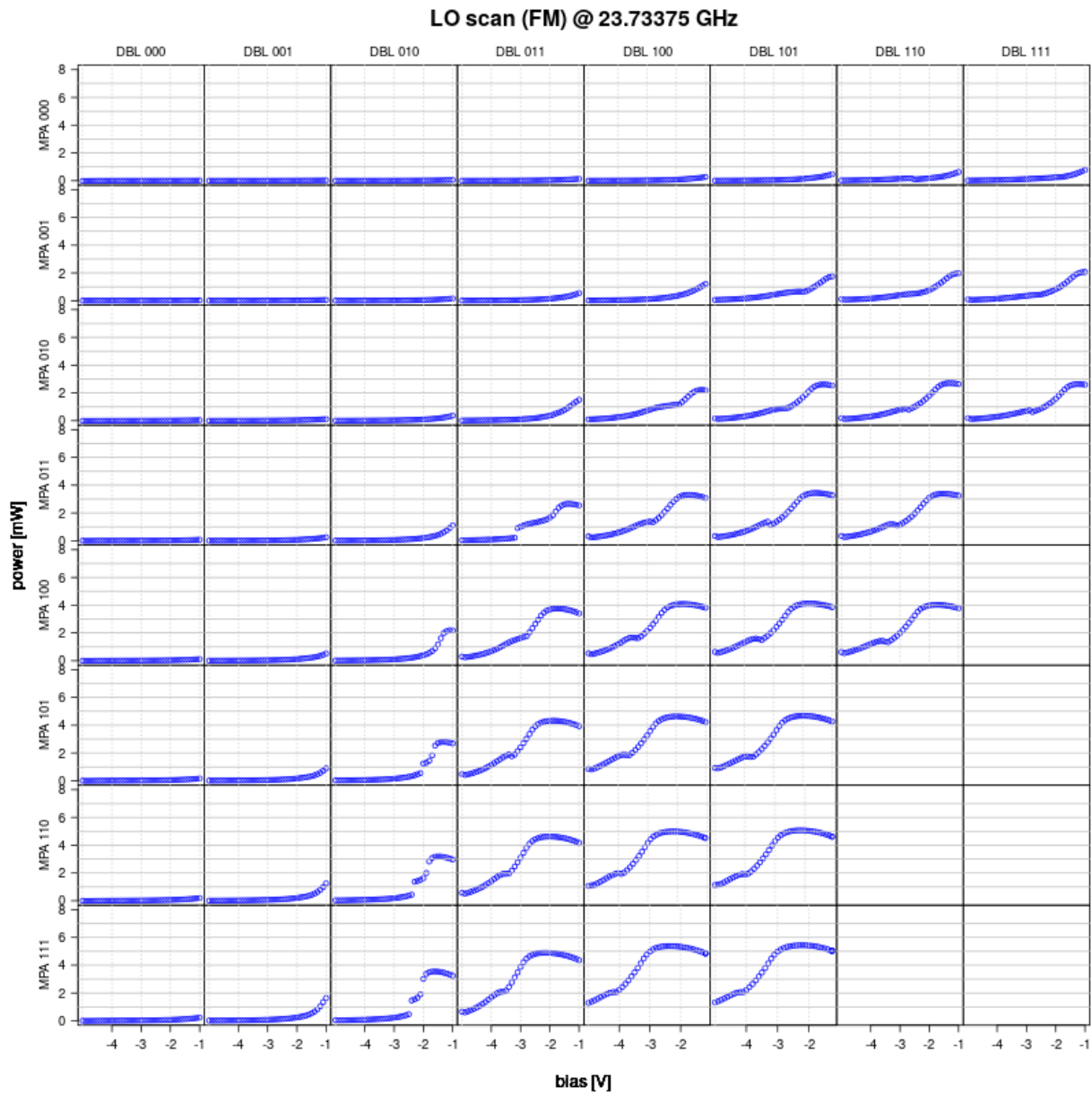


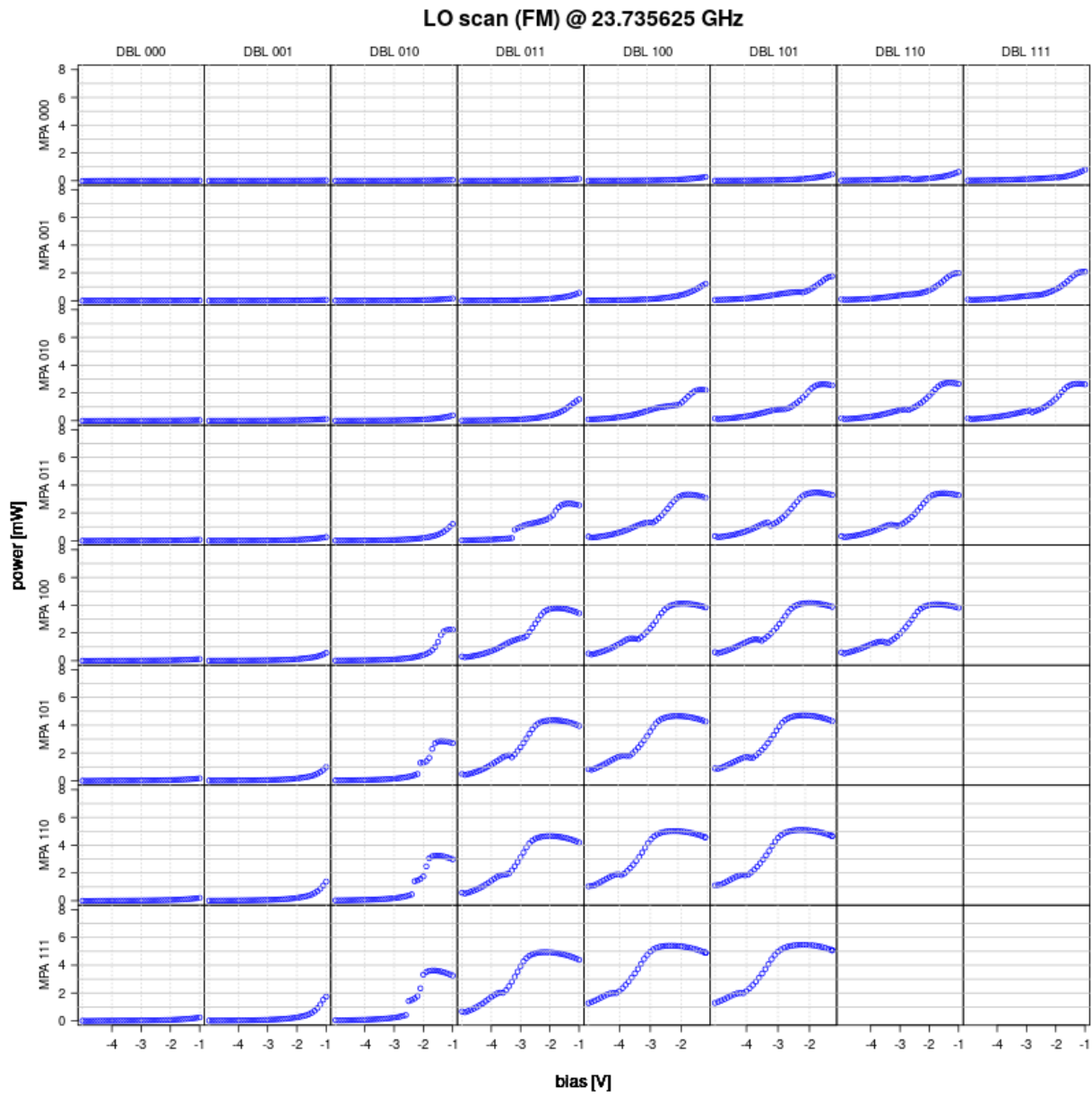


**LO scan (FM) @ 23.73 GHz**

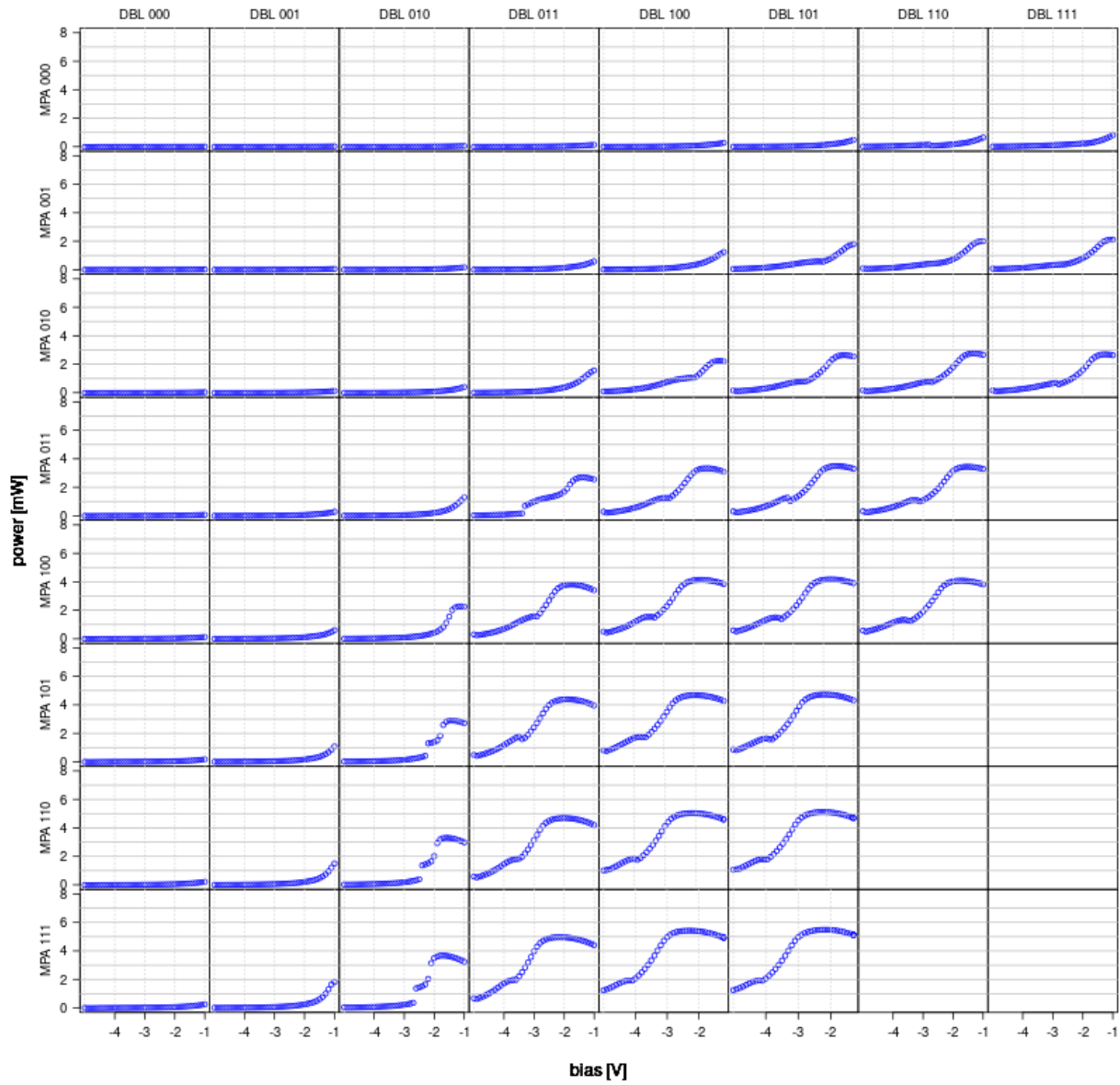


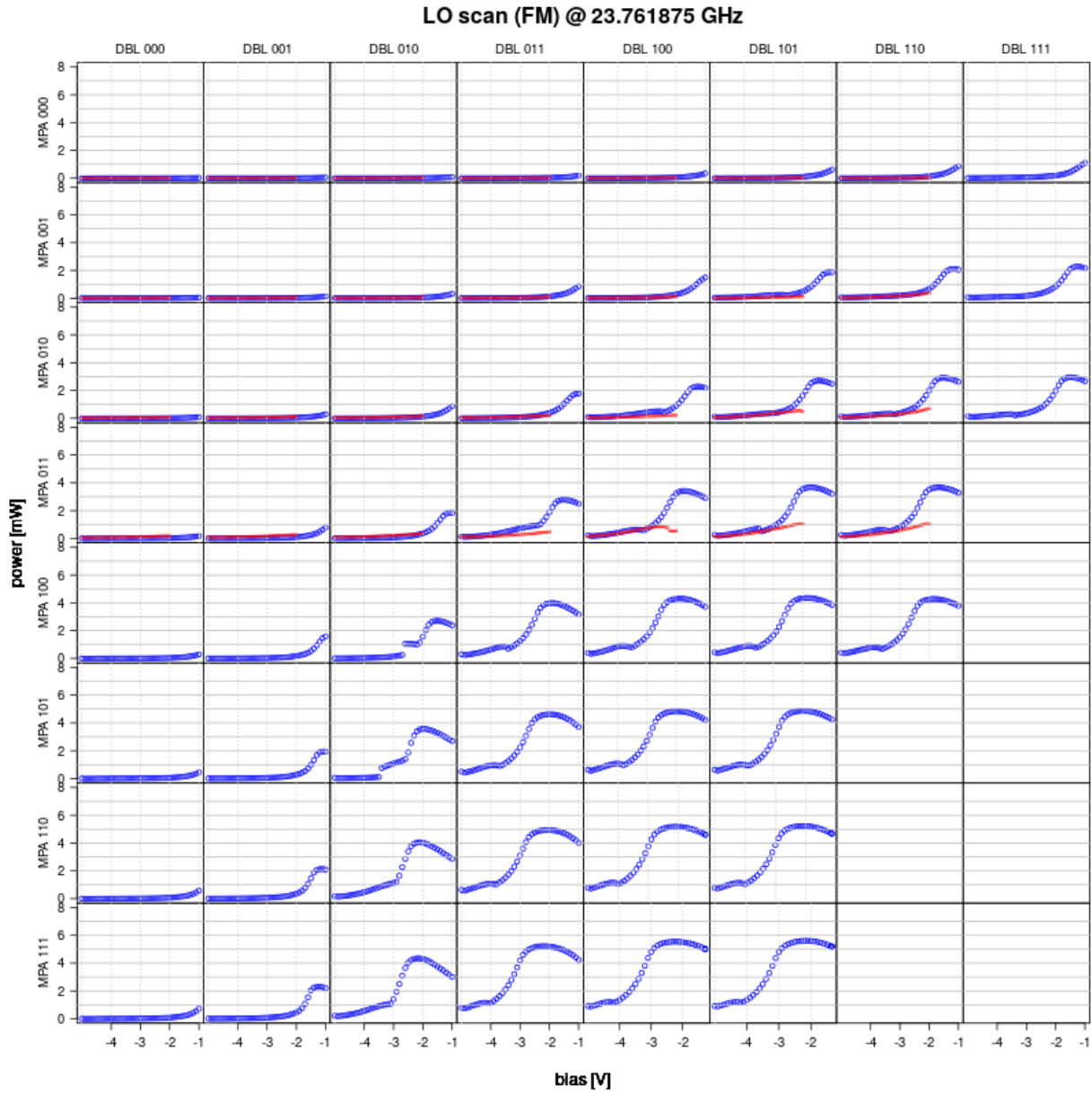


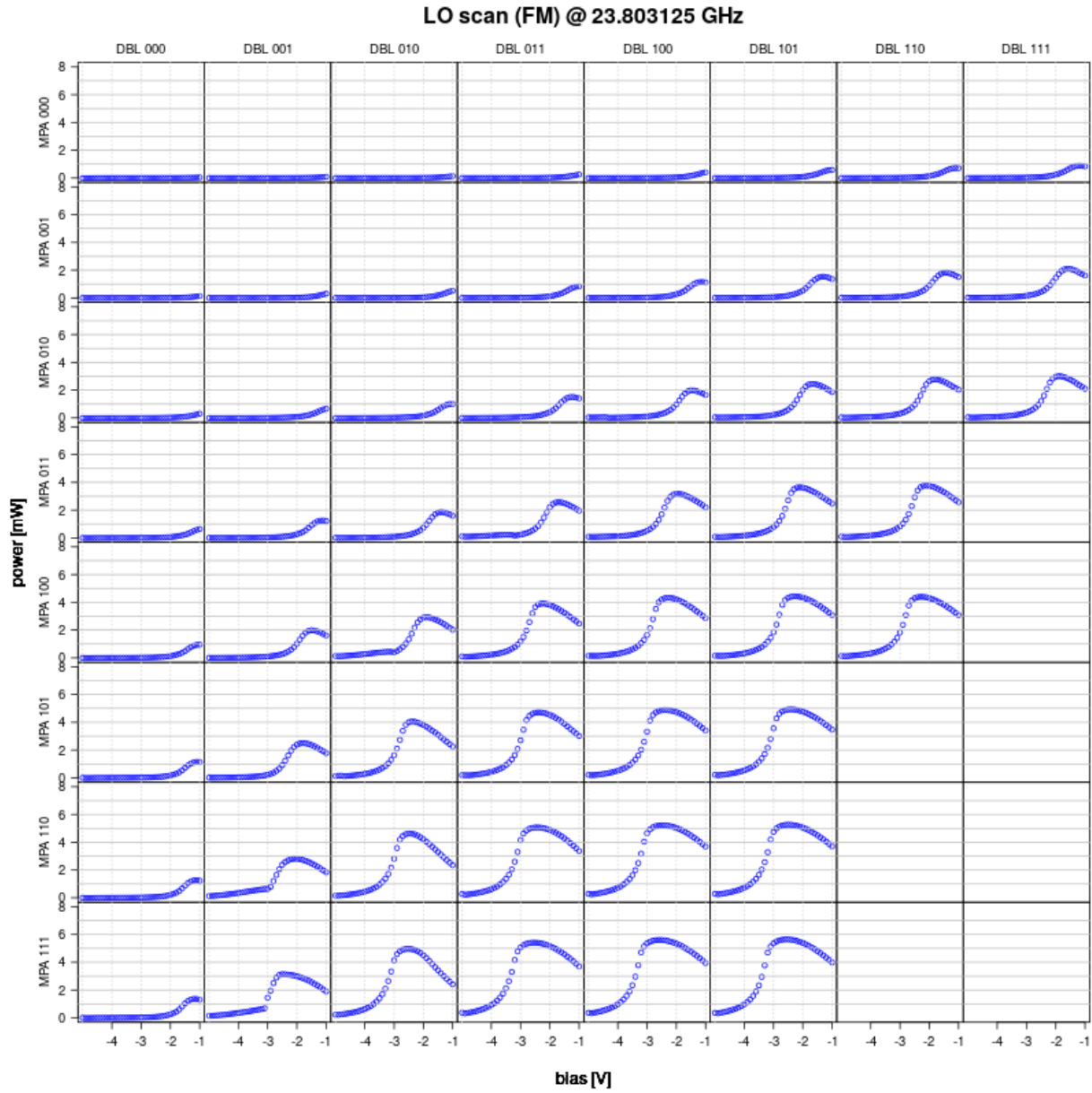




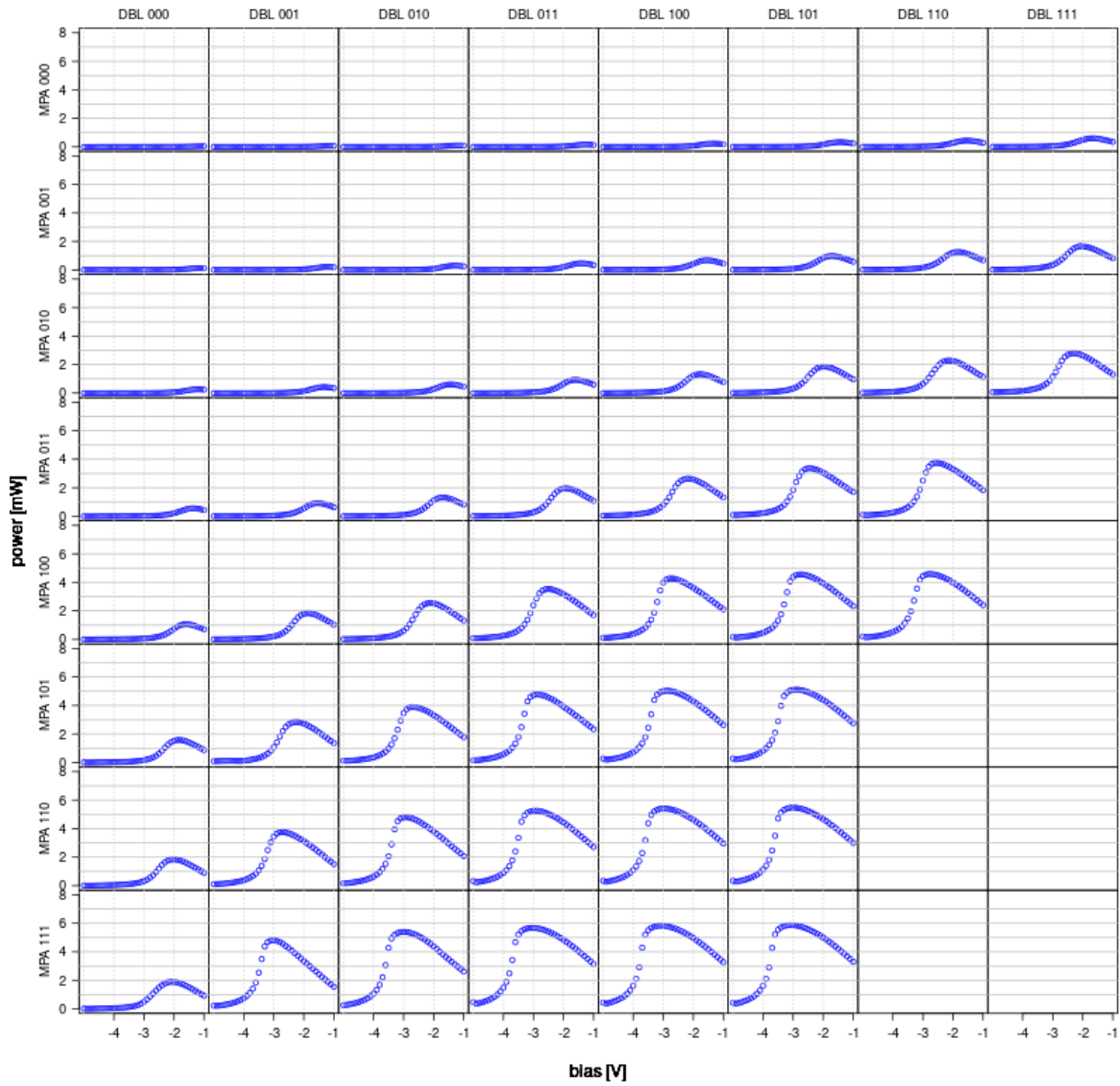
**LO scan (FM) @ 23.7375 GHz**



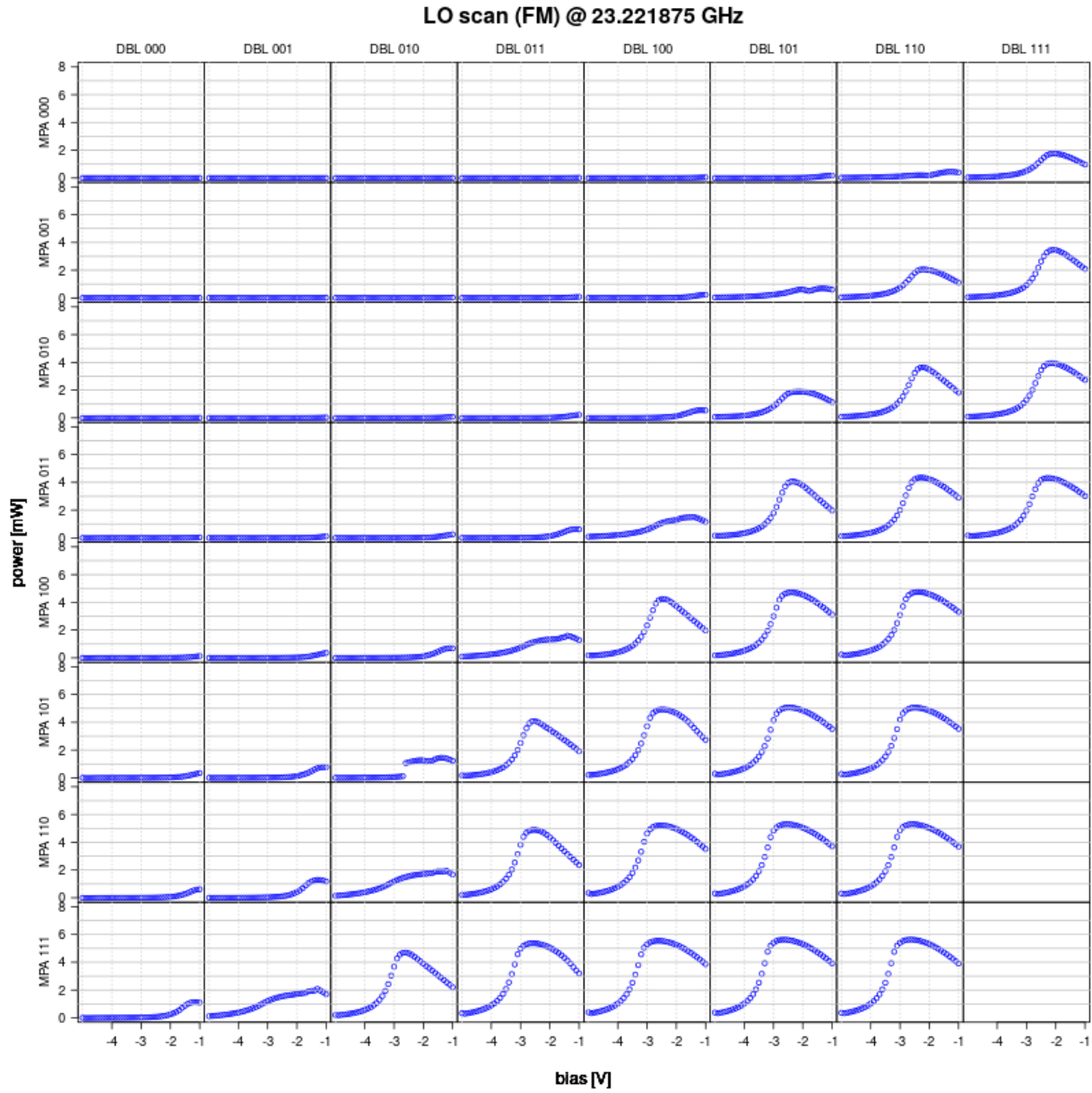




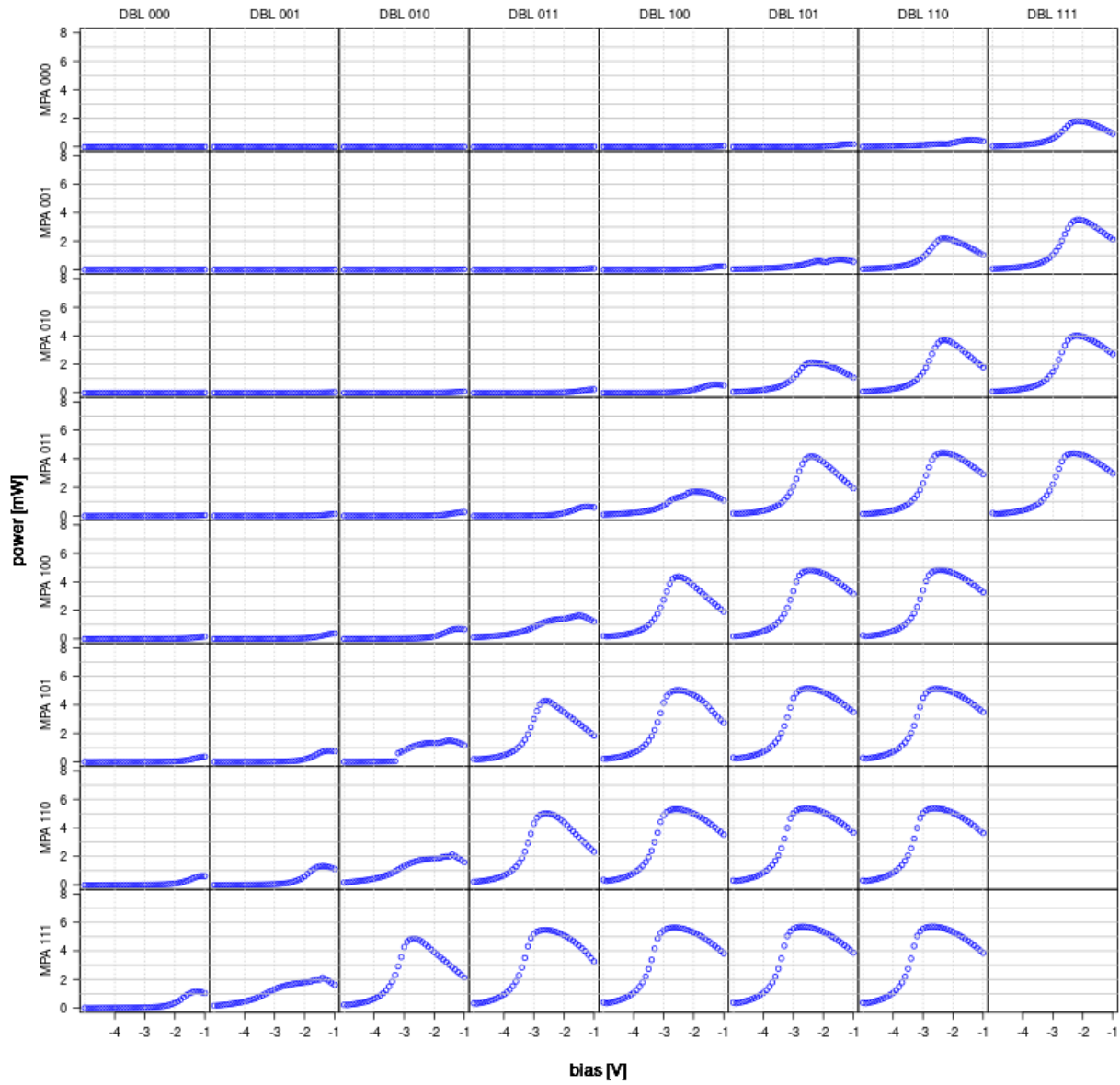
**LO scan (FM) @ 23.8425 GHz**

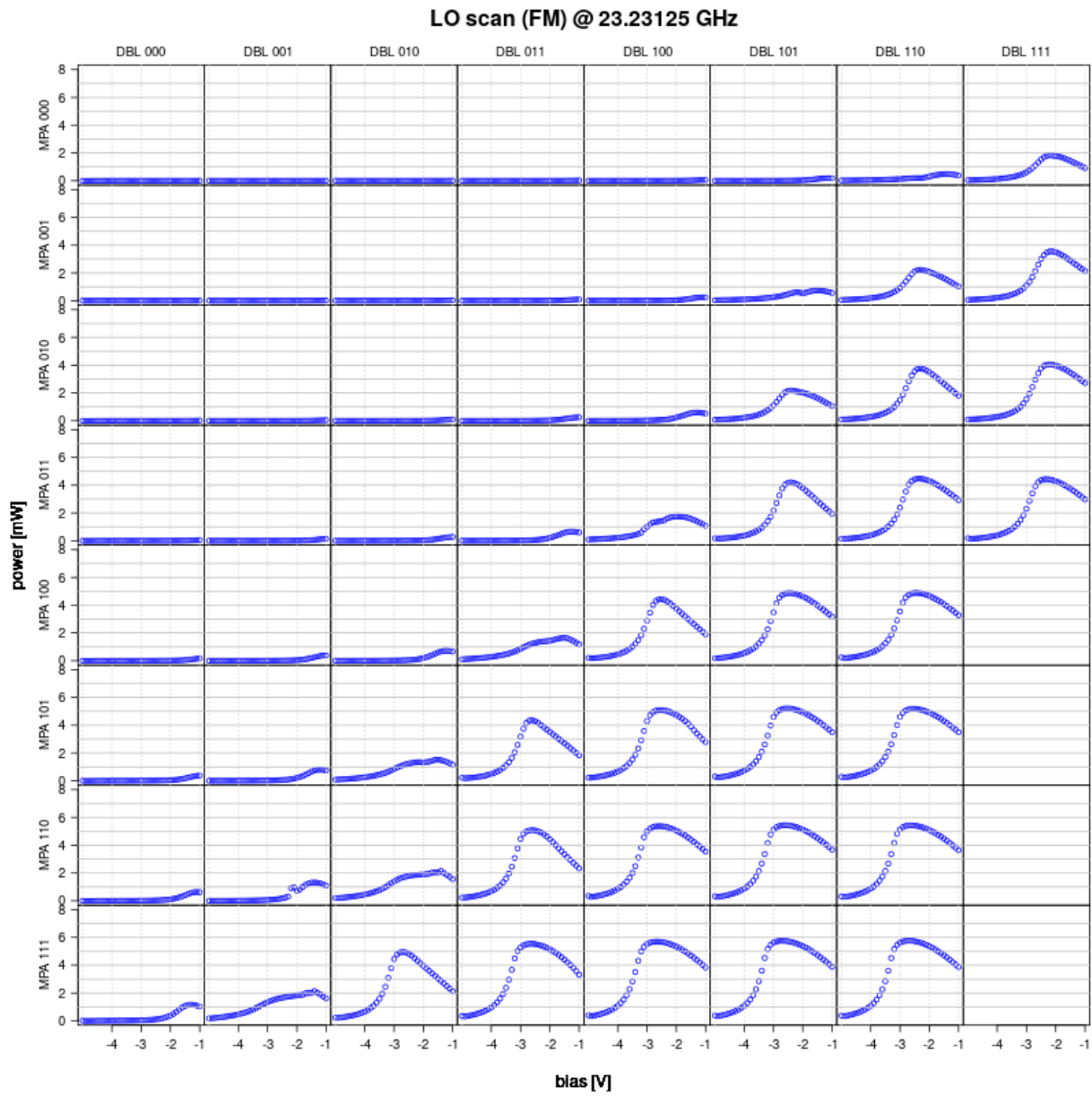


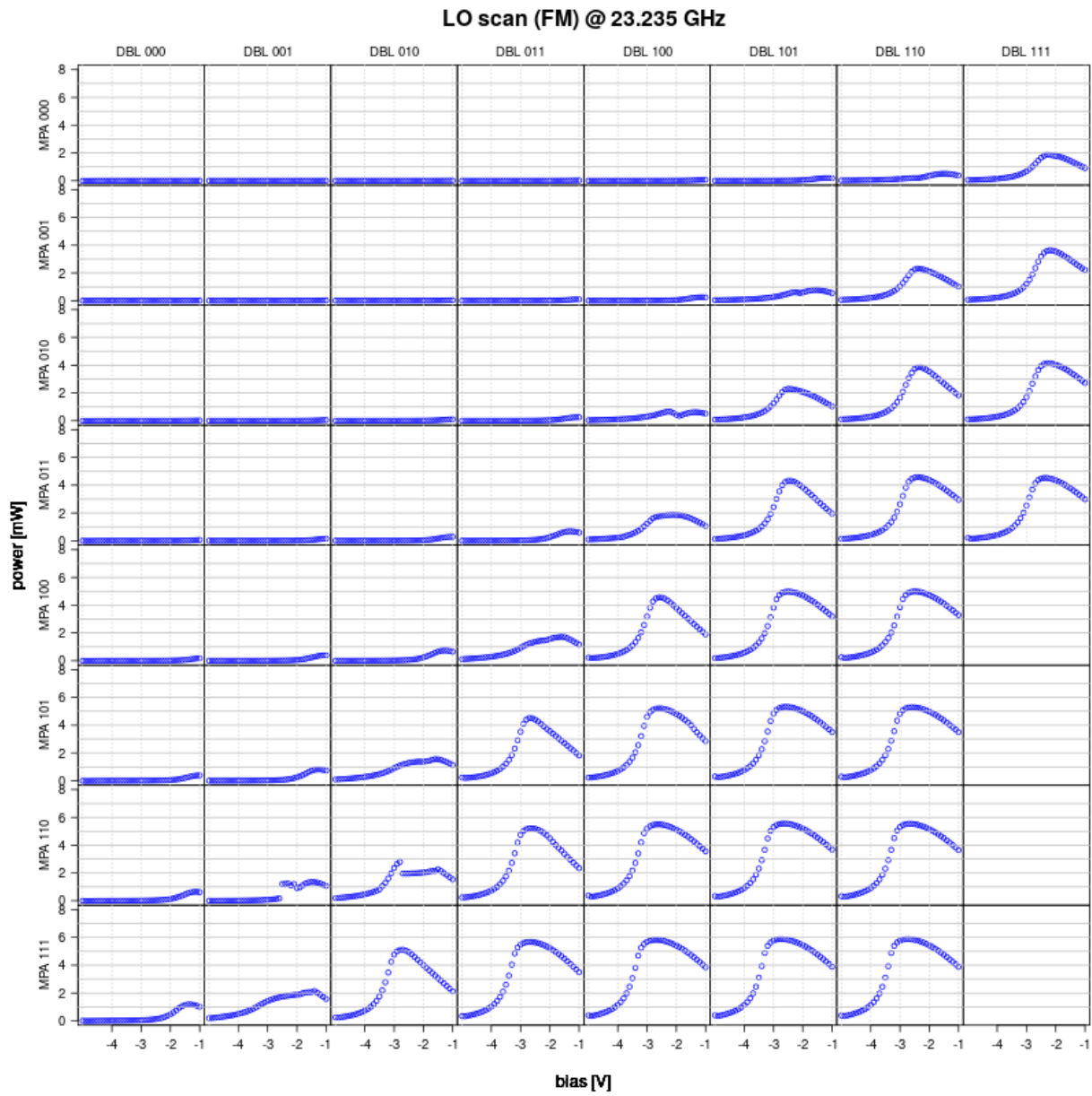


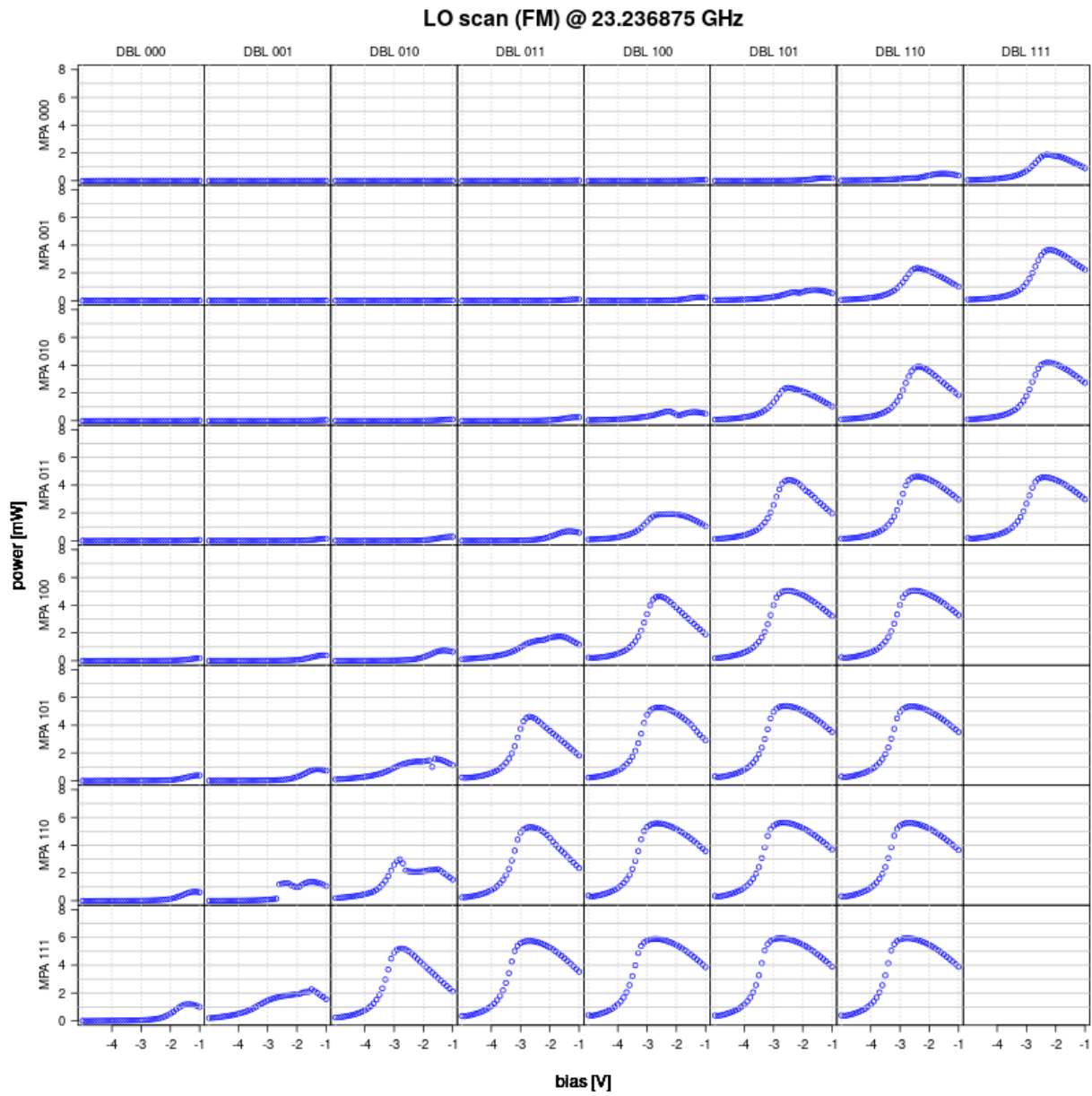


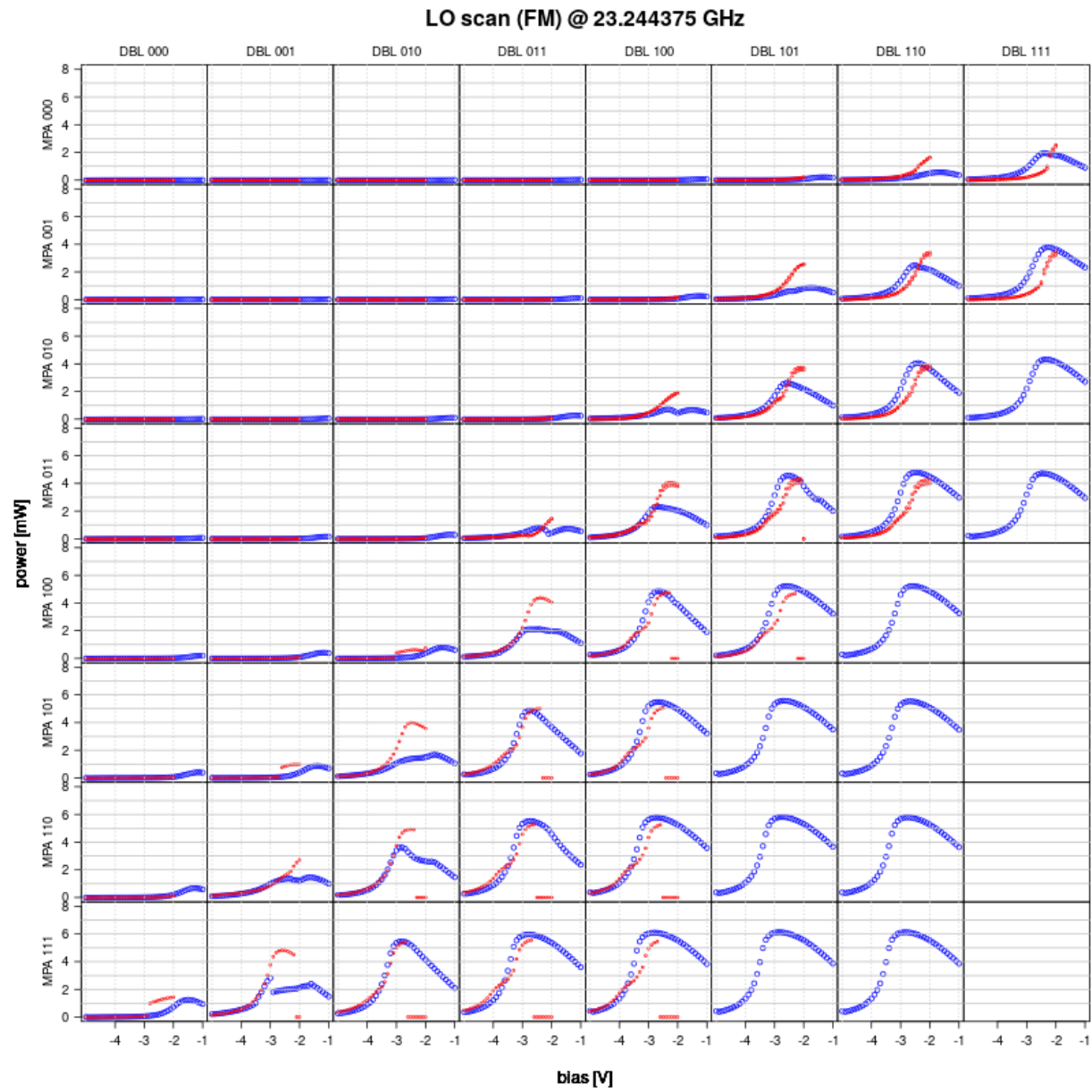
**LO scan (FM) @ 23.229375 GHz**

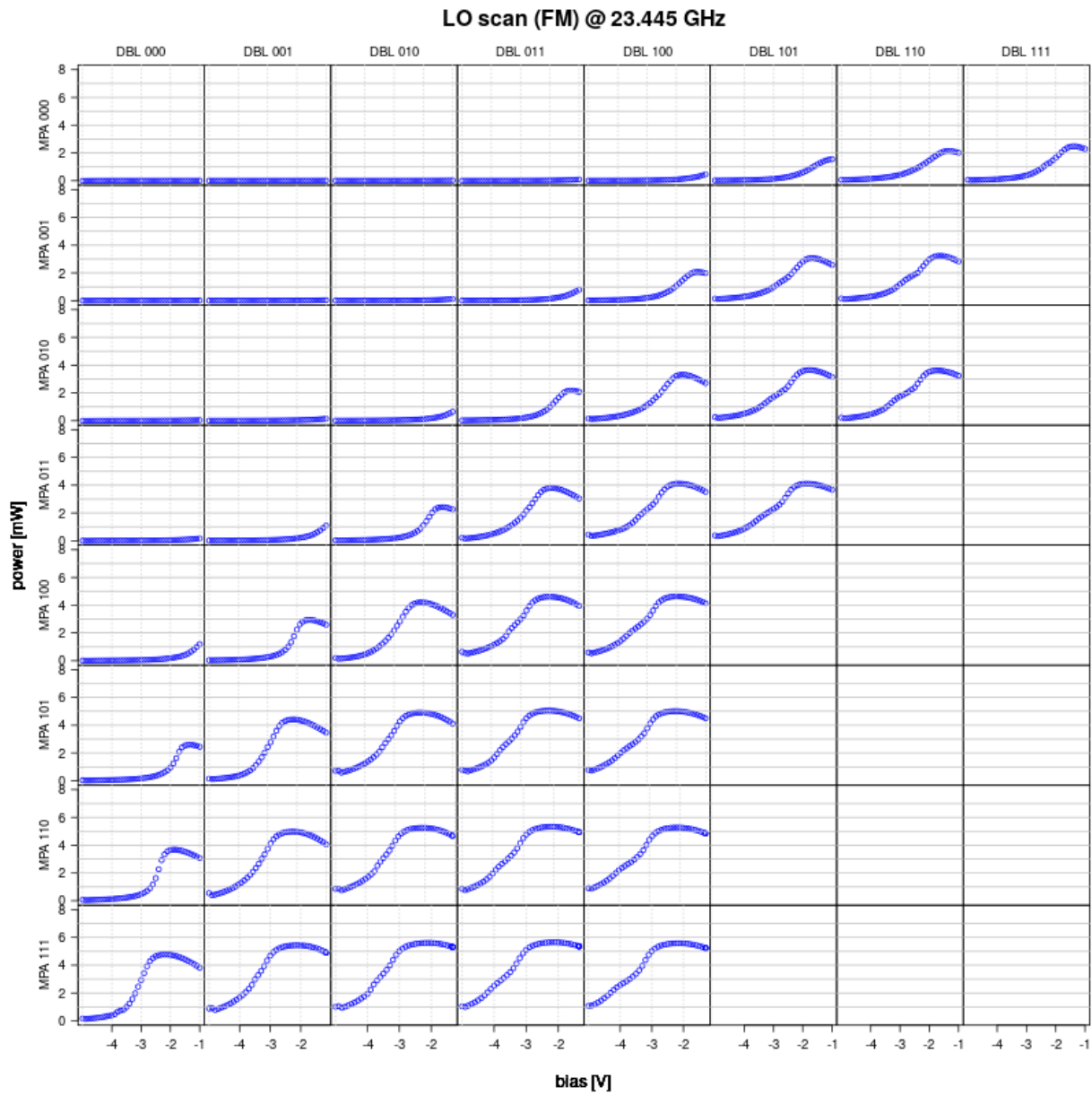




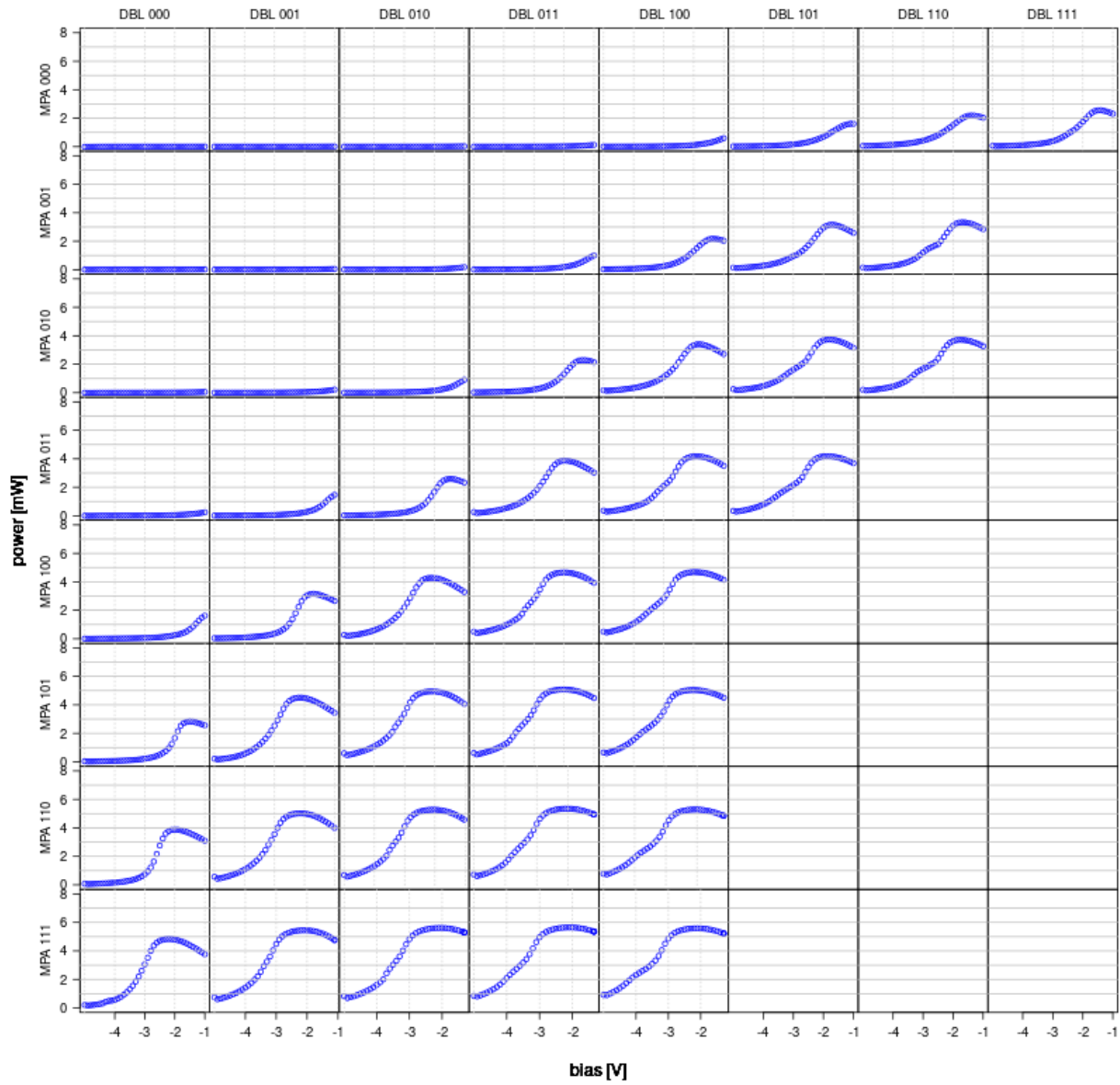




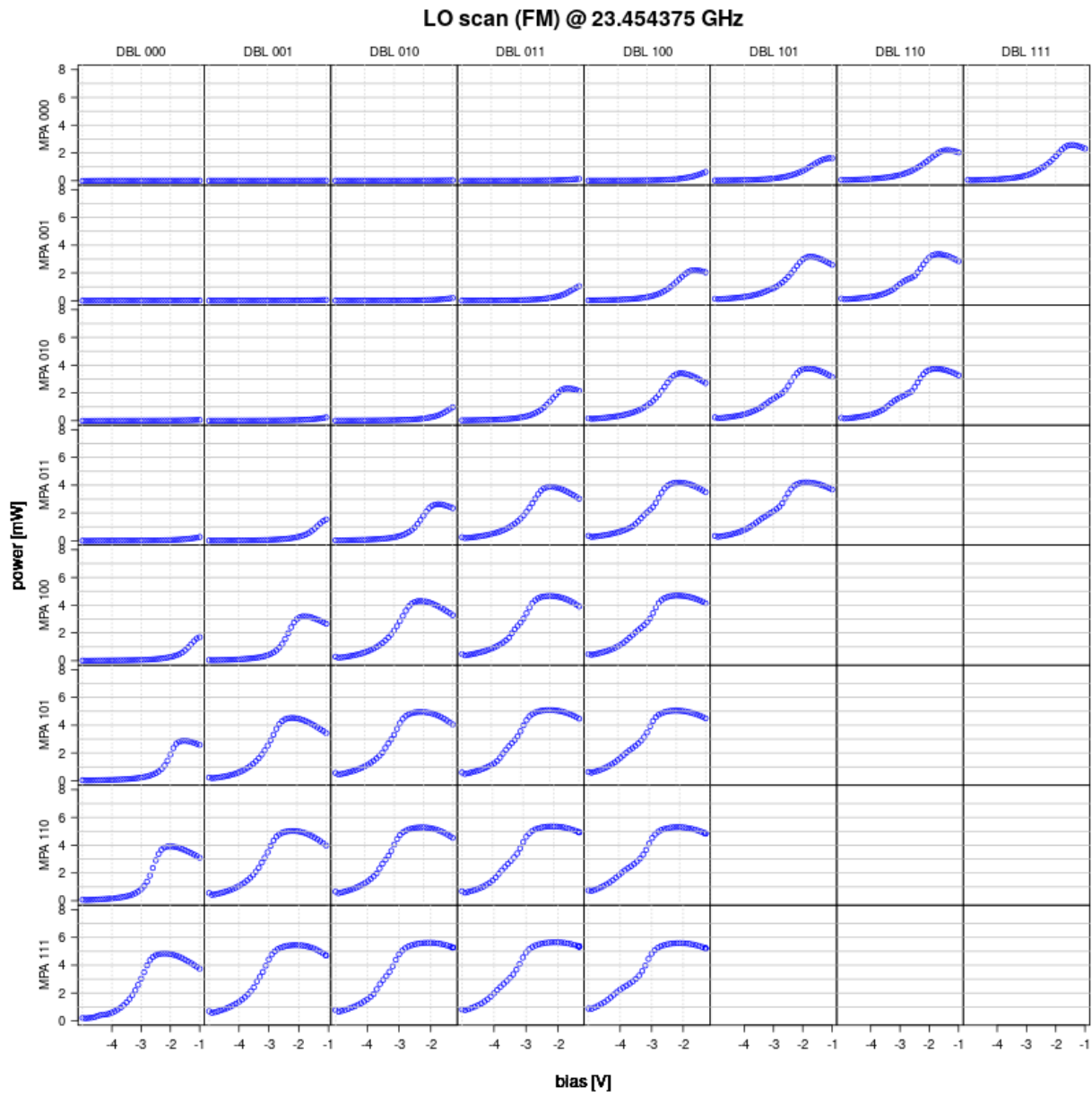


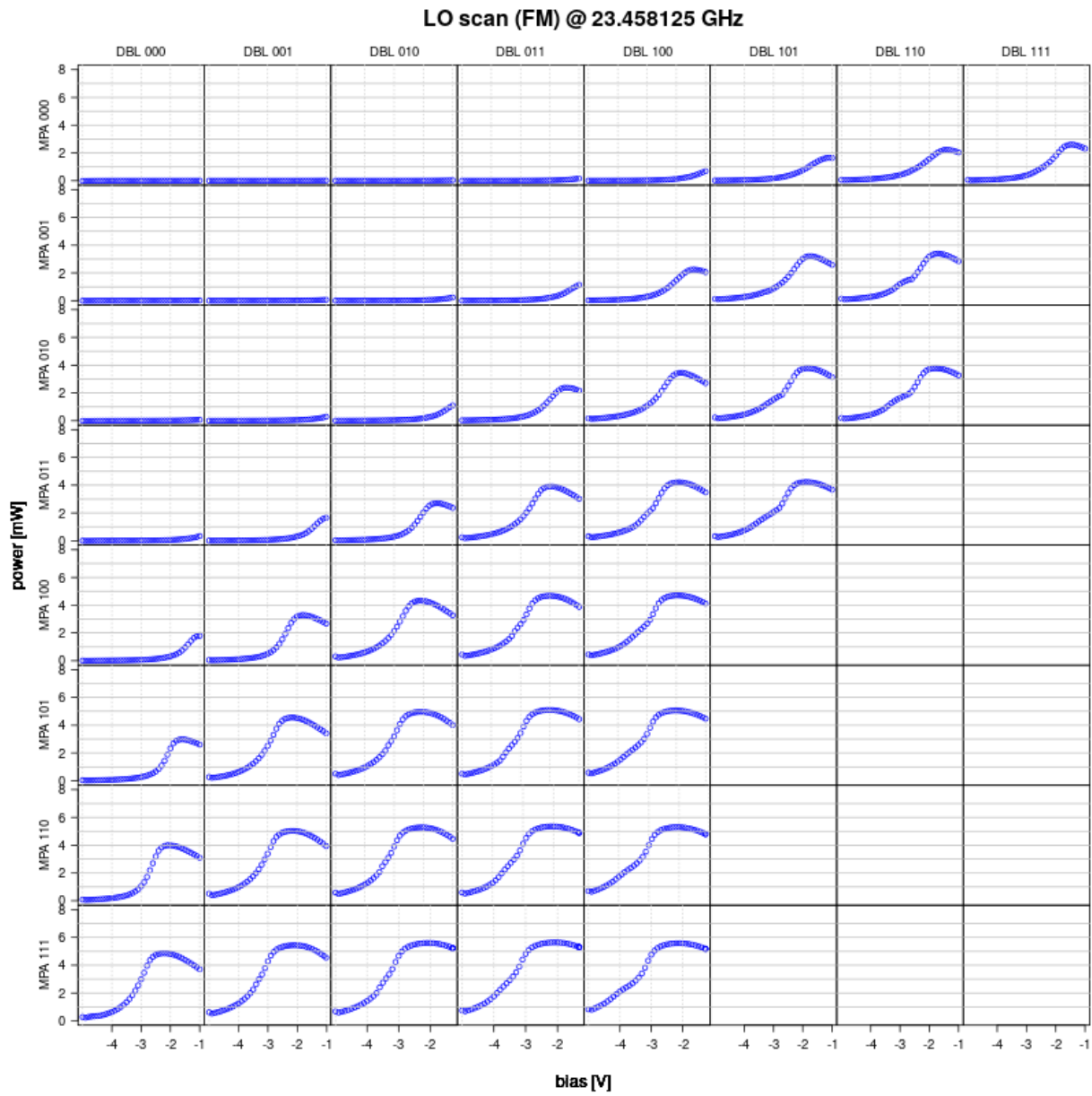


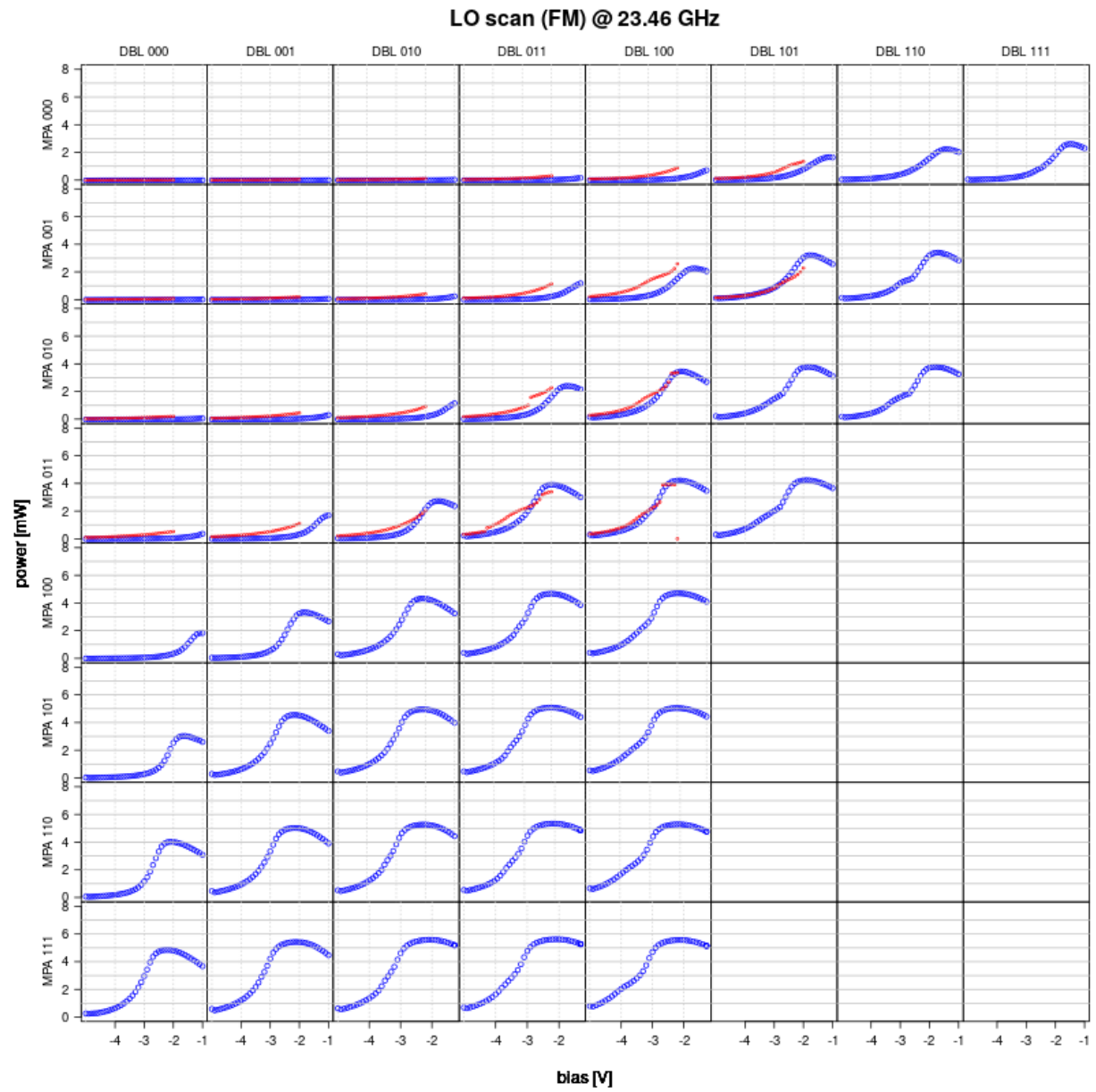
**LO scan (FM) @ 23.4525 GHz**



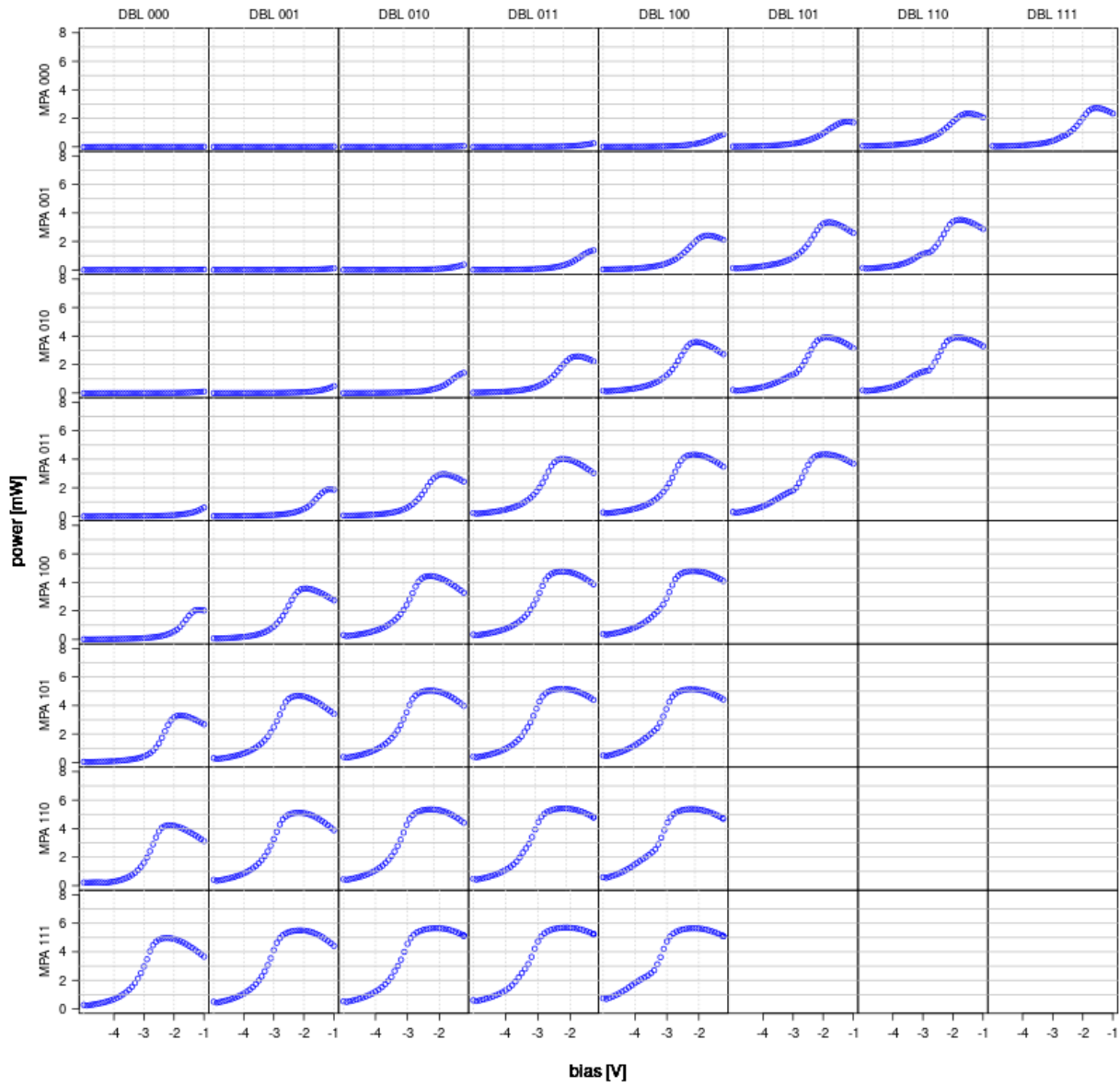


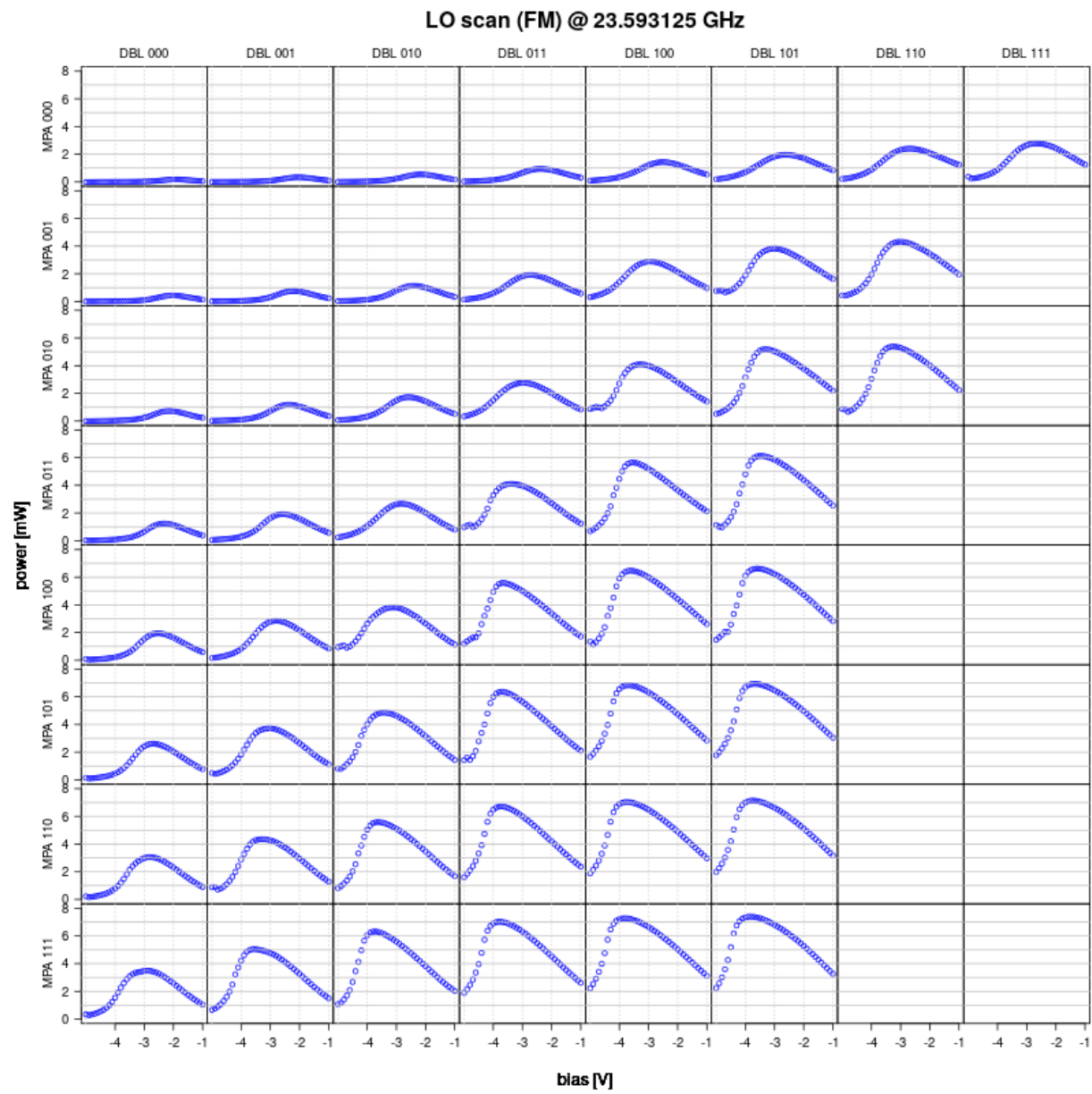


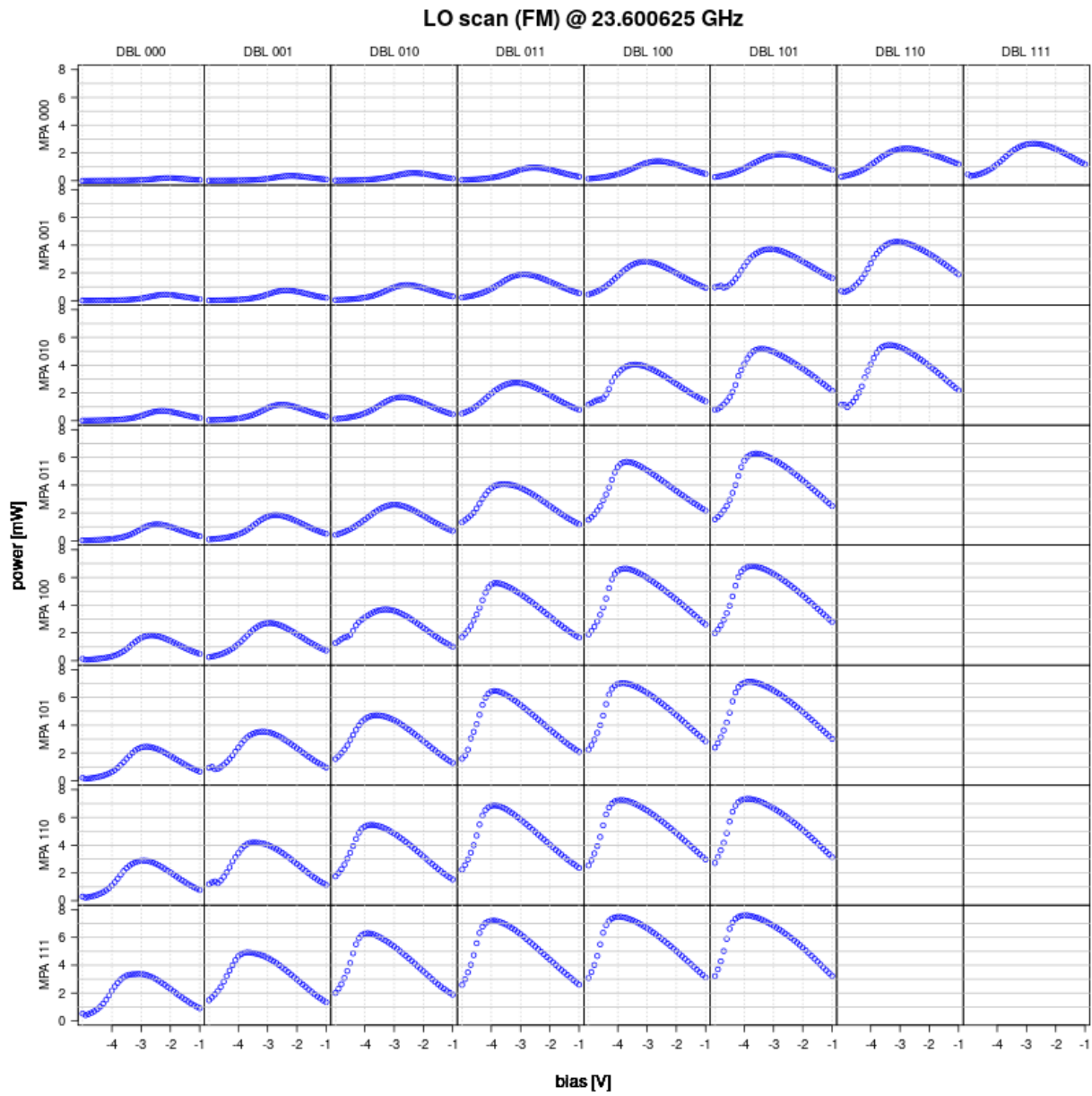




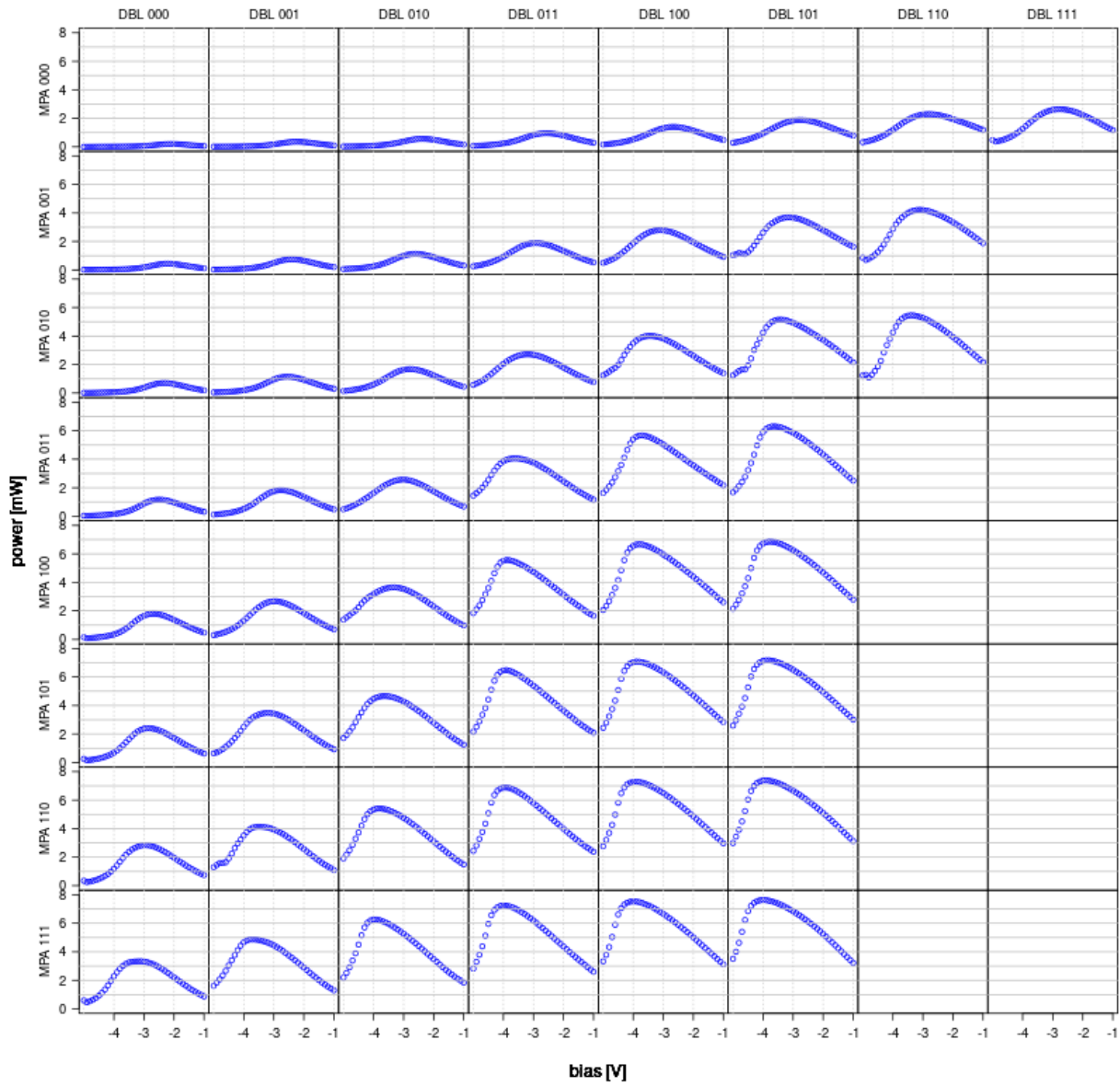
LO scan (FM) @ 23.4675 GHz

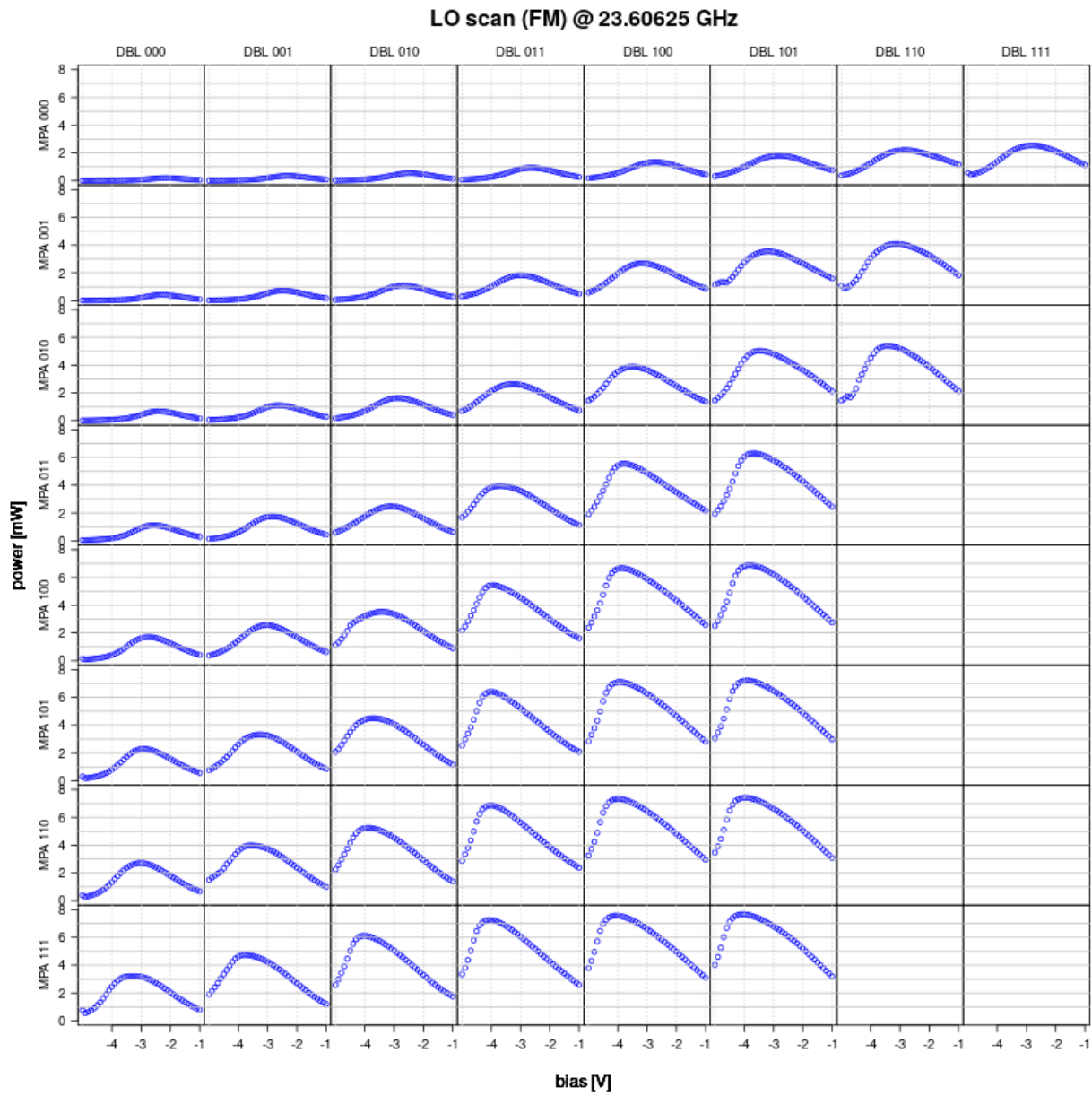




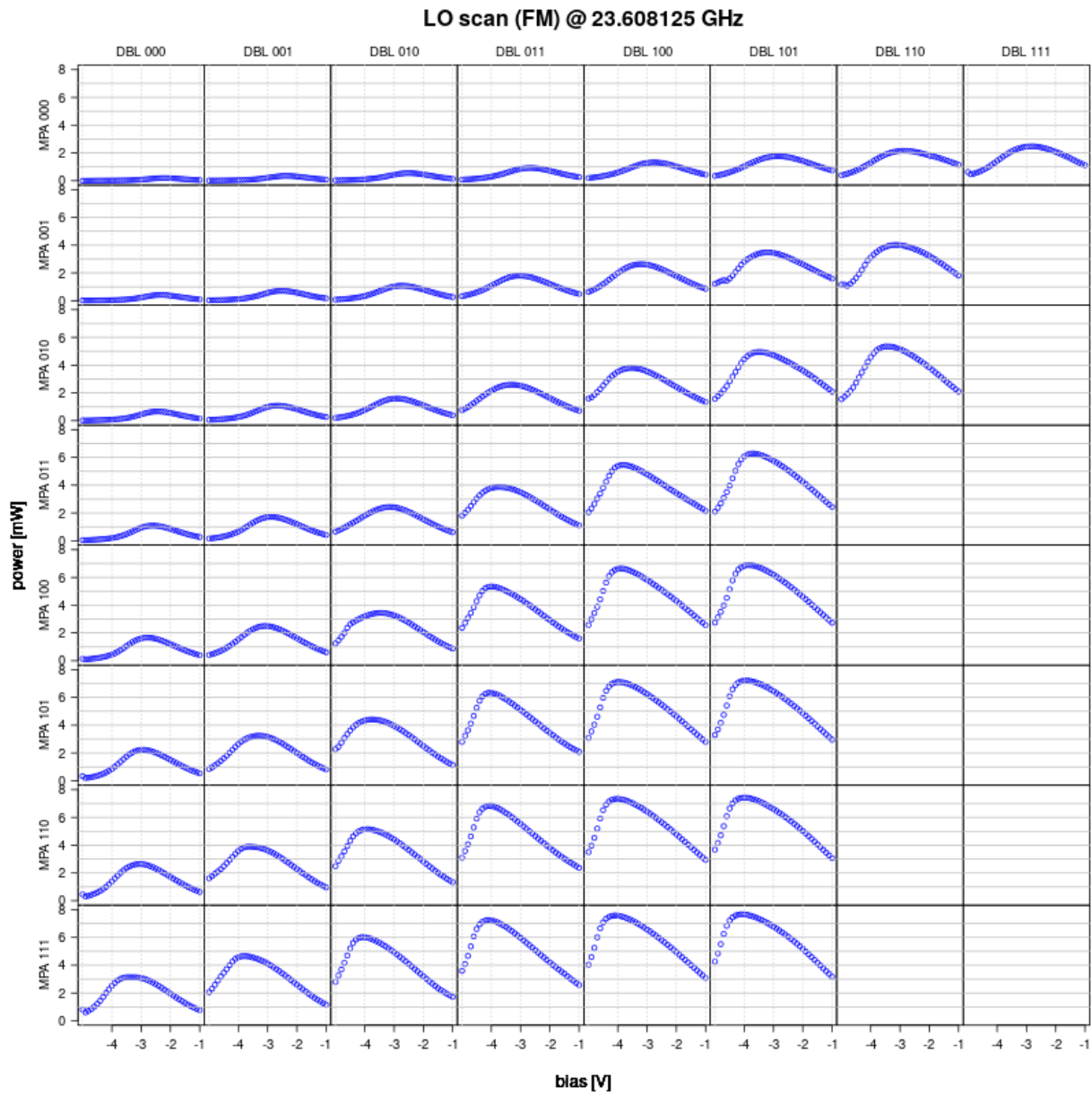


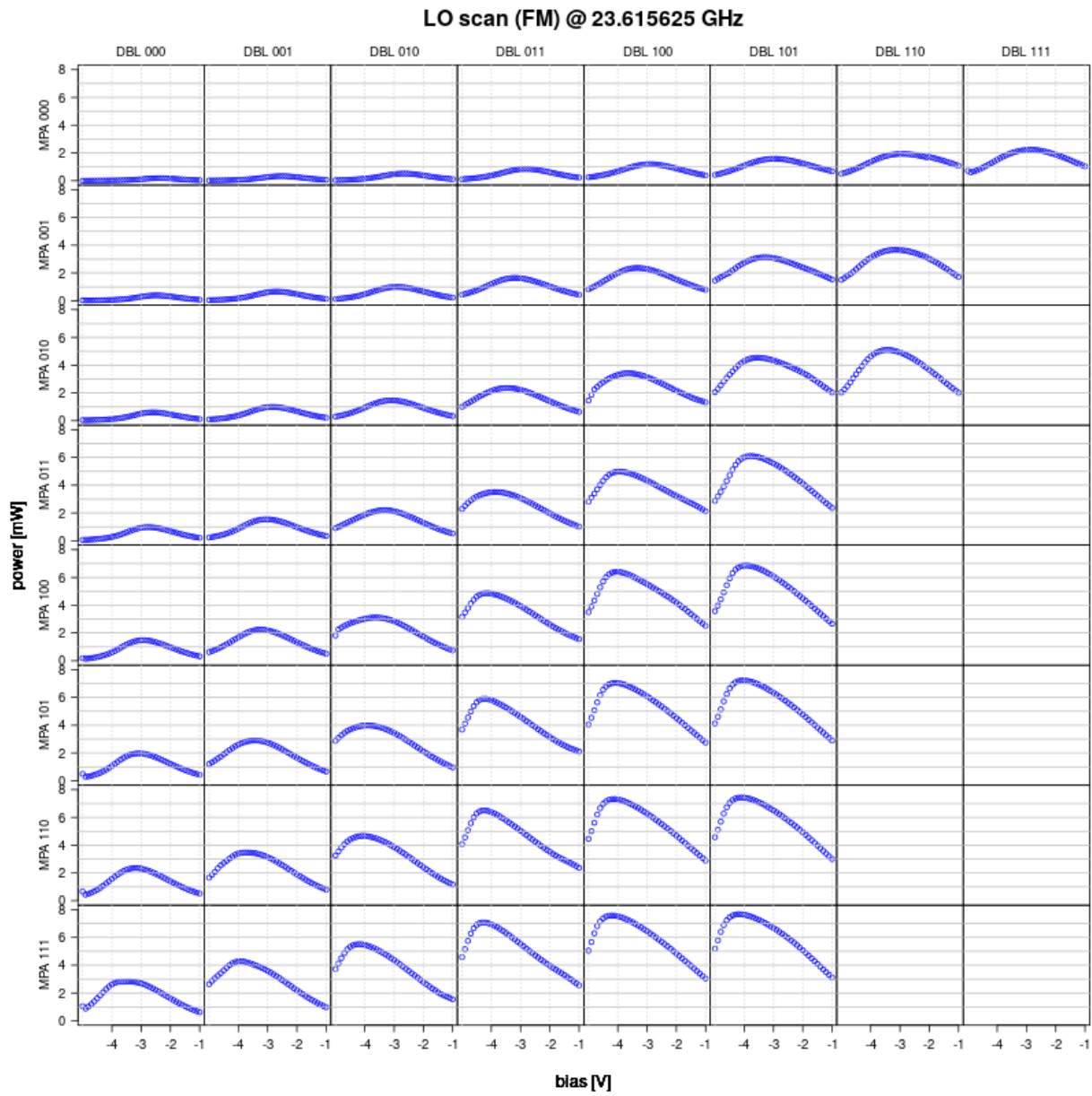
LO scan (FM) @ 23.6025 GHz

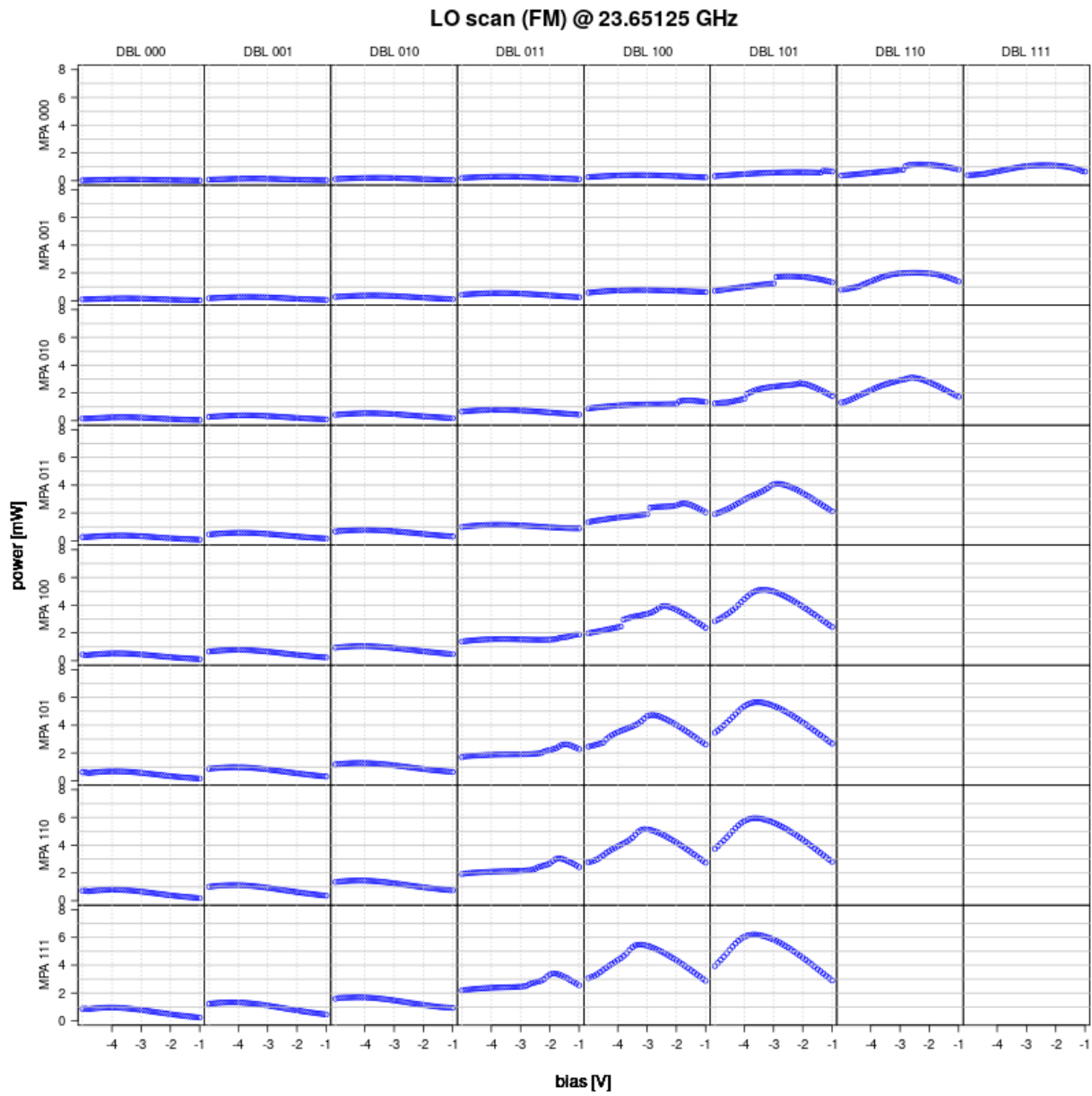


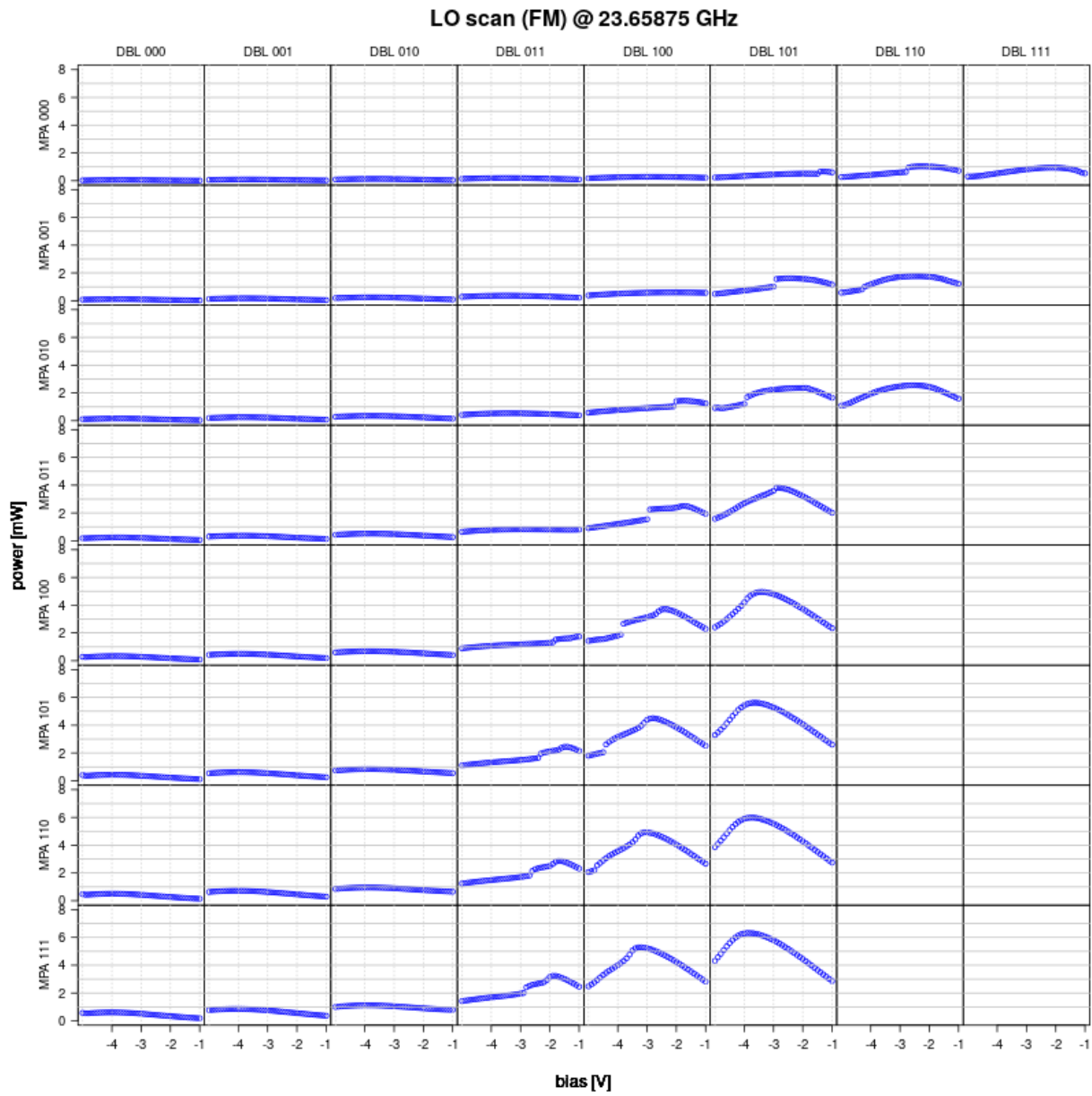




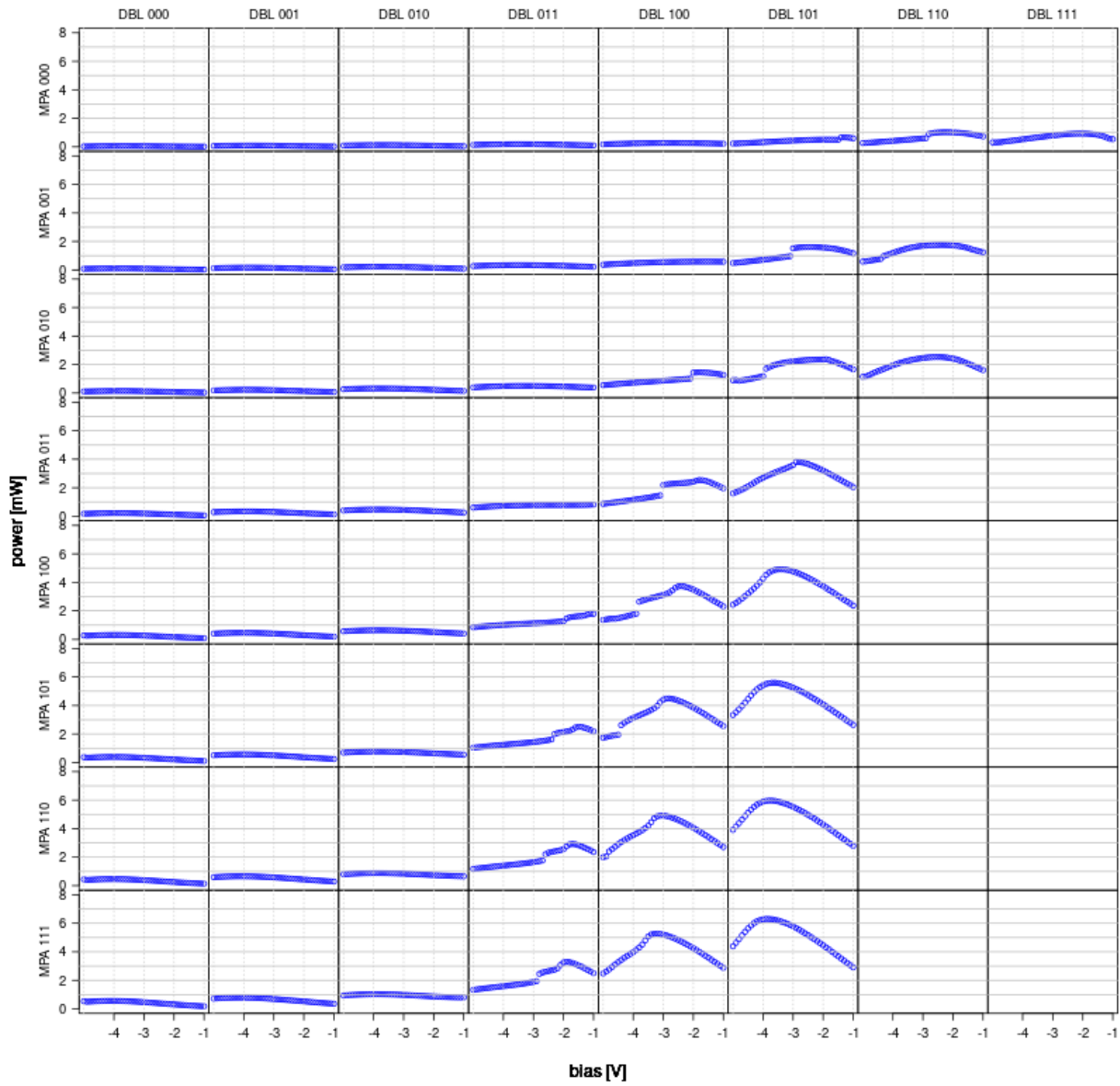


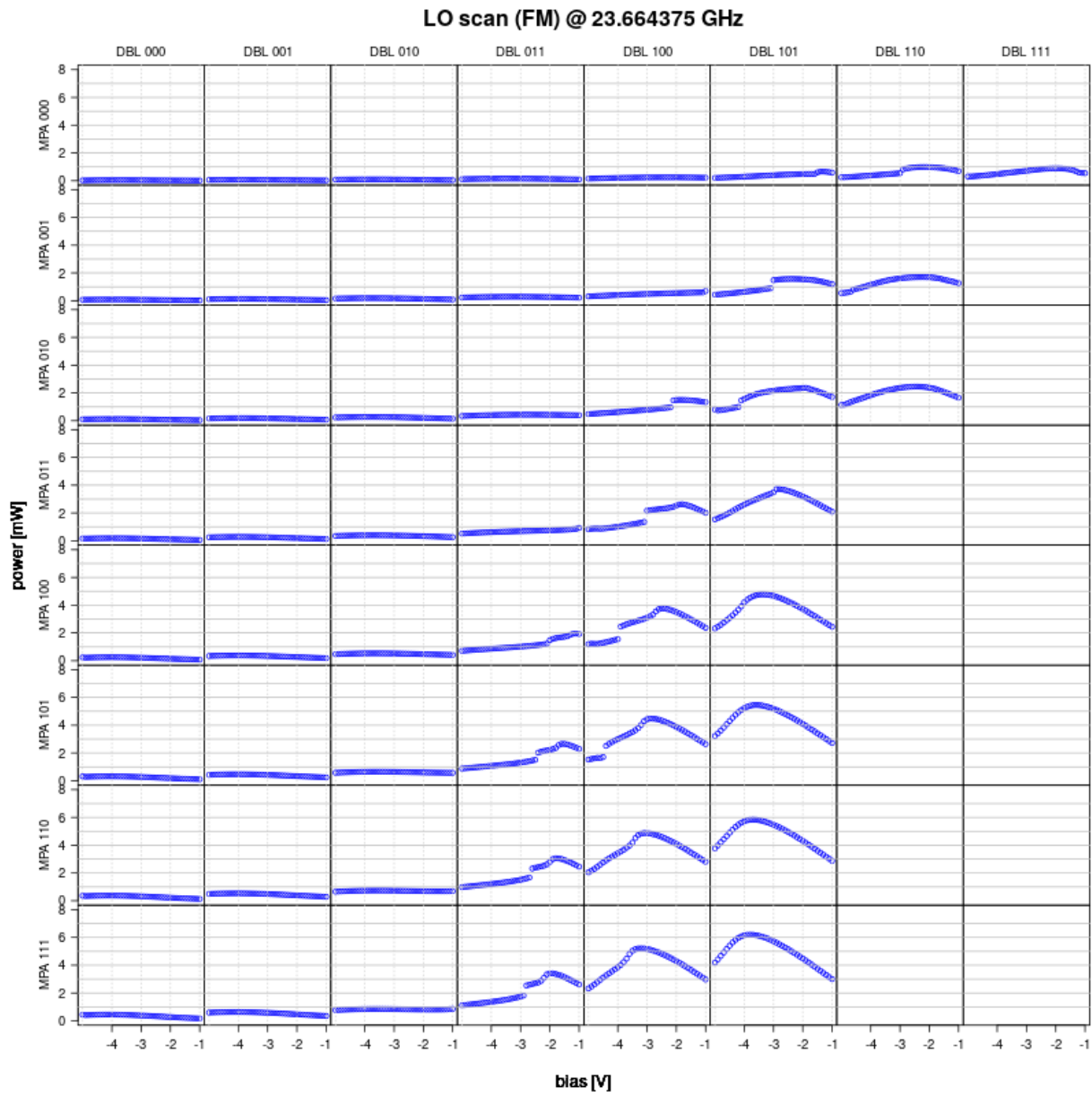


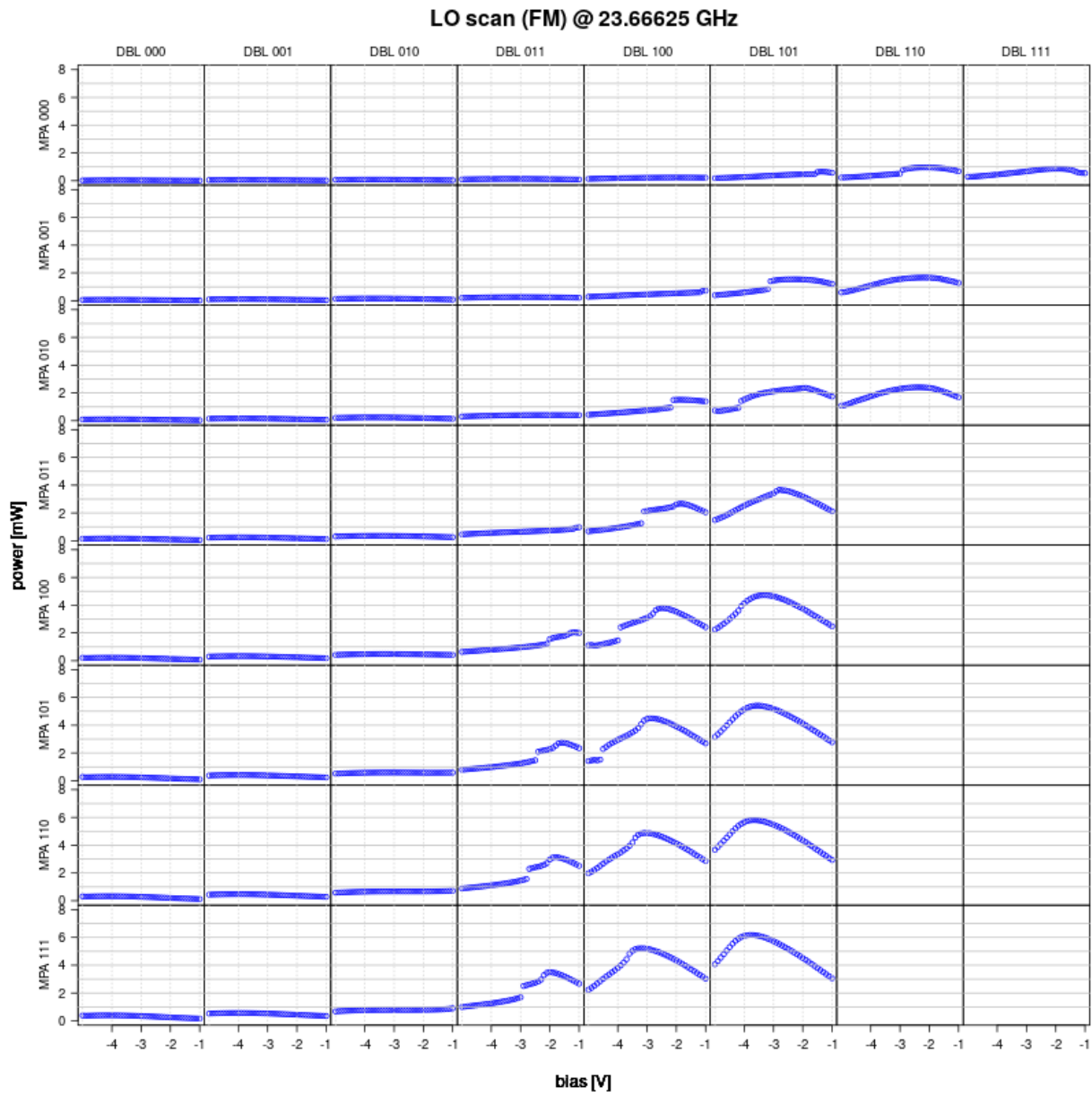


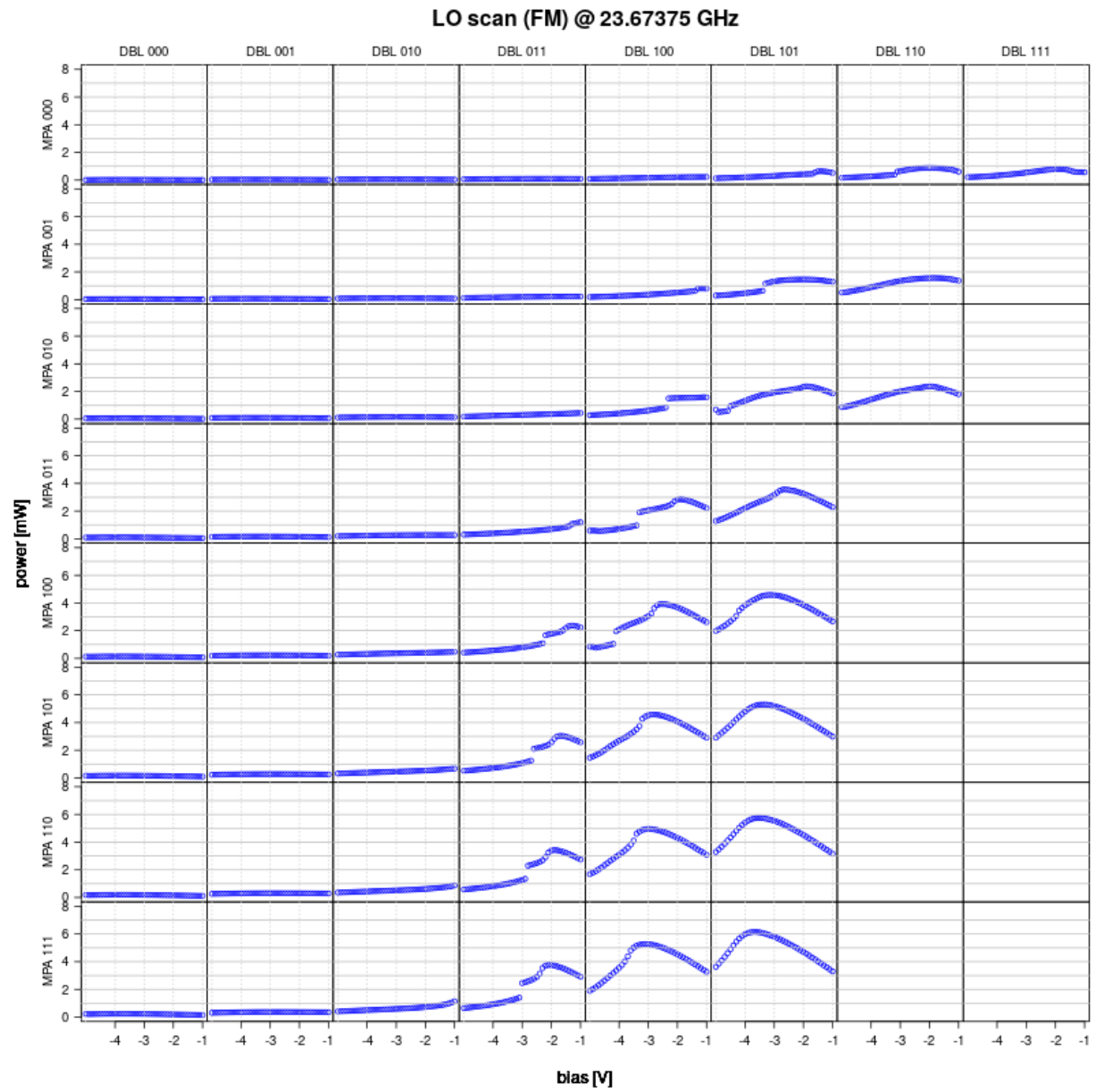


**LO scan (FM) @ 23.660625 GHz**

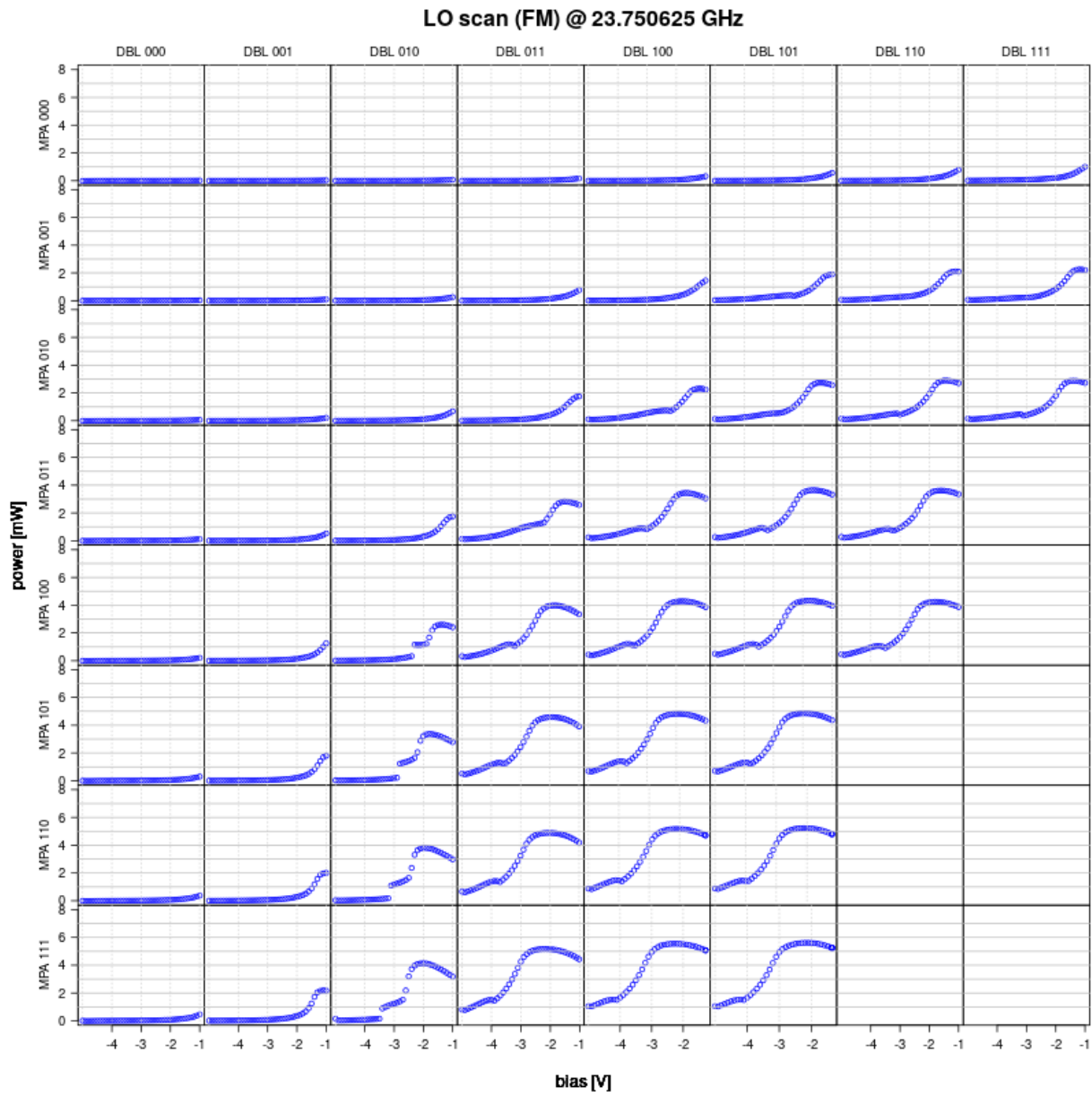


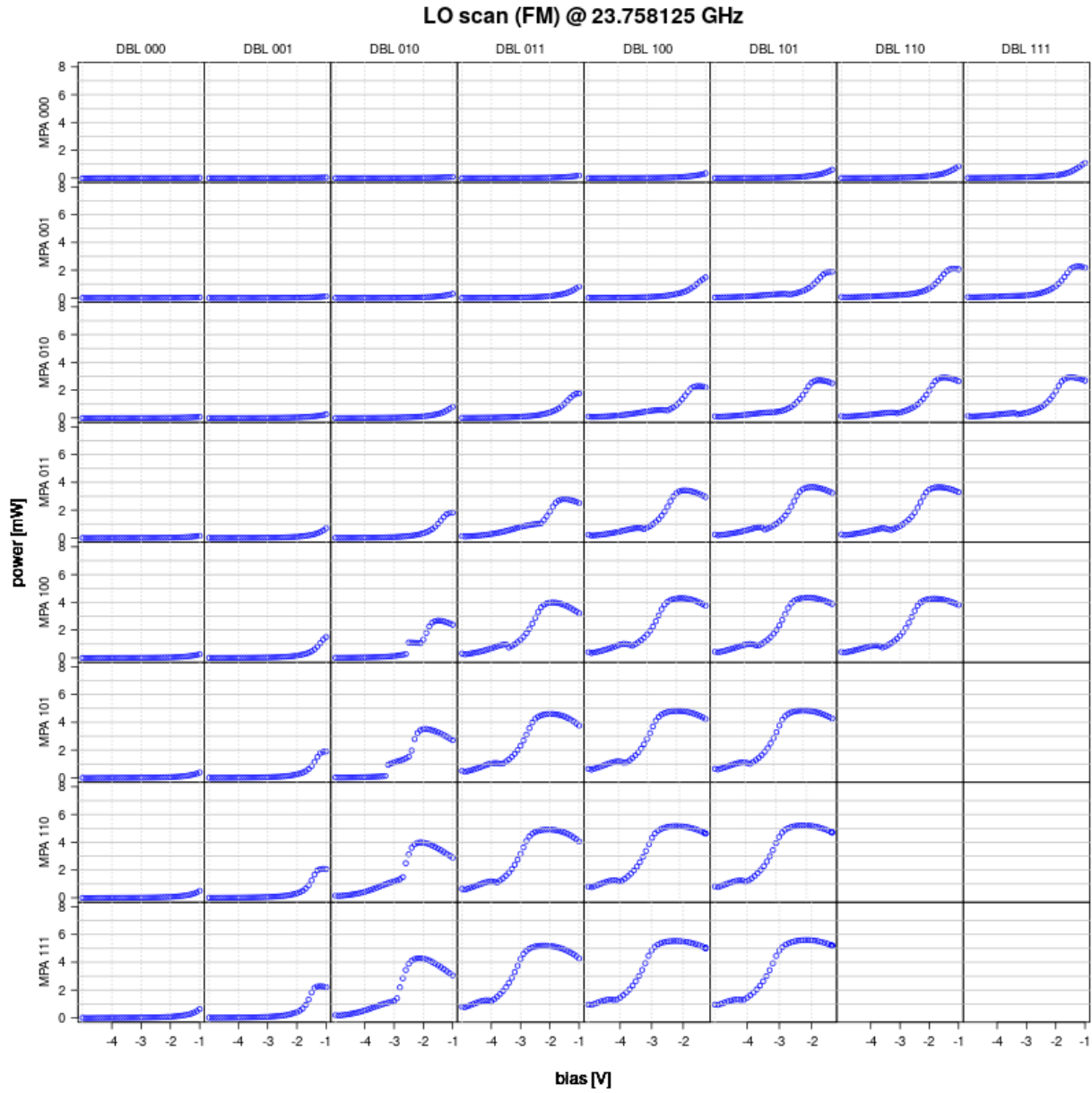




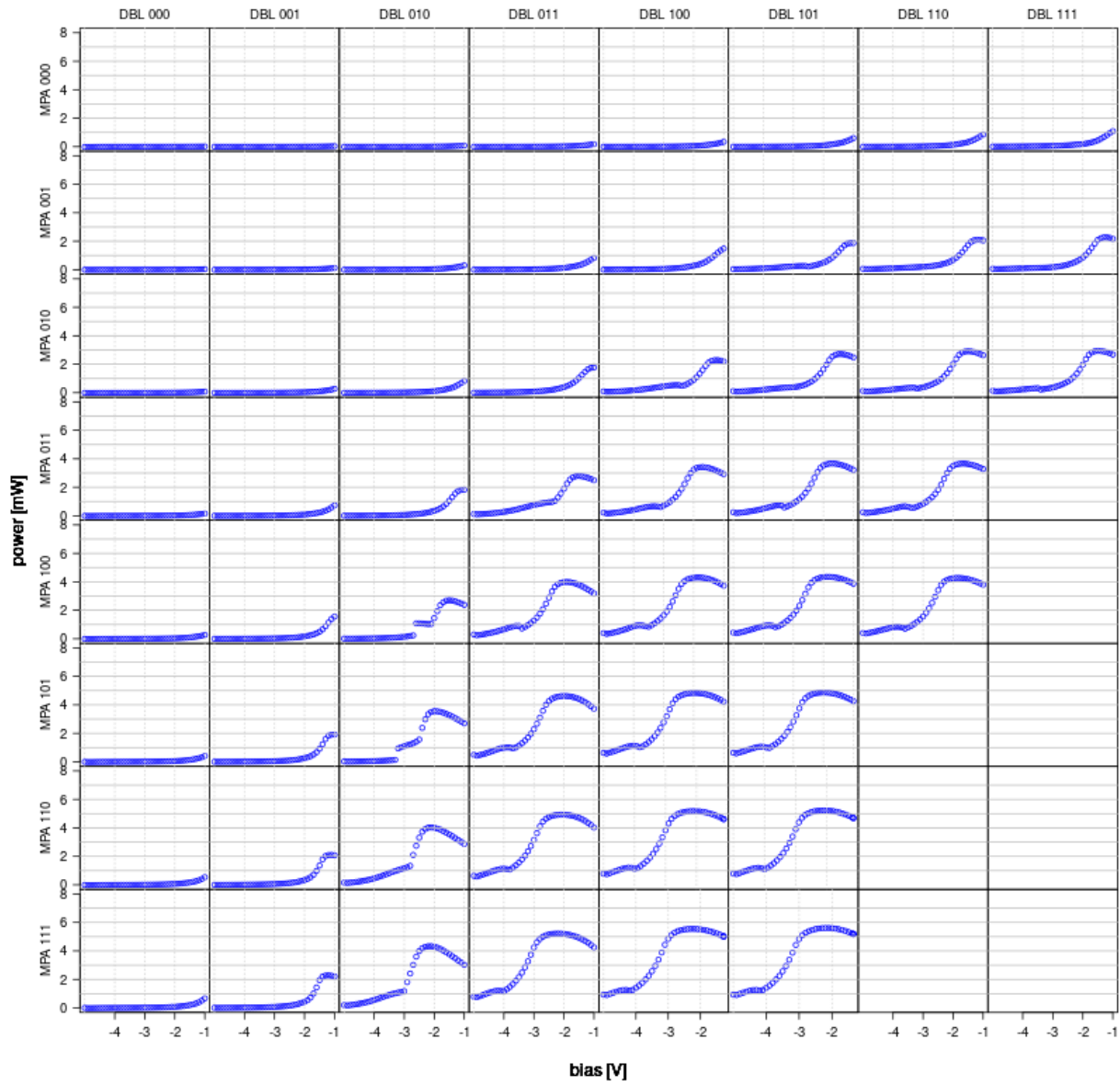


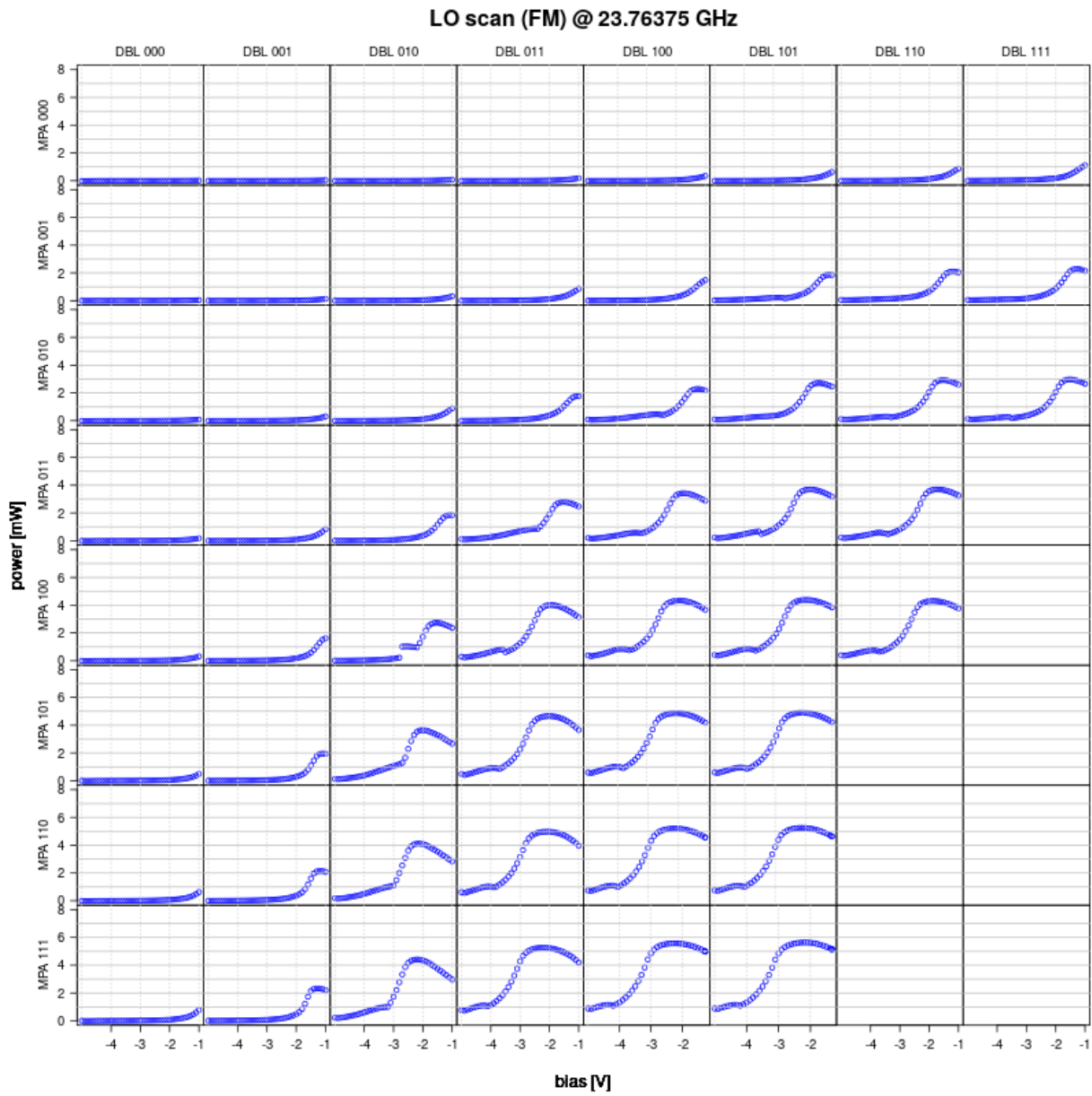


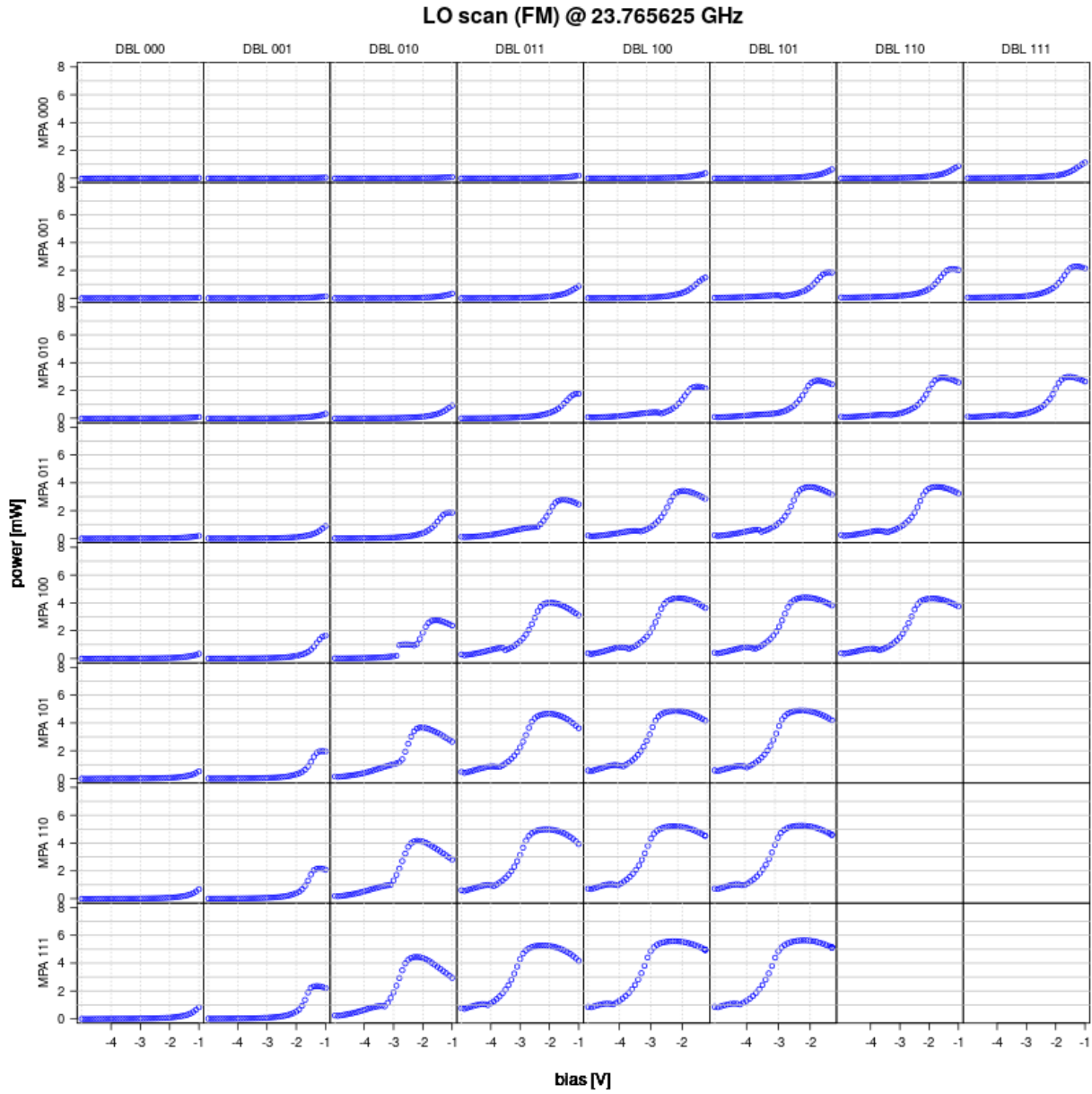


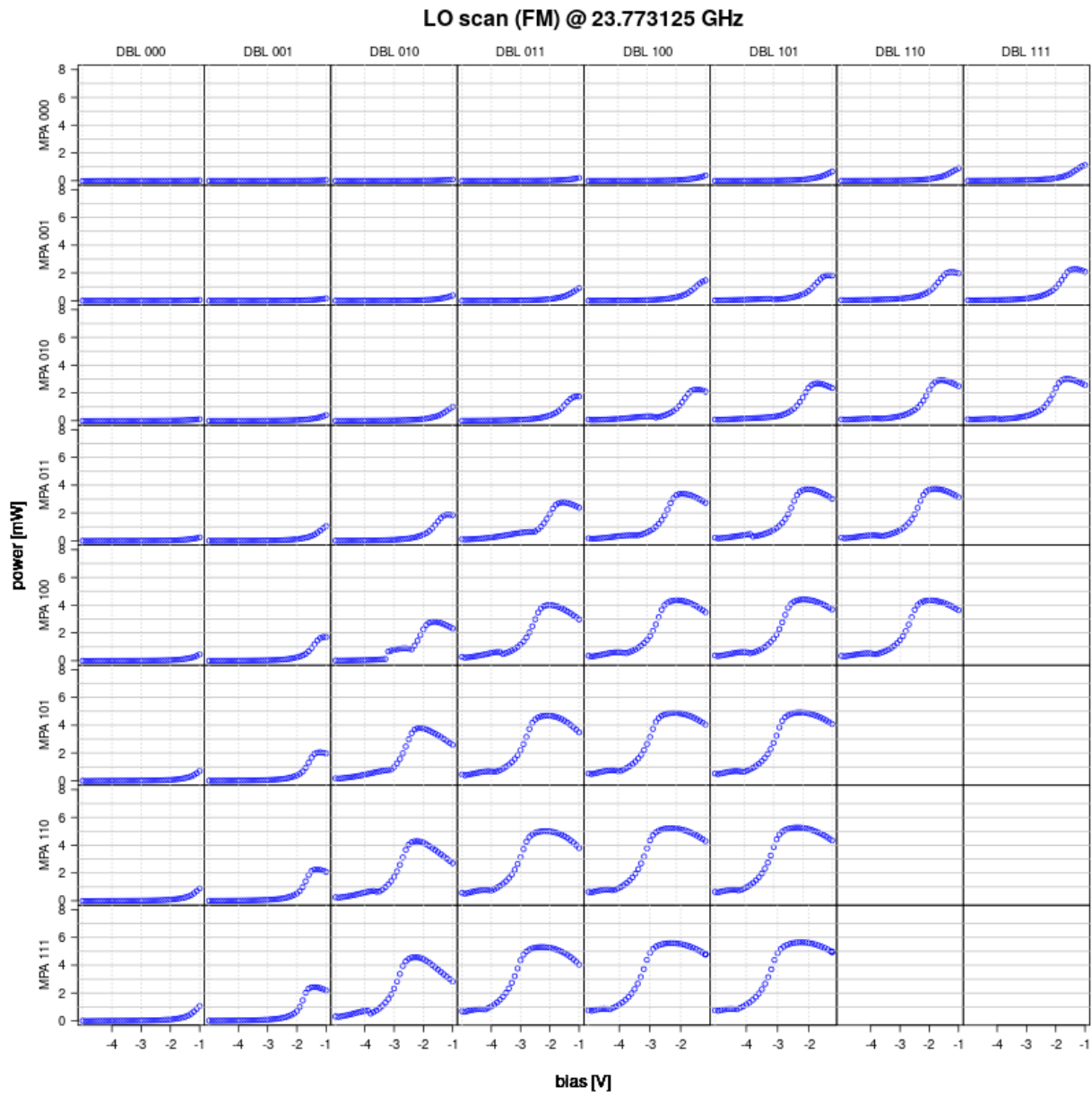


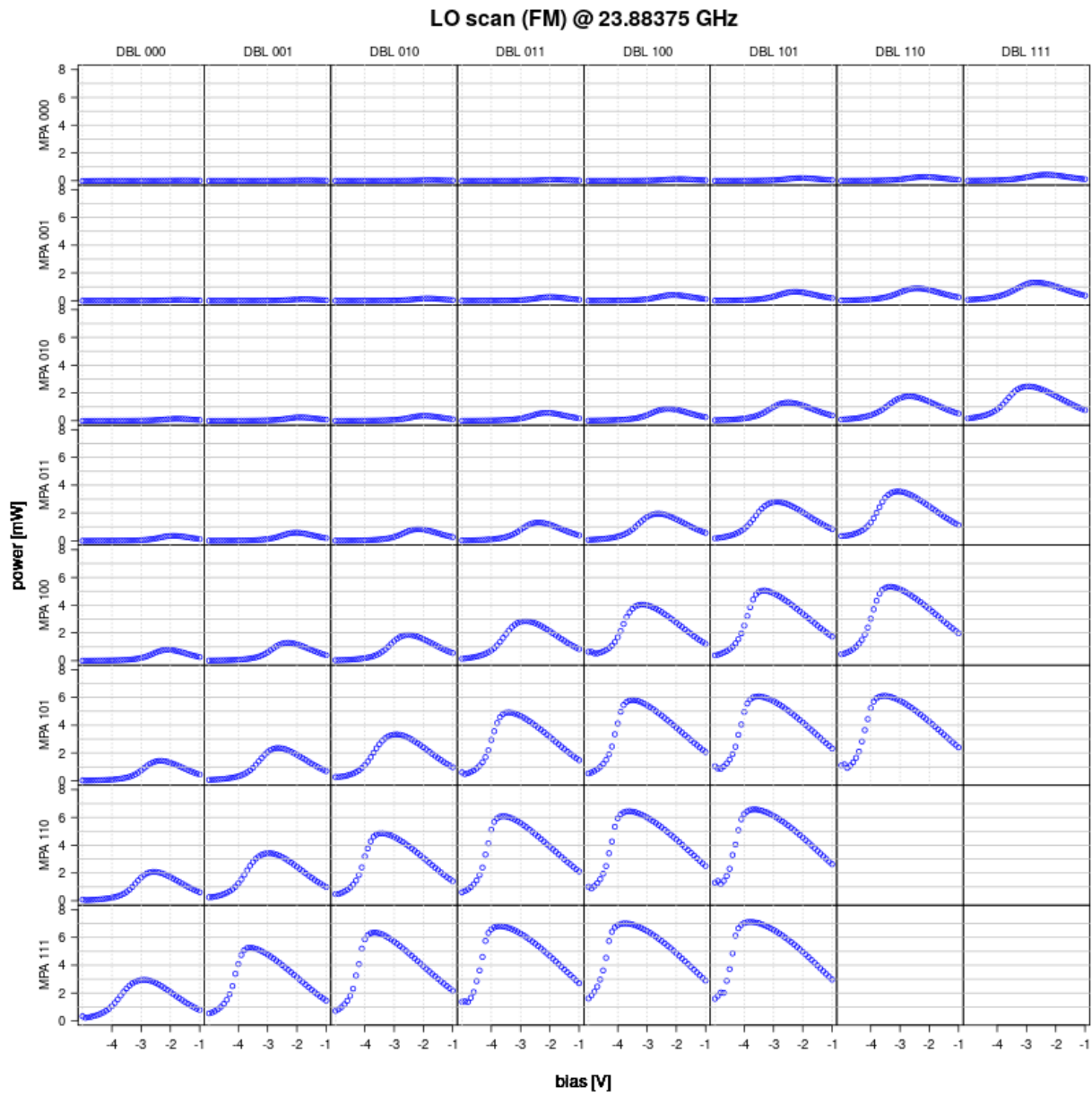
LO scan (FM) @ 23.76 GHz

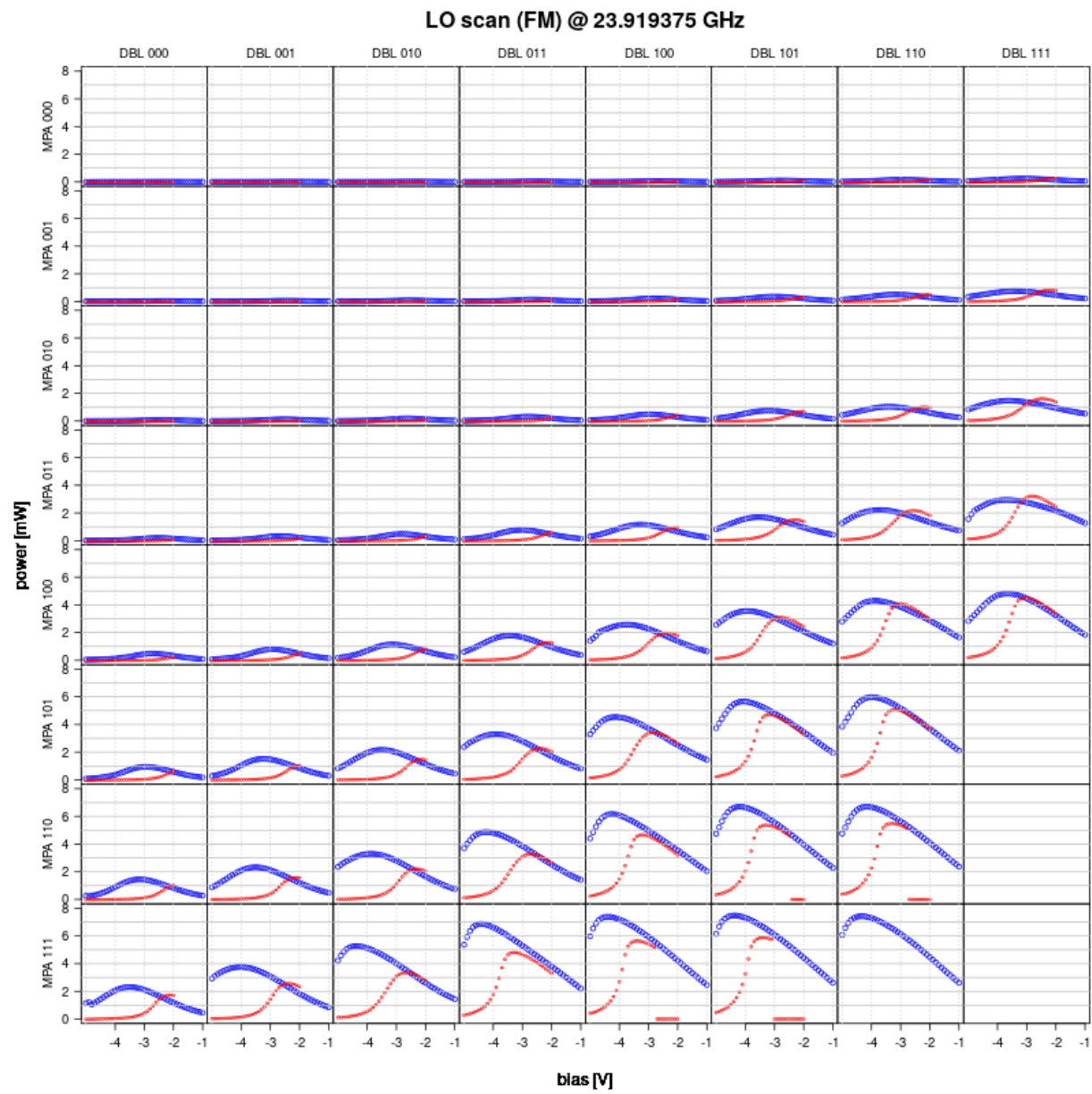




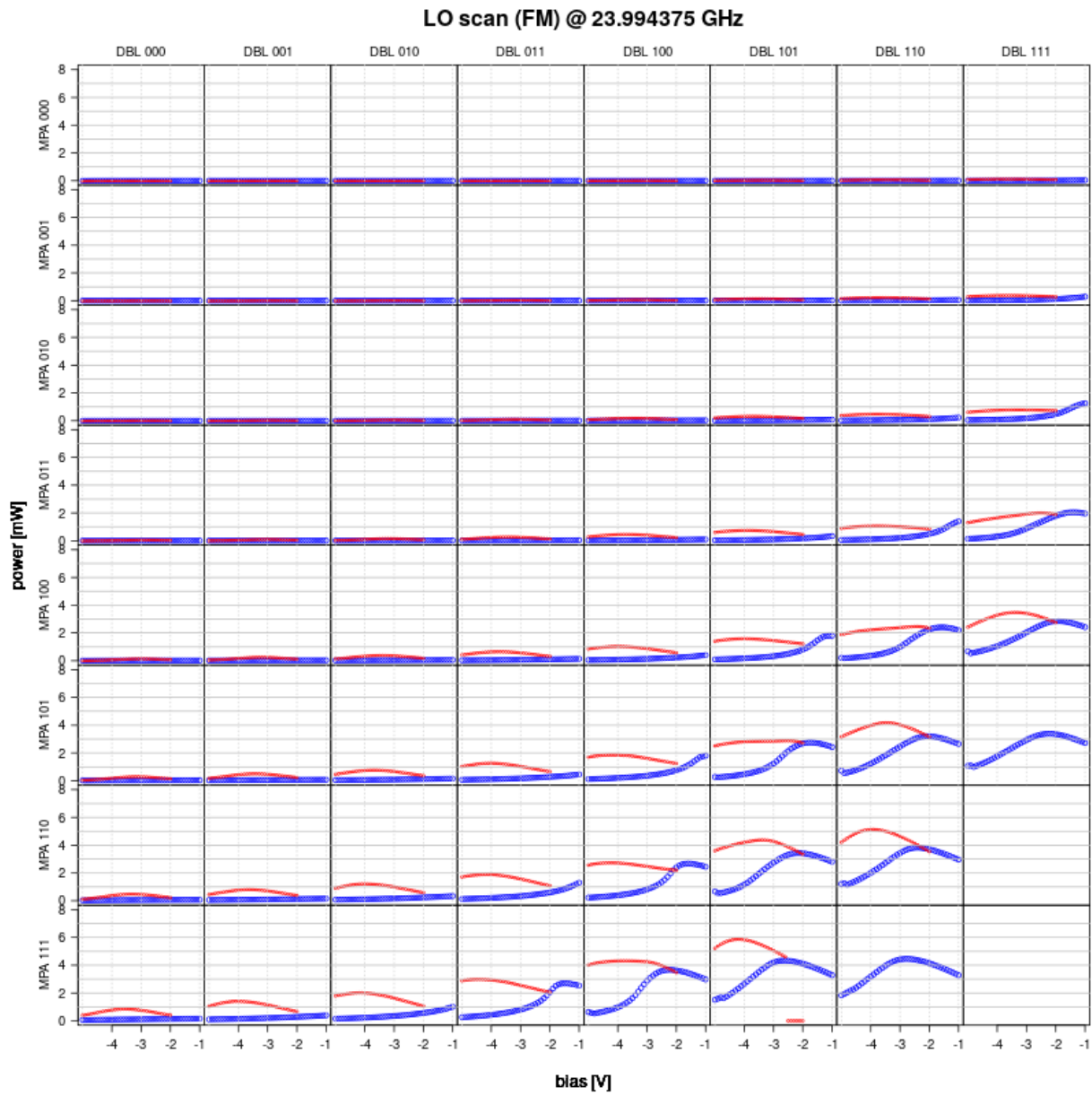


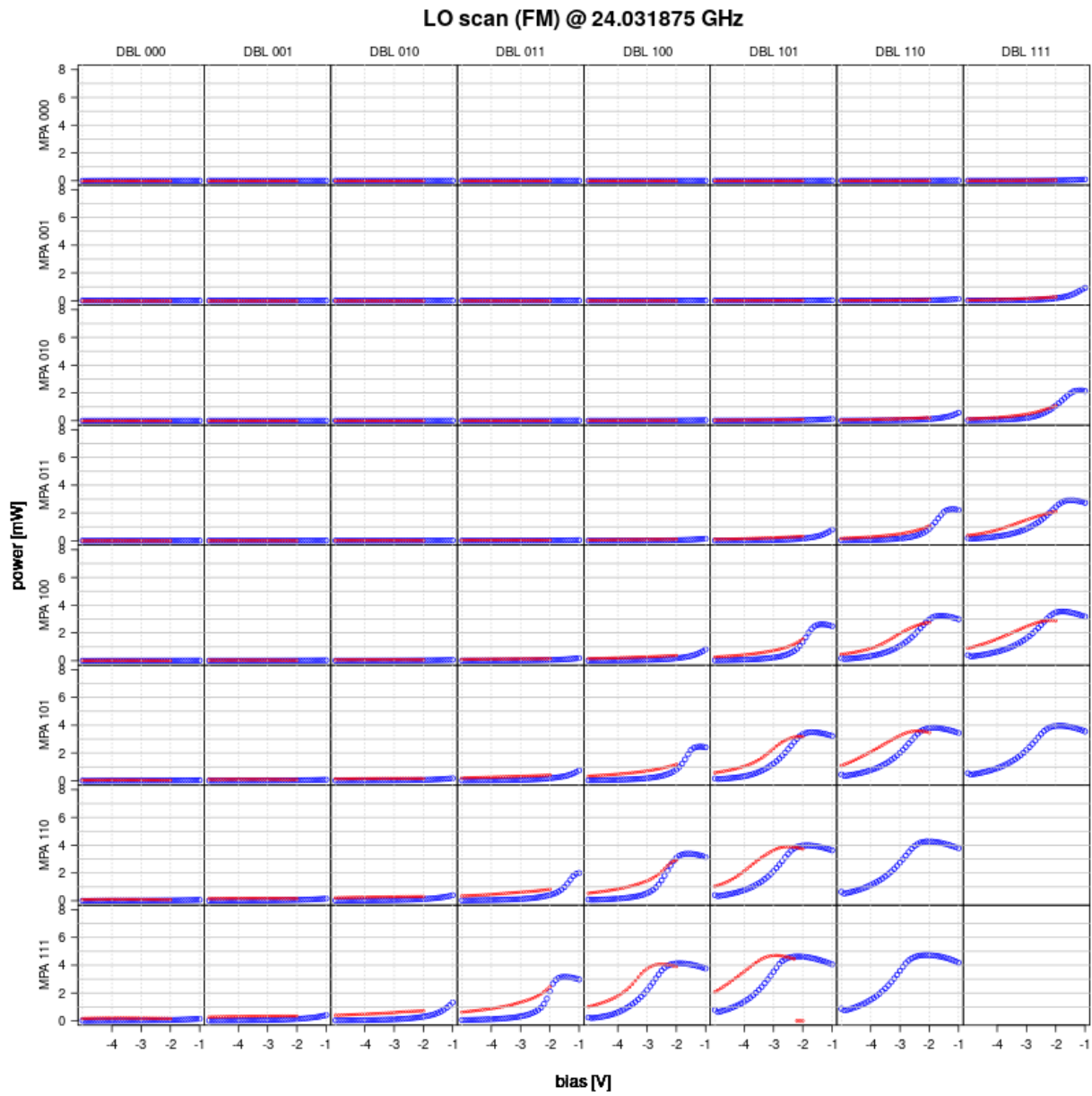


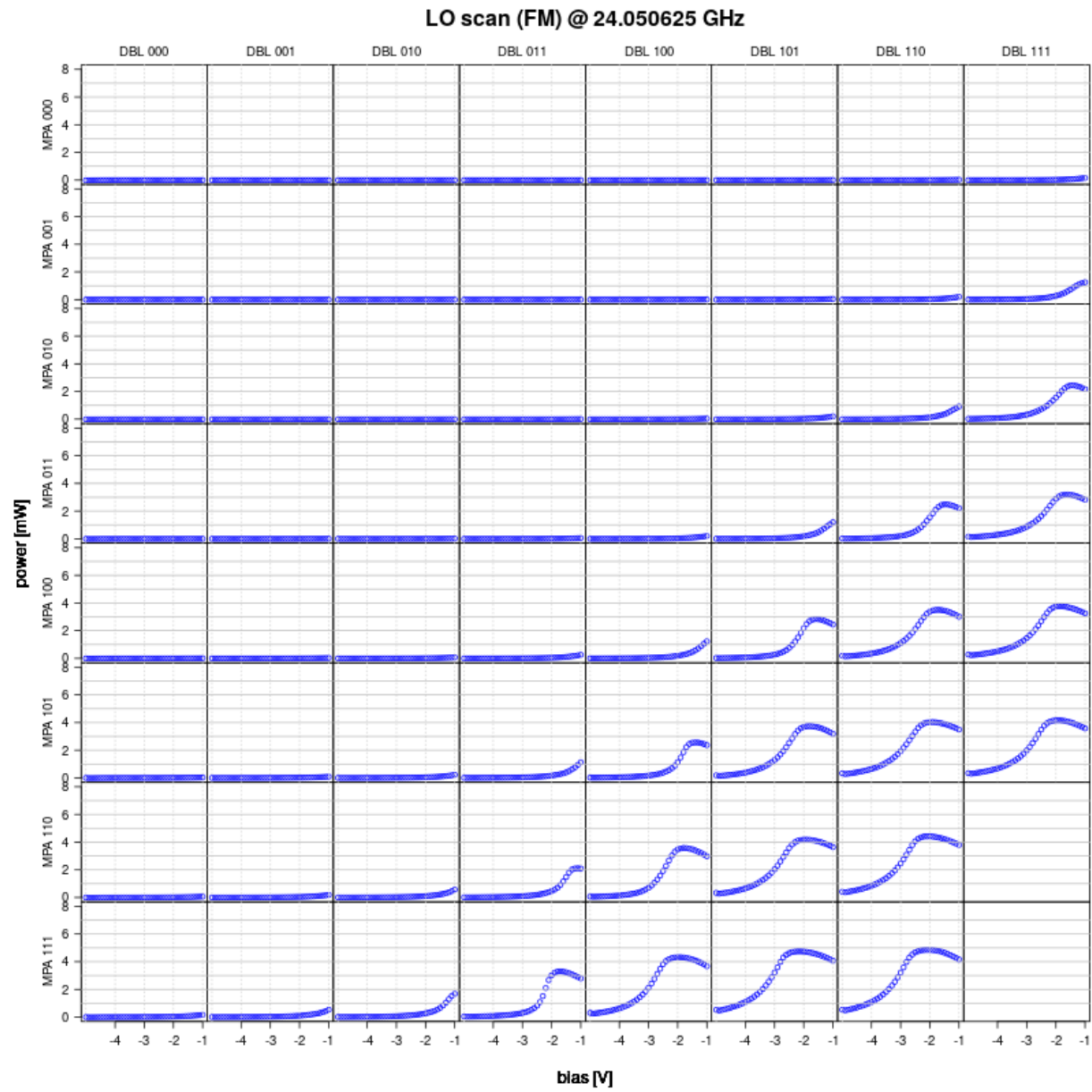




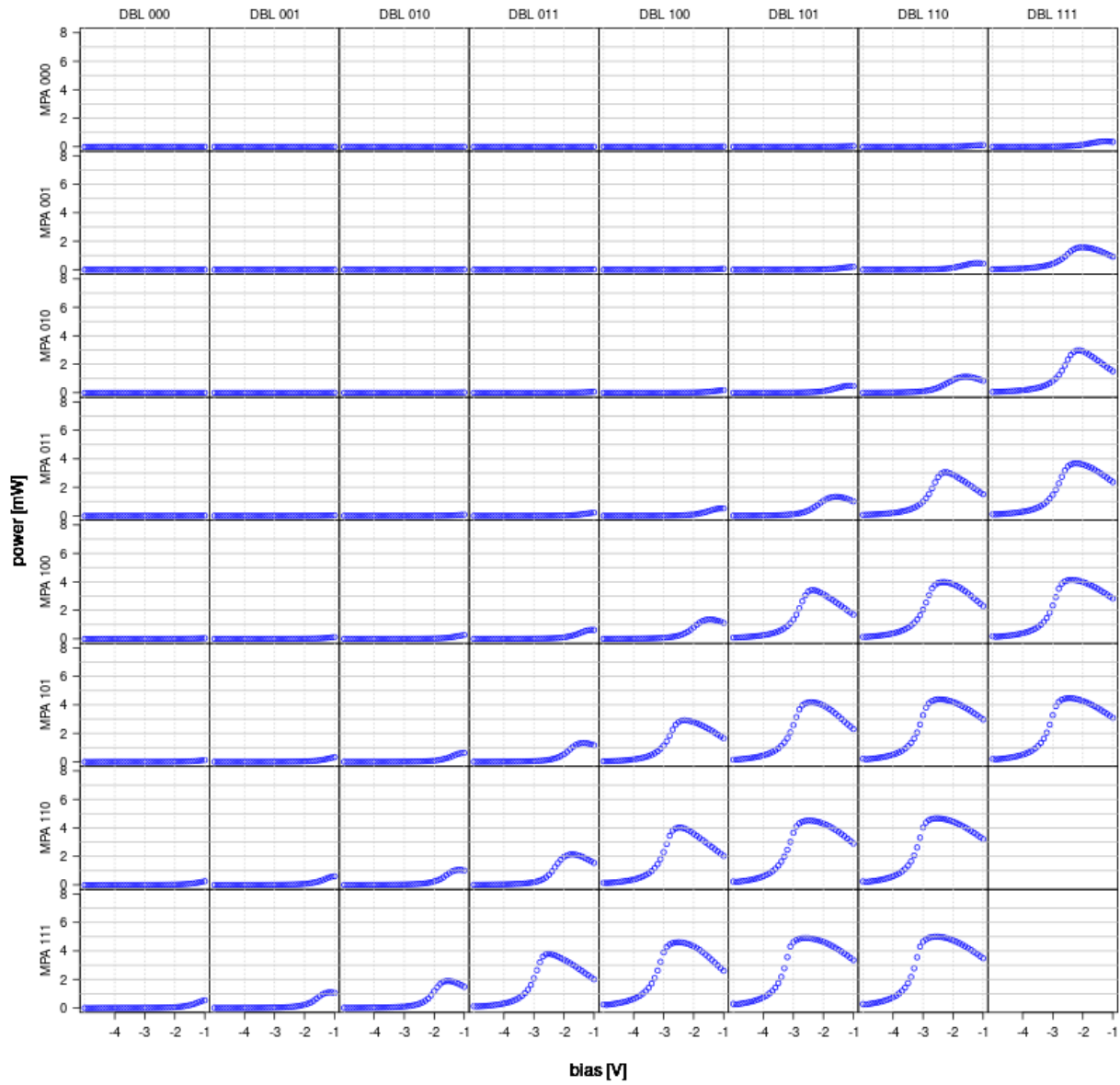


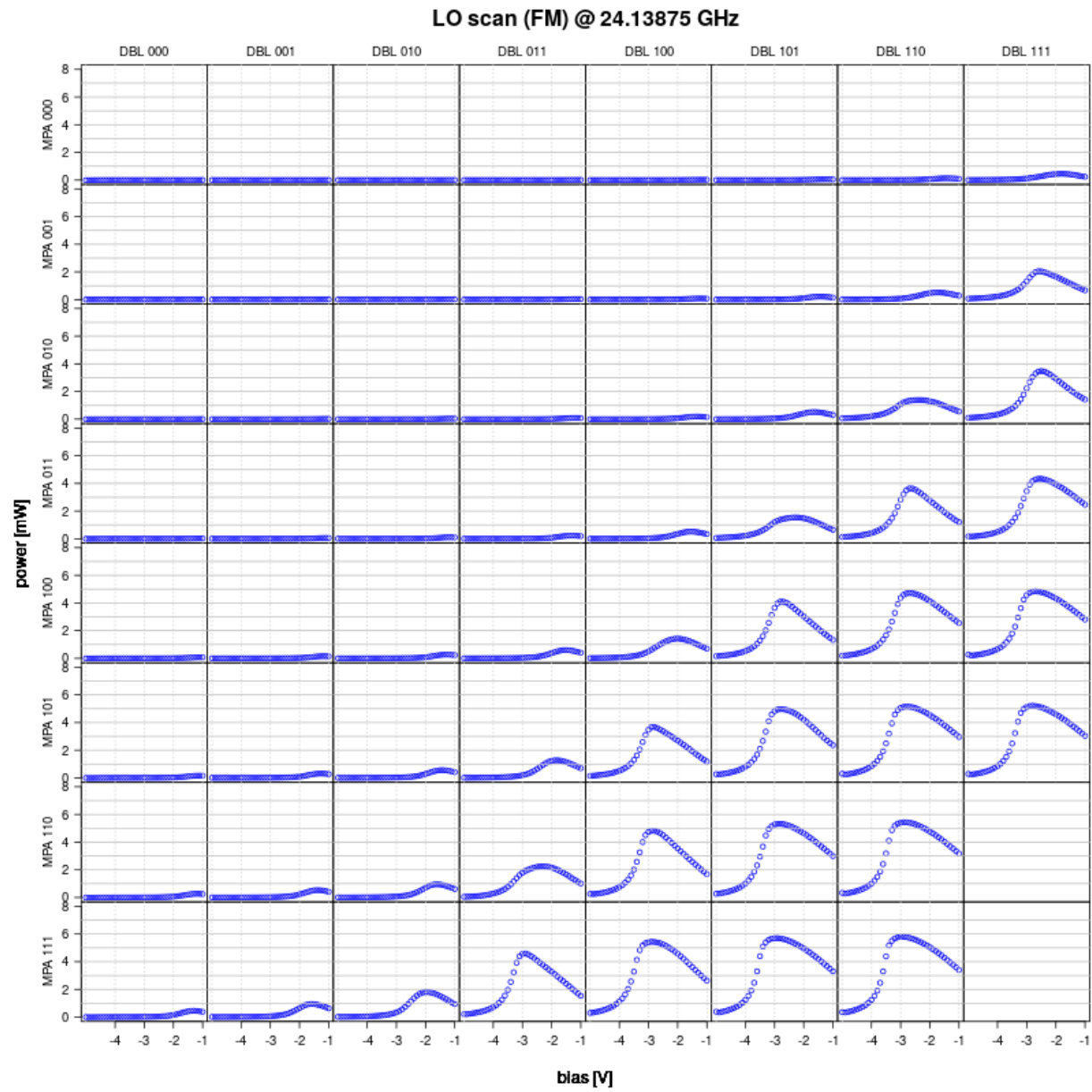




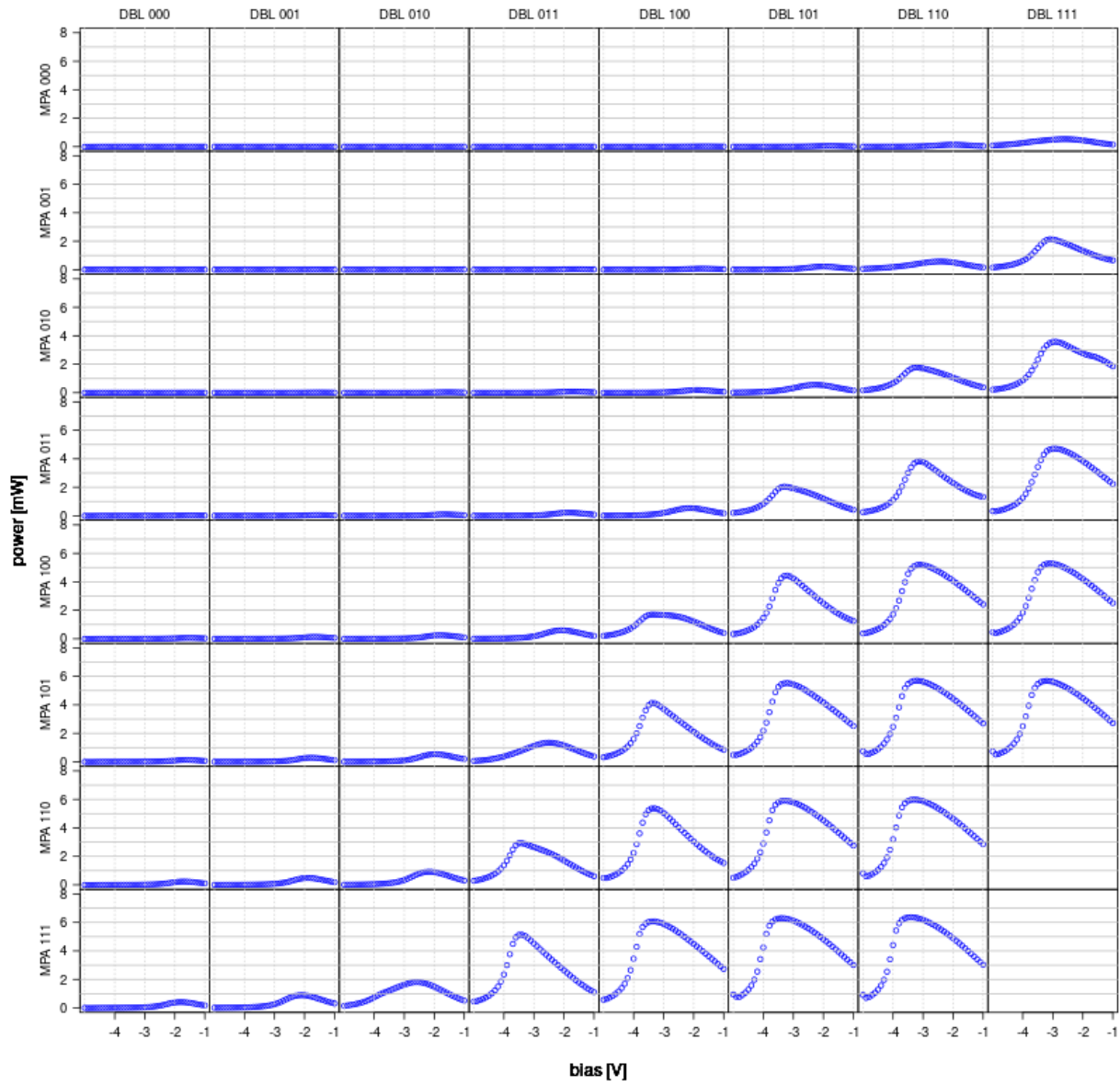


**LO scan (FM) @ 24.105 GHz**

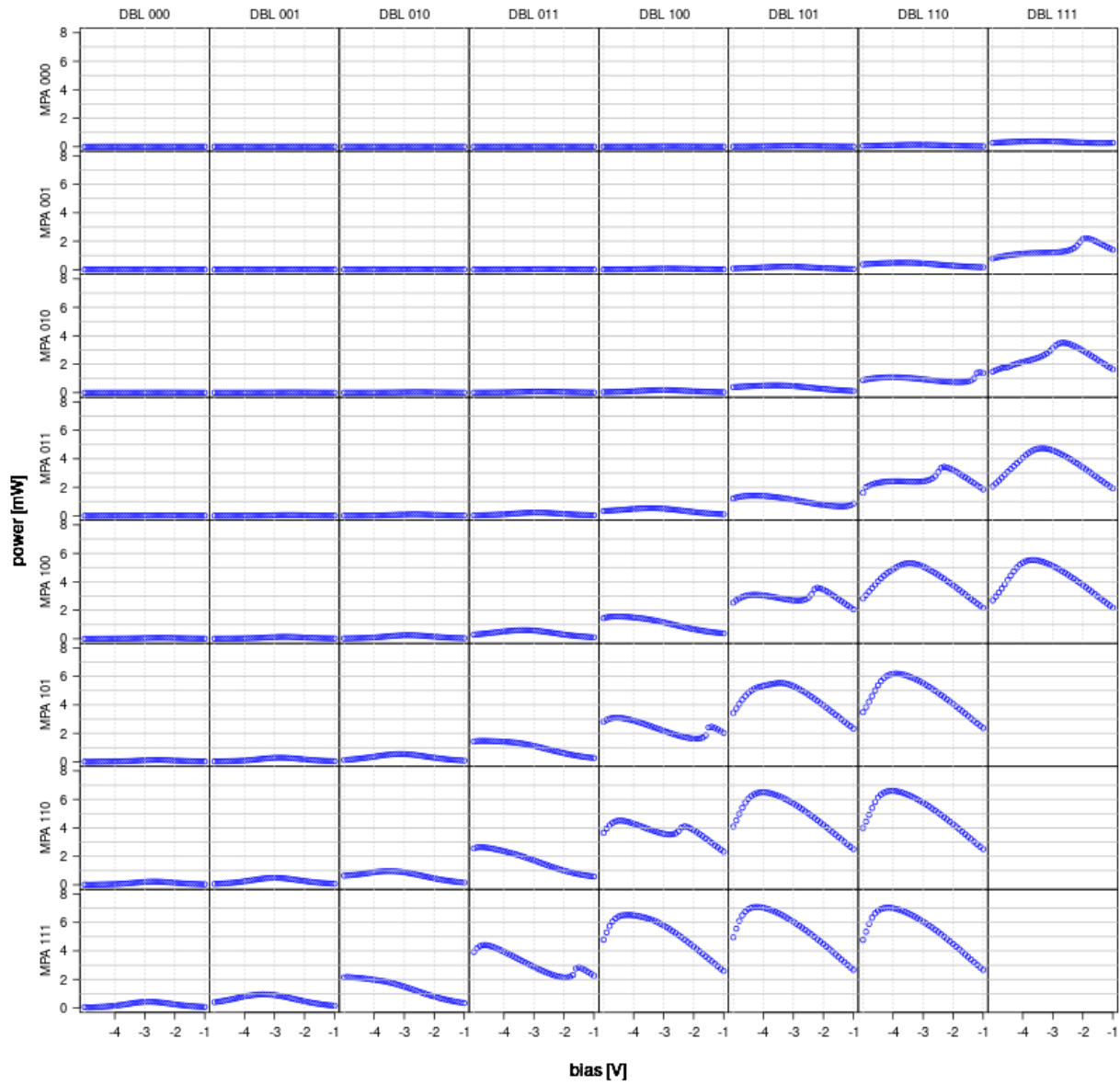




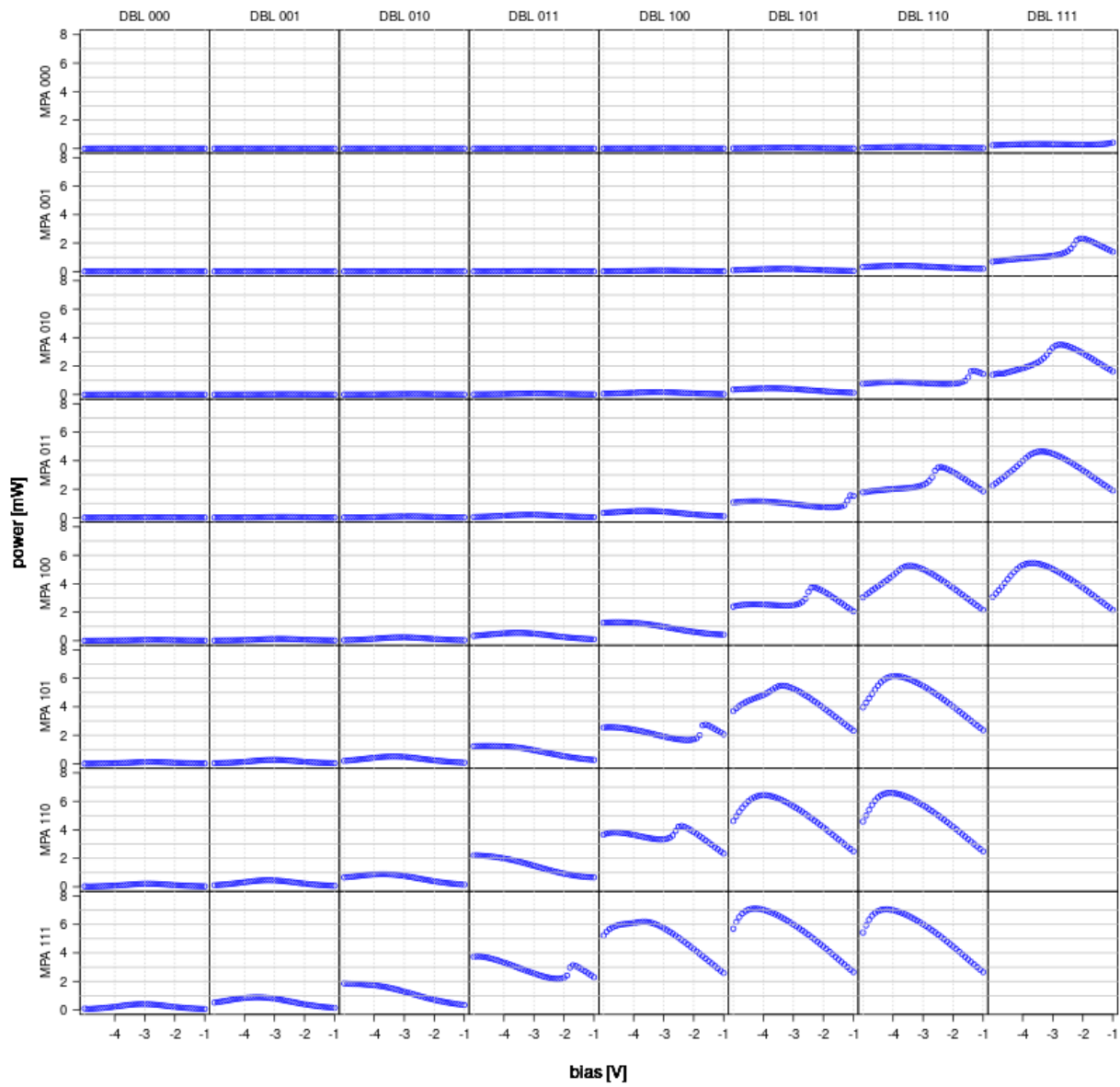
LO scan (FM) @ 24.1725 GHz



**LO scan (FM) @ 24.223125 GHz**

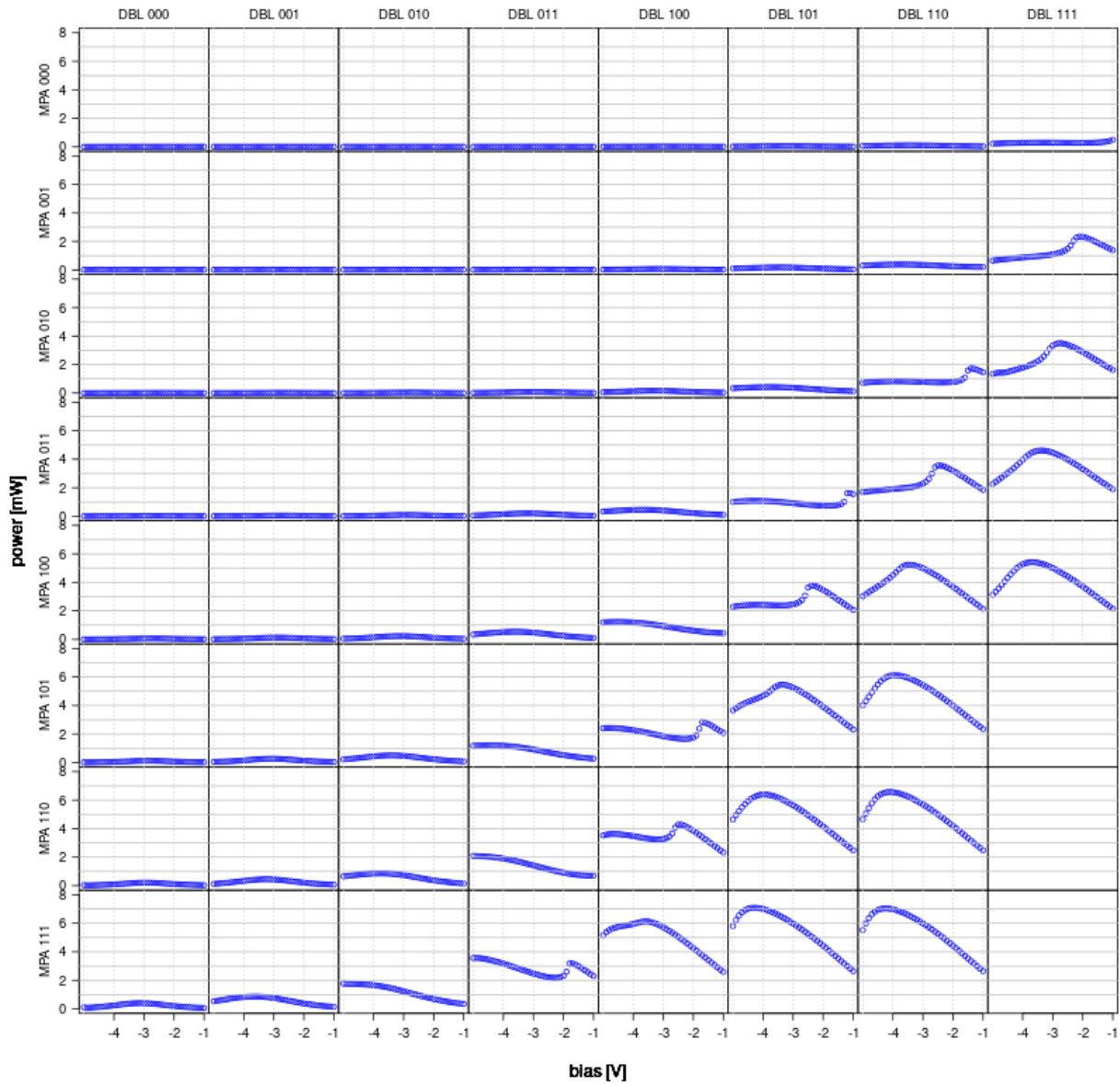


LO scan (FM) @ 24.230625 GHz

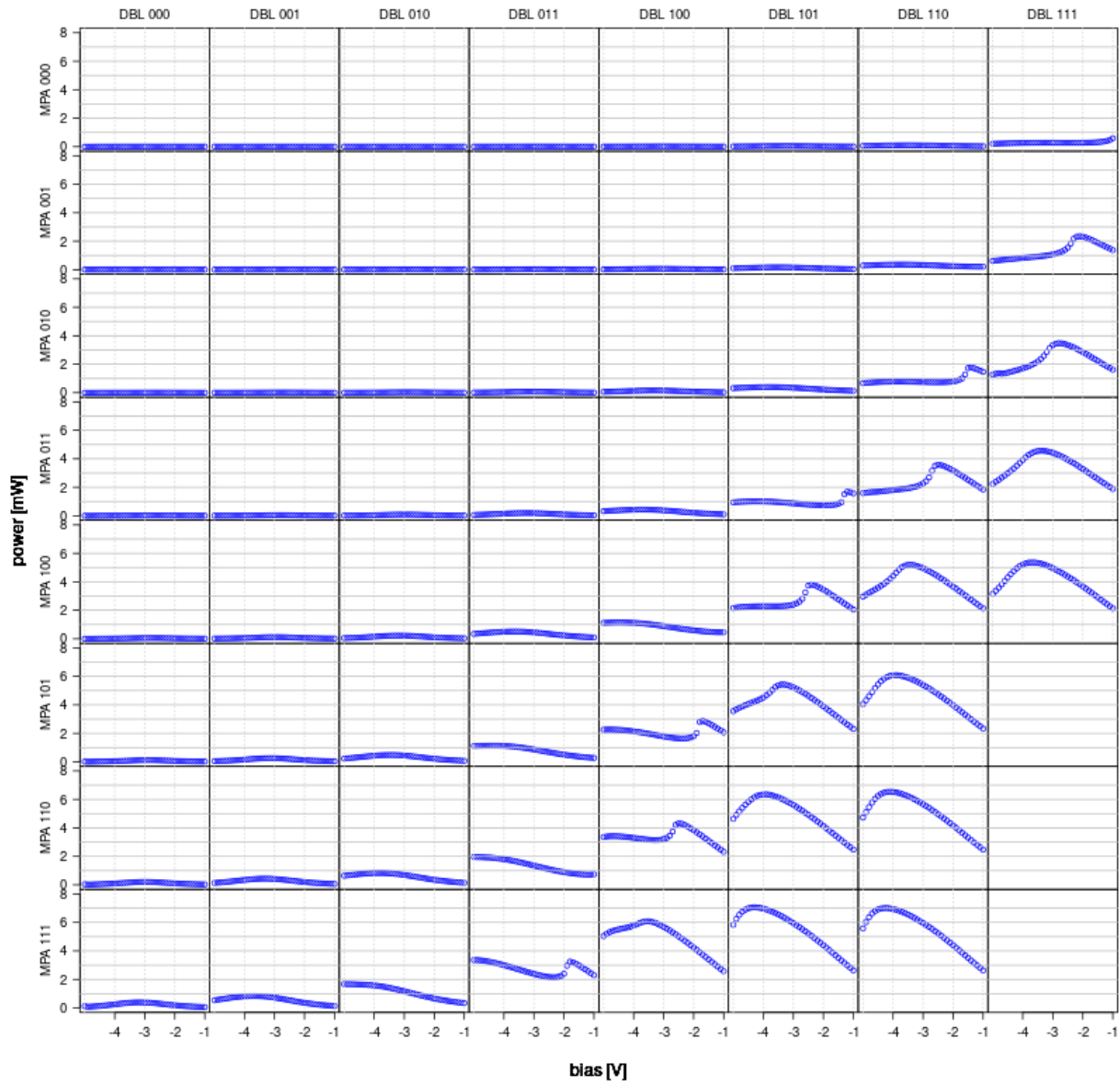


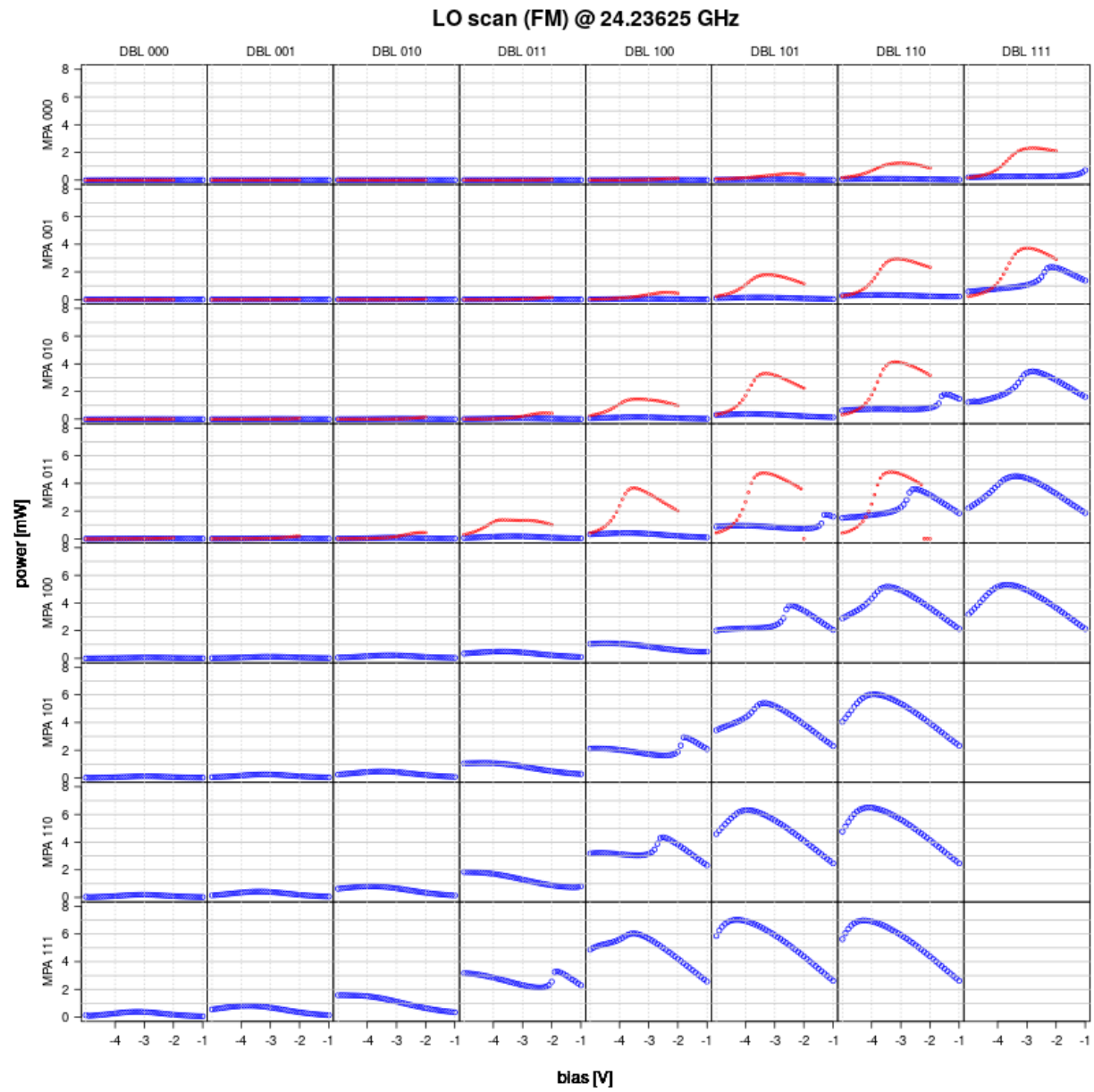


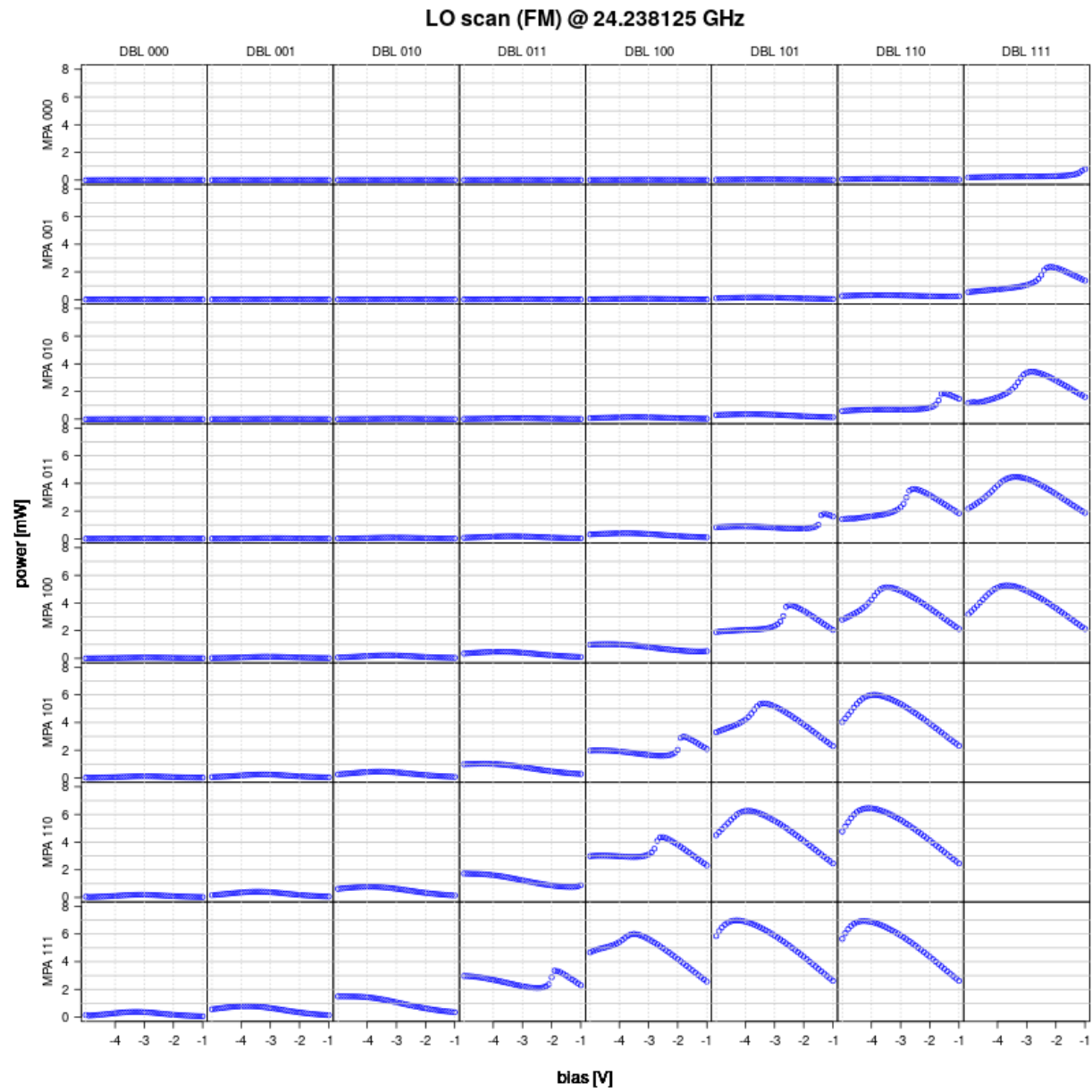
**LO scan (FM) @ 24.2325 GHz**

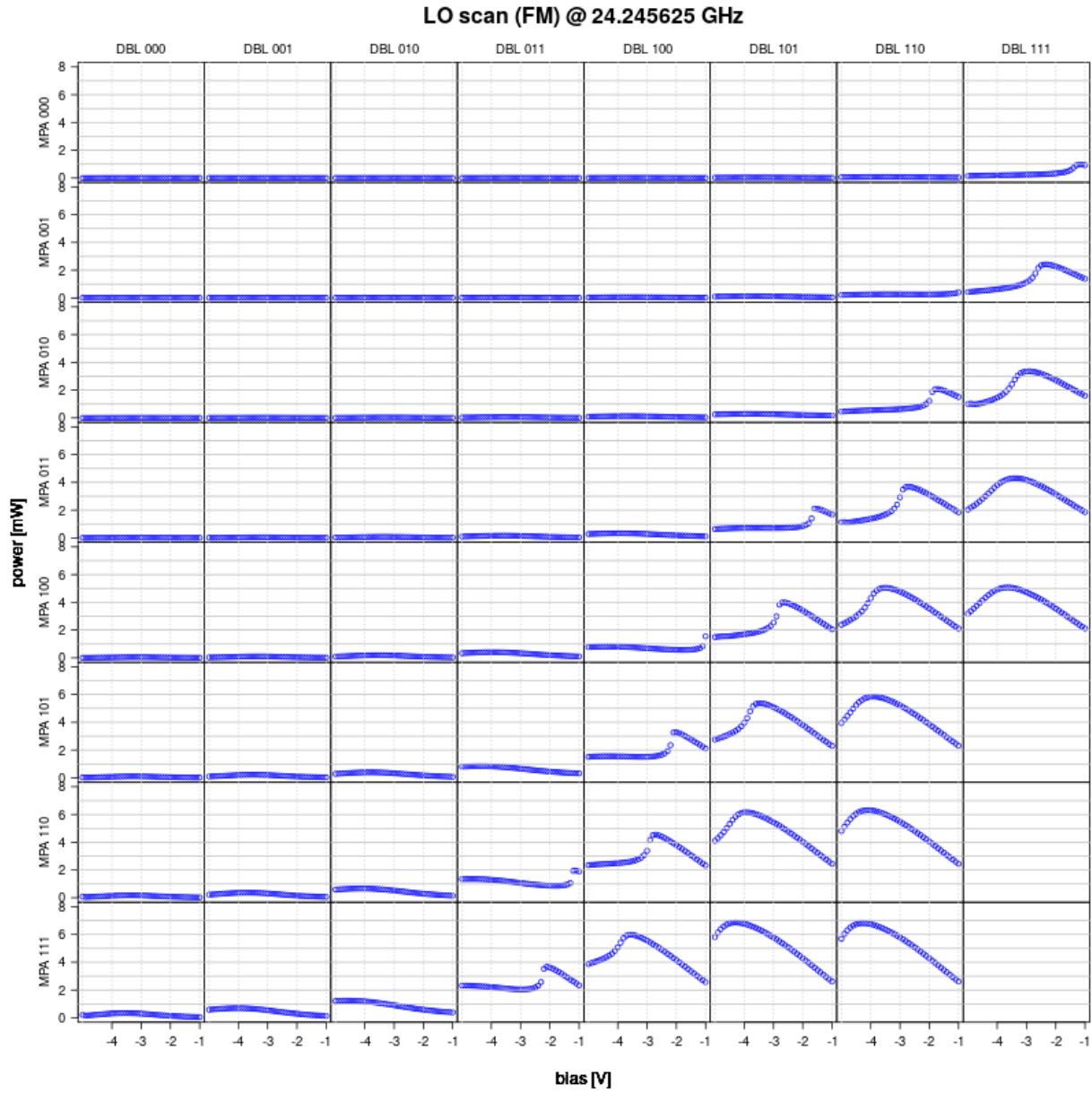


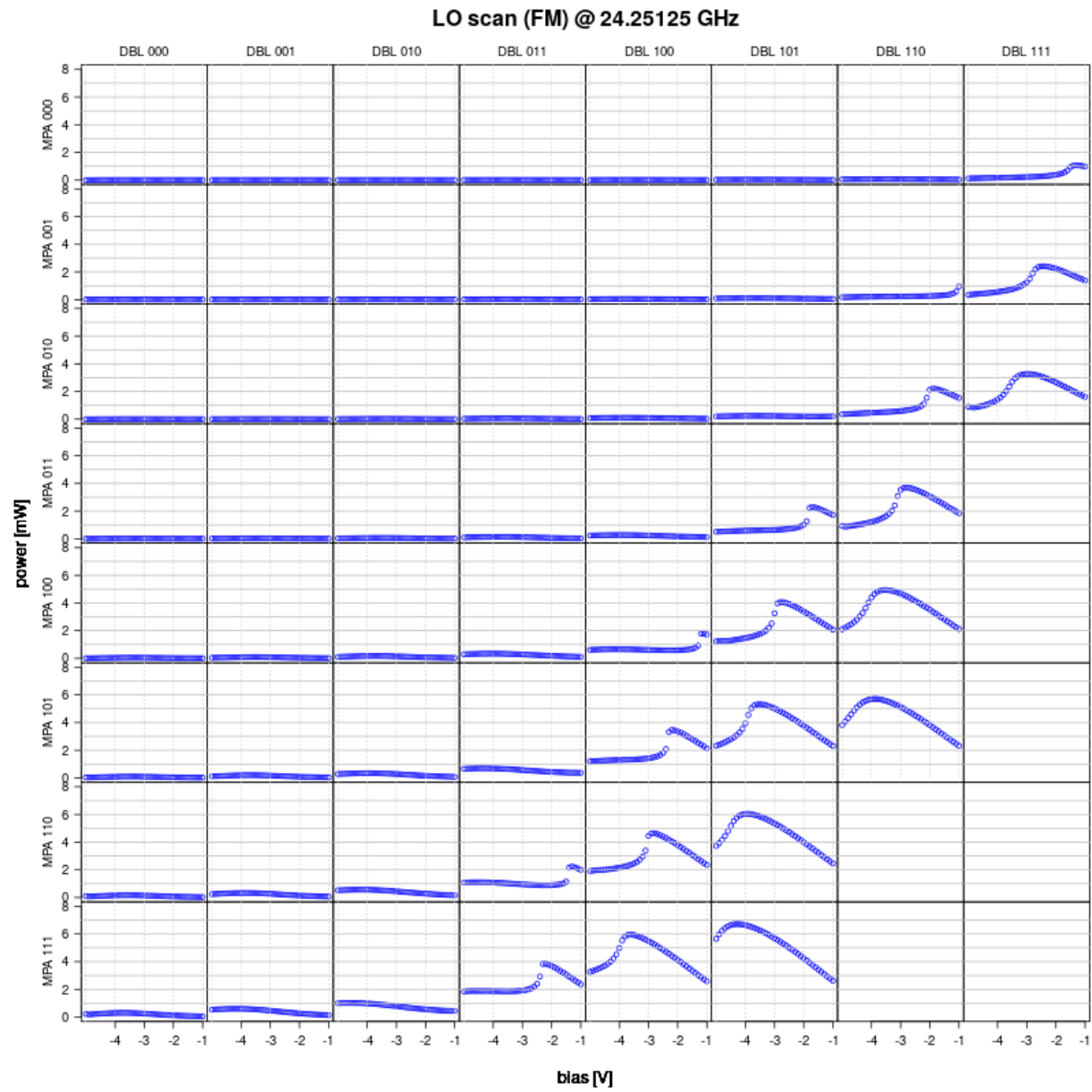
LO scan (FM) @ 24.234375 GHz

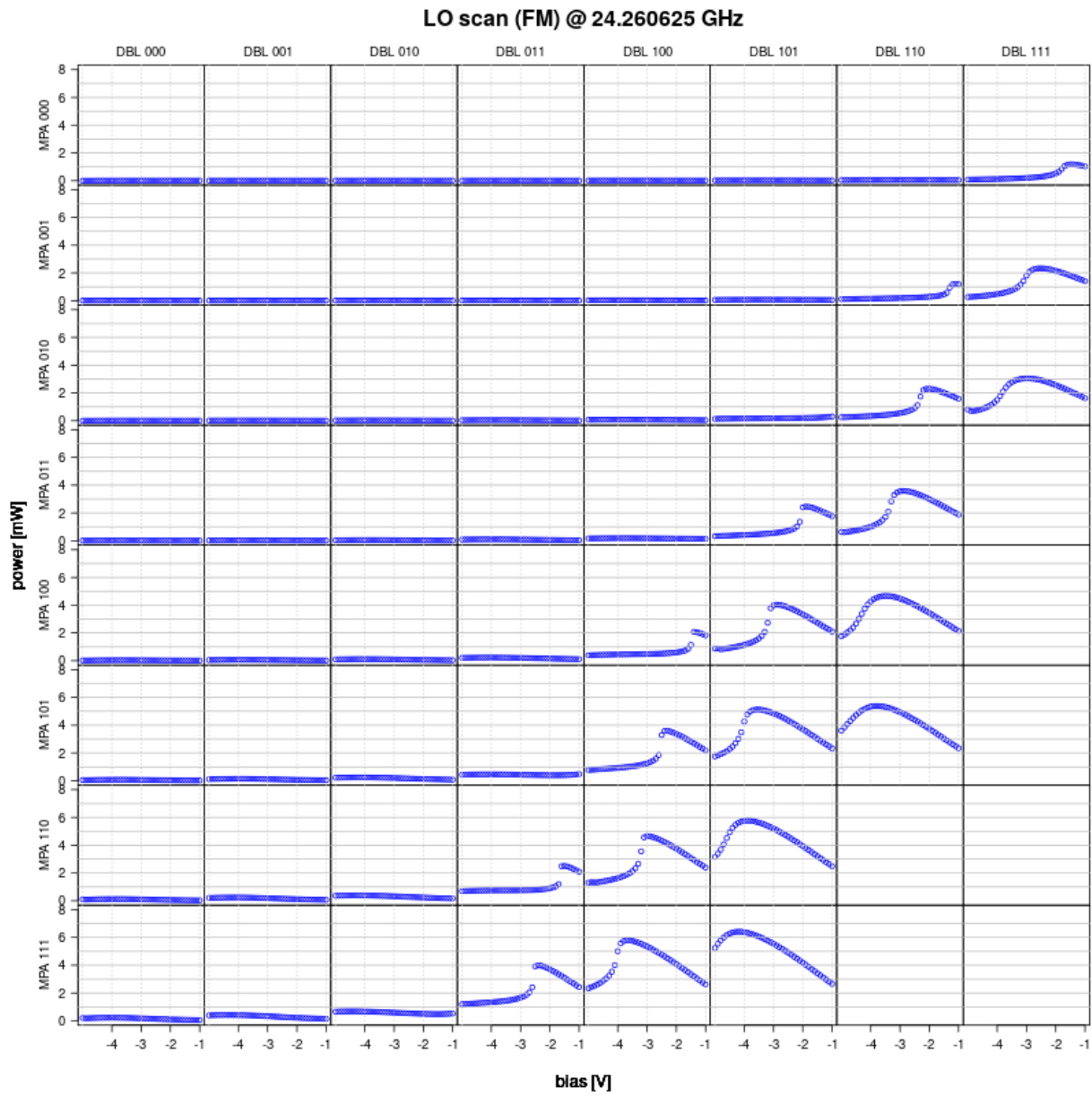


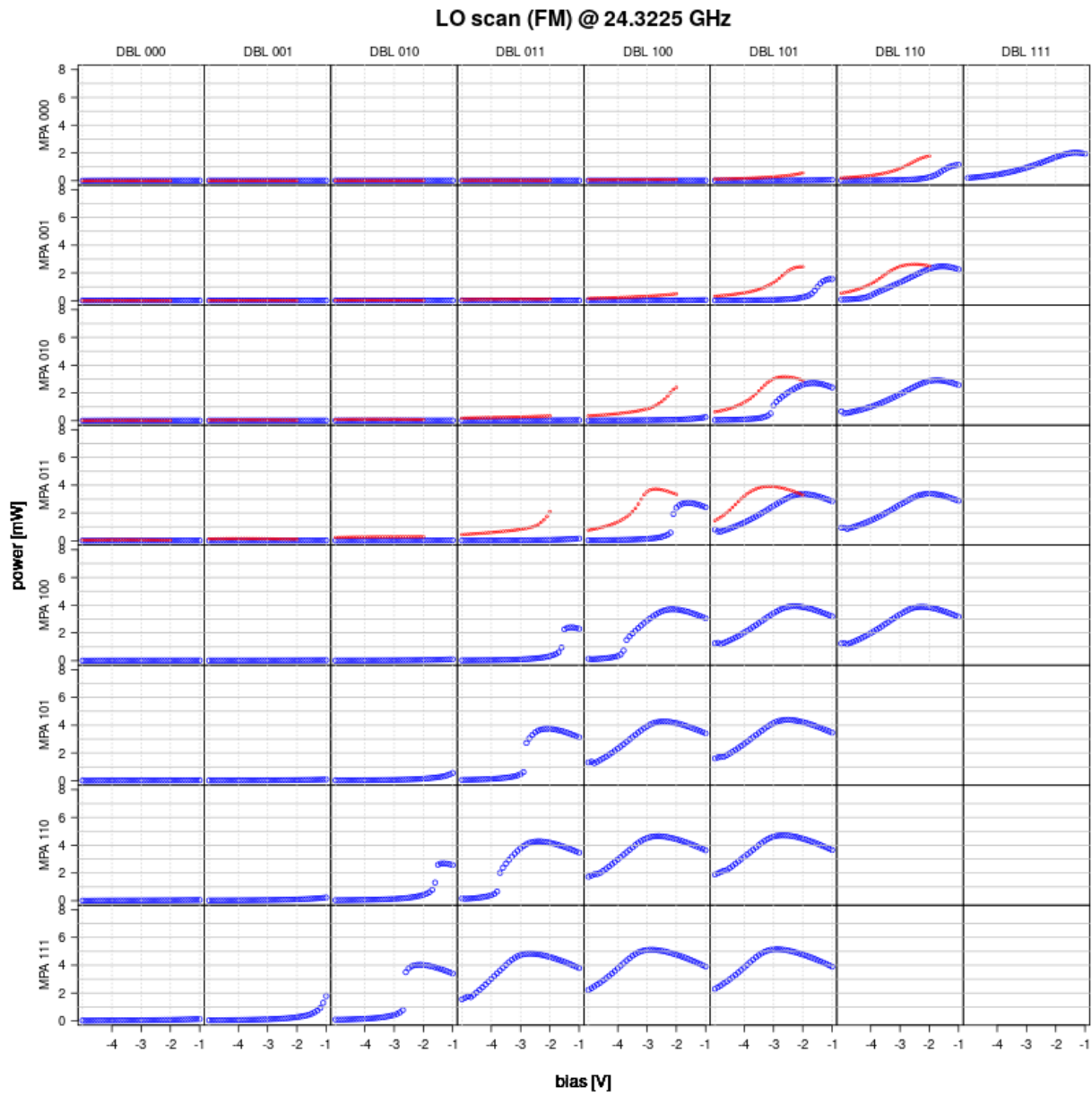




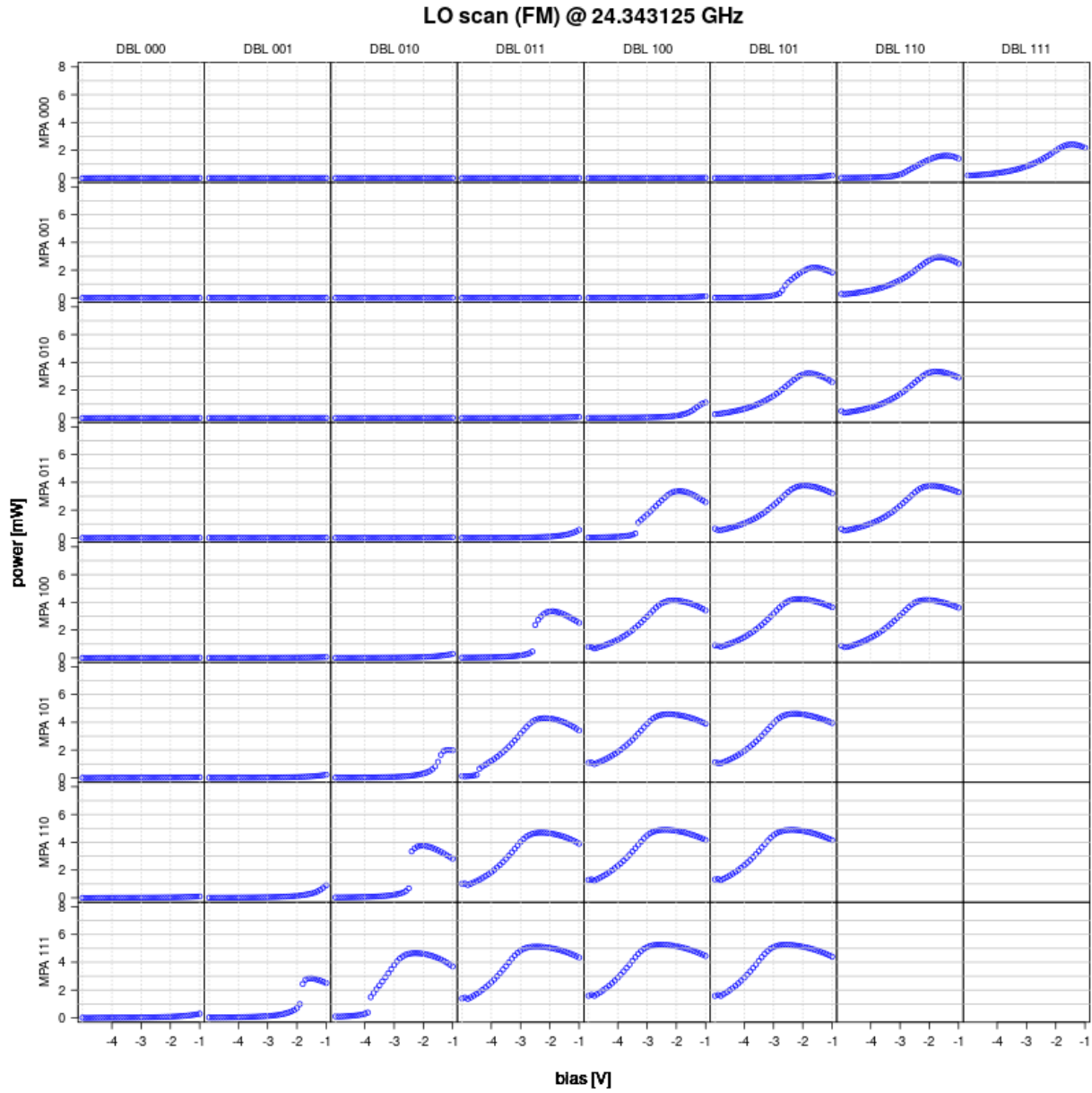




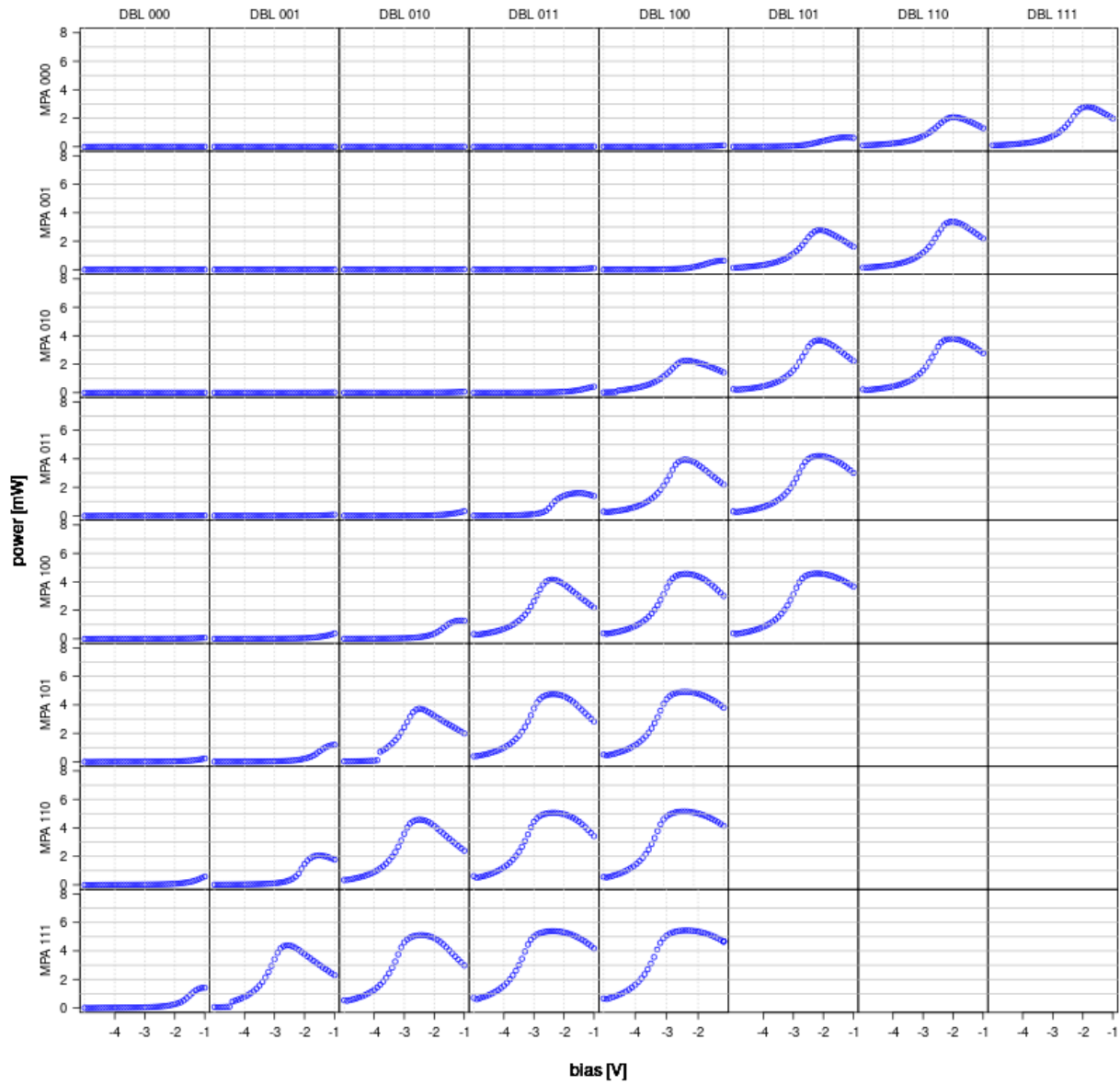


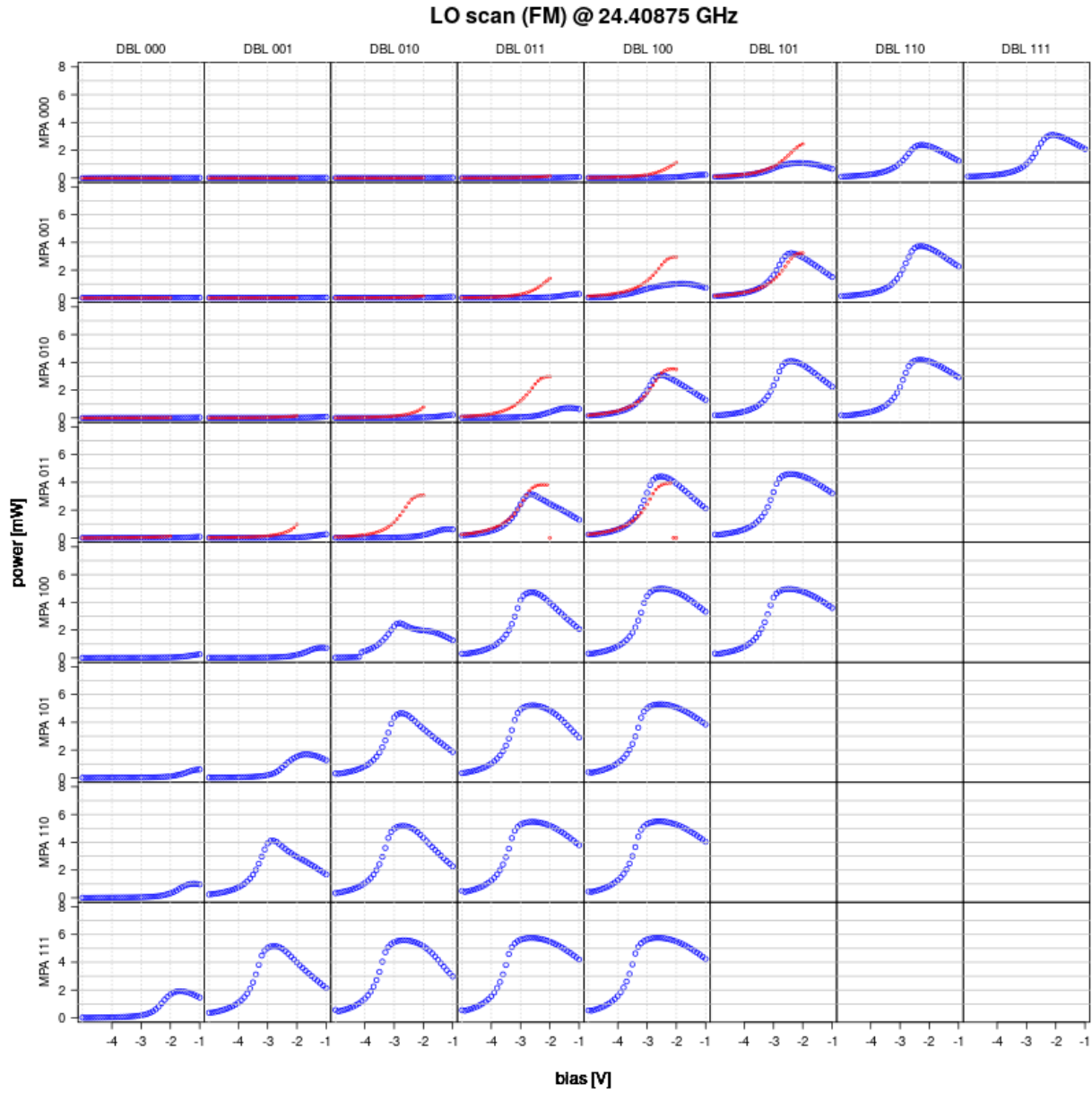


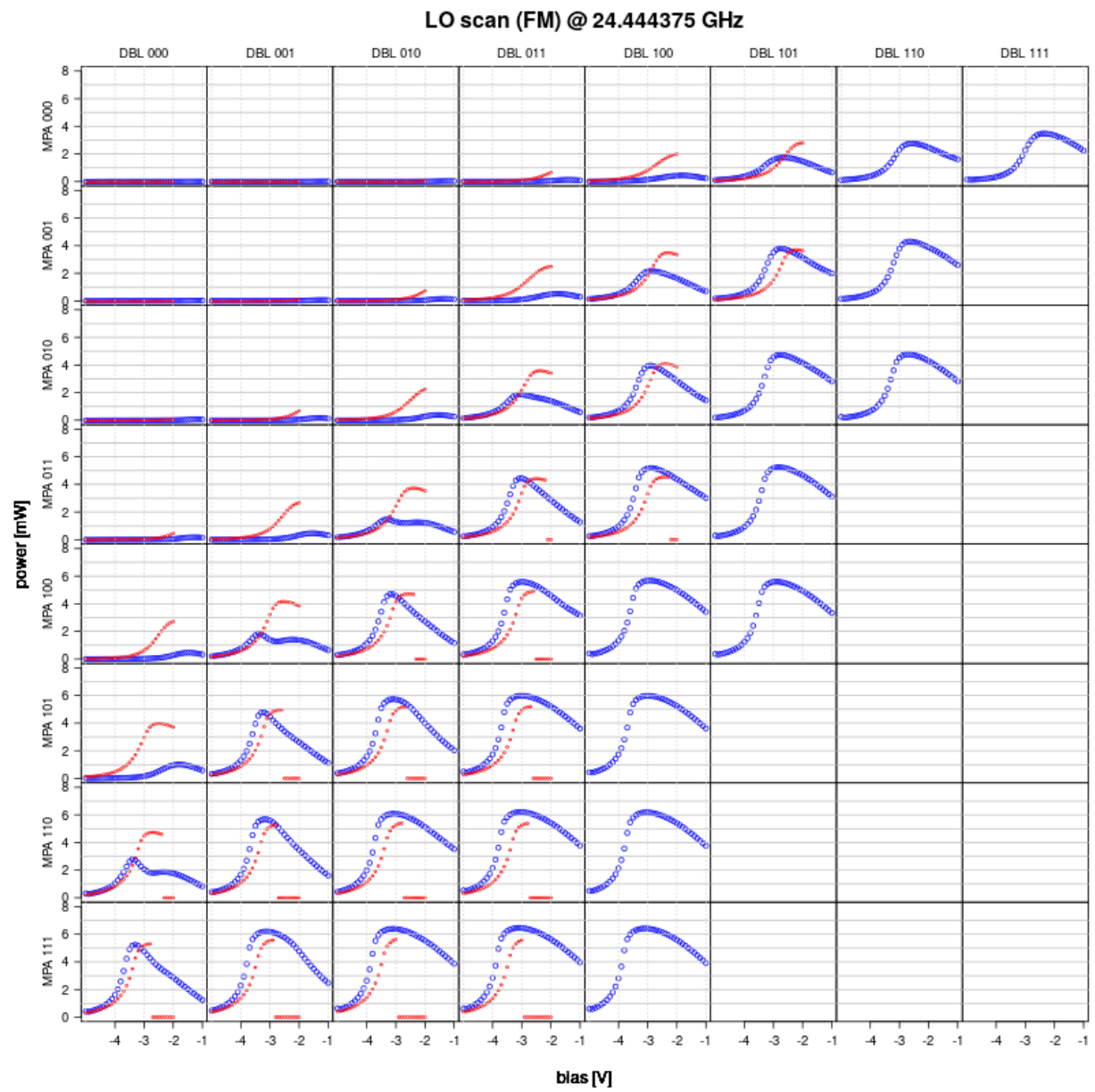


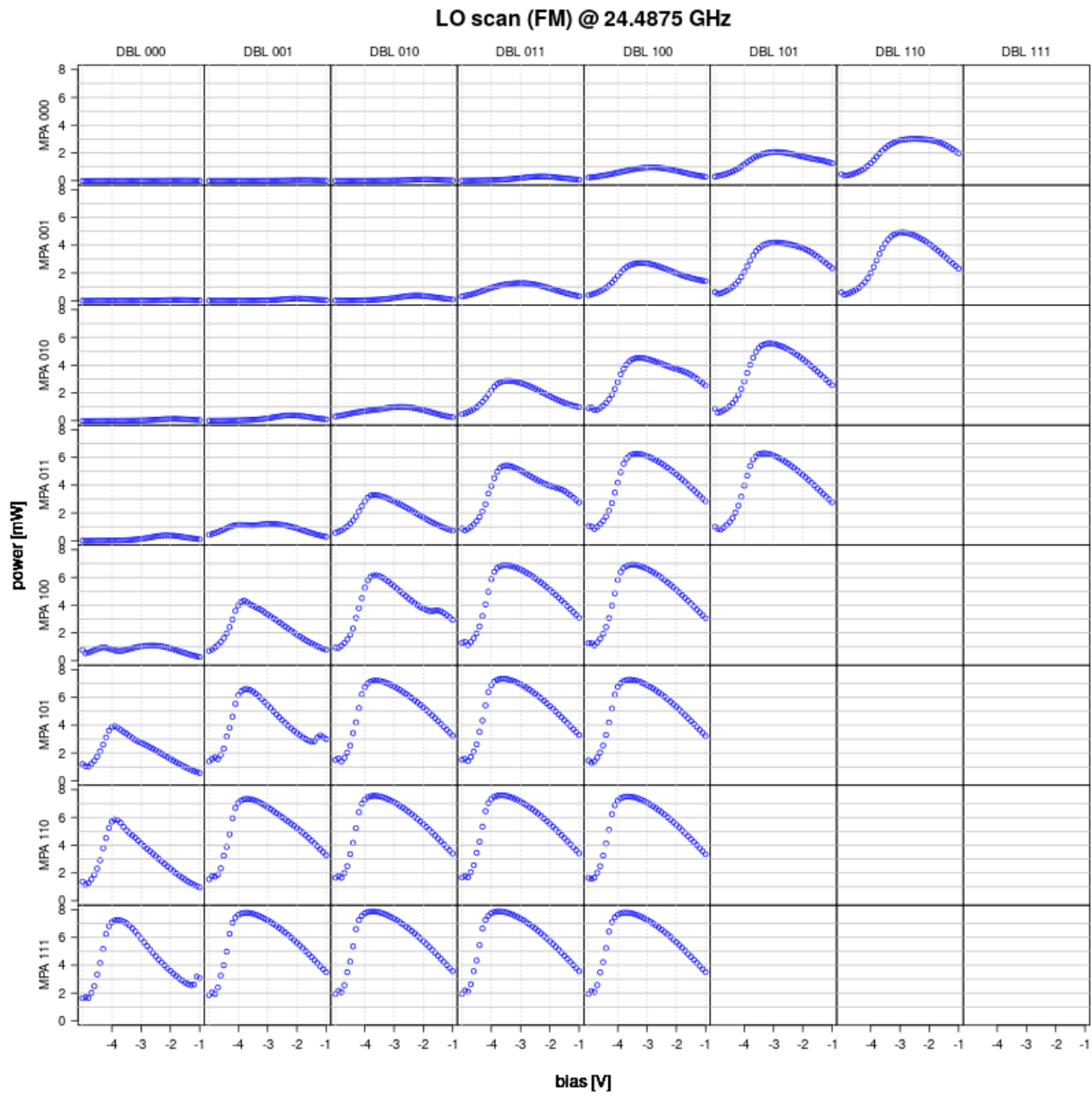


LO scan (FM) @ 24.3825 GHz

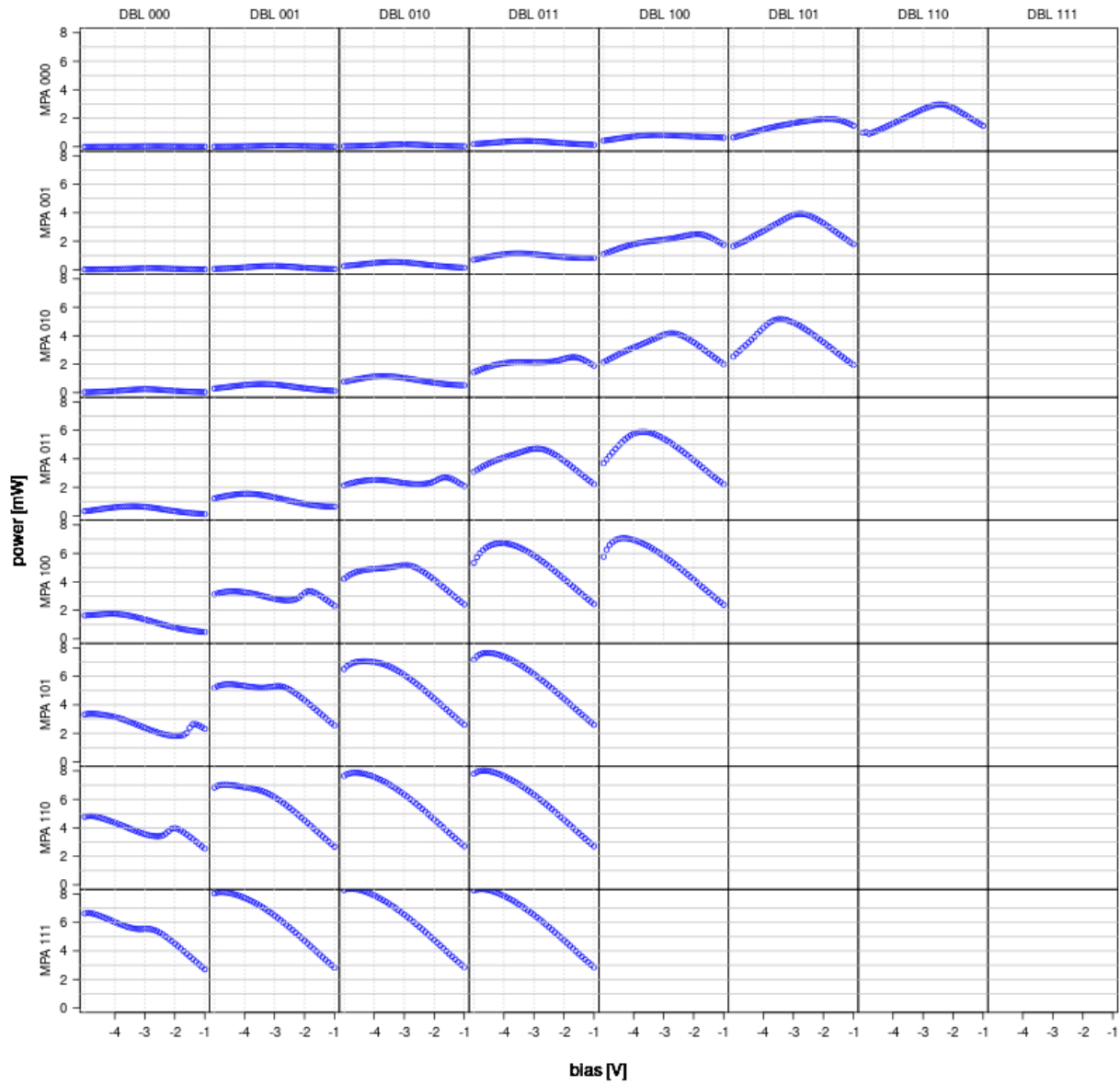


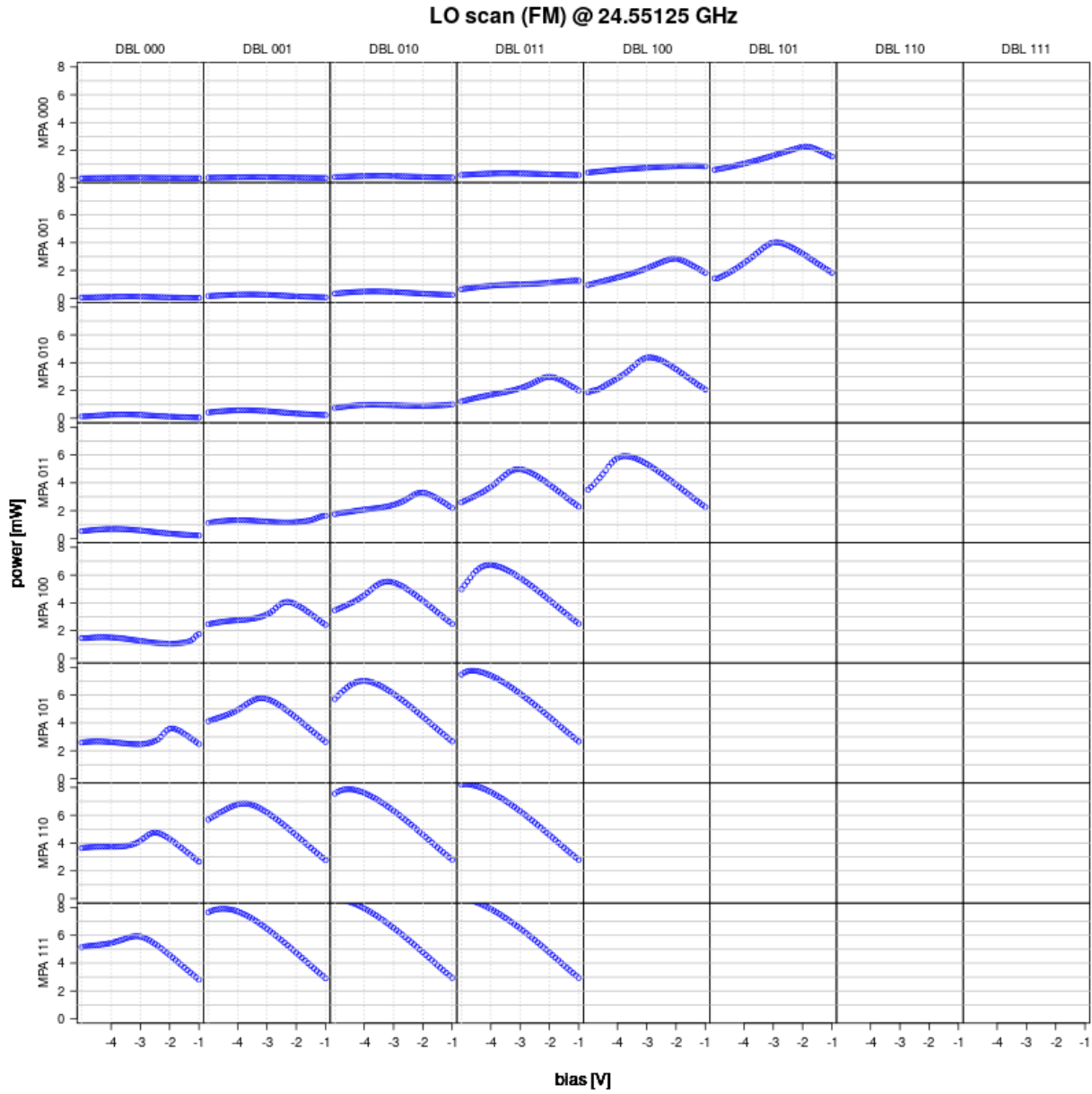


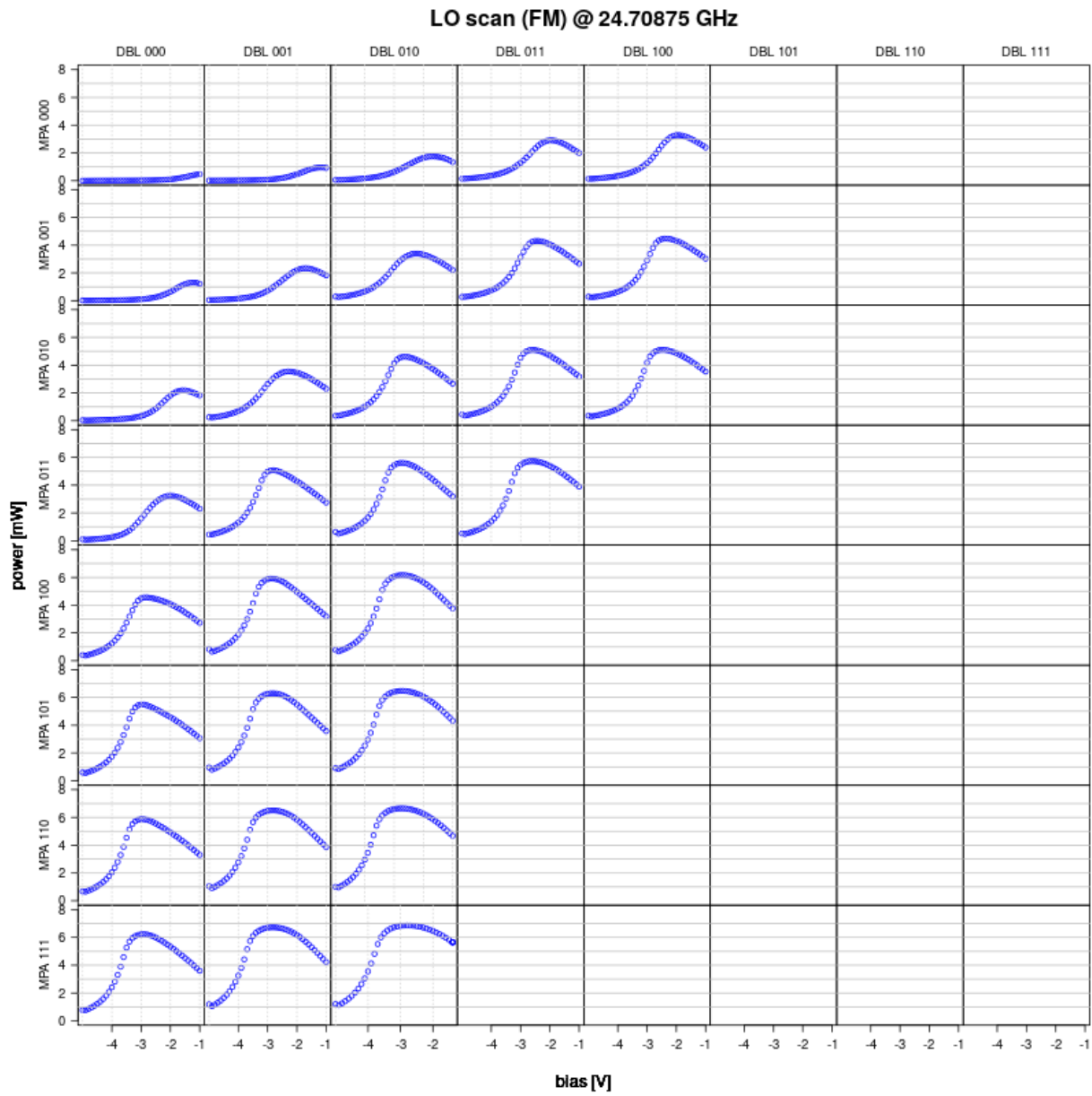




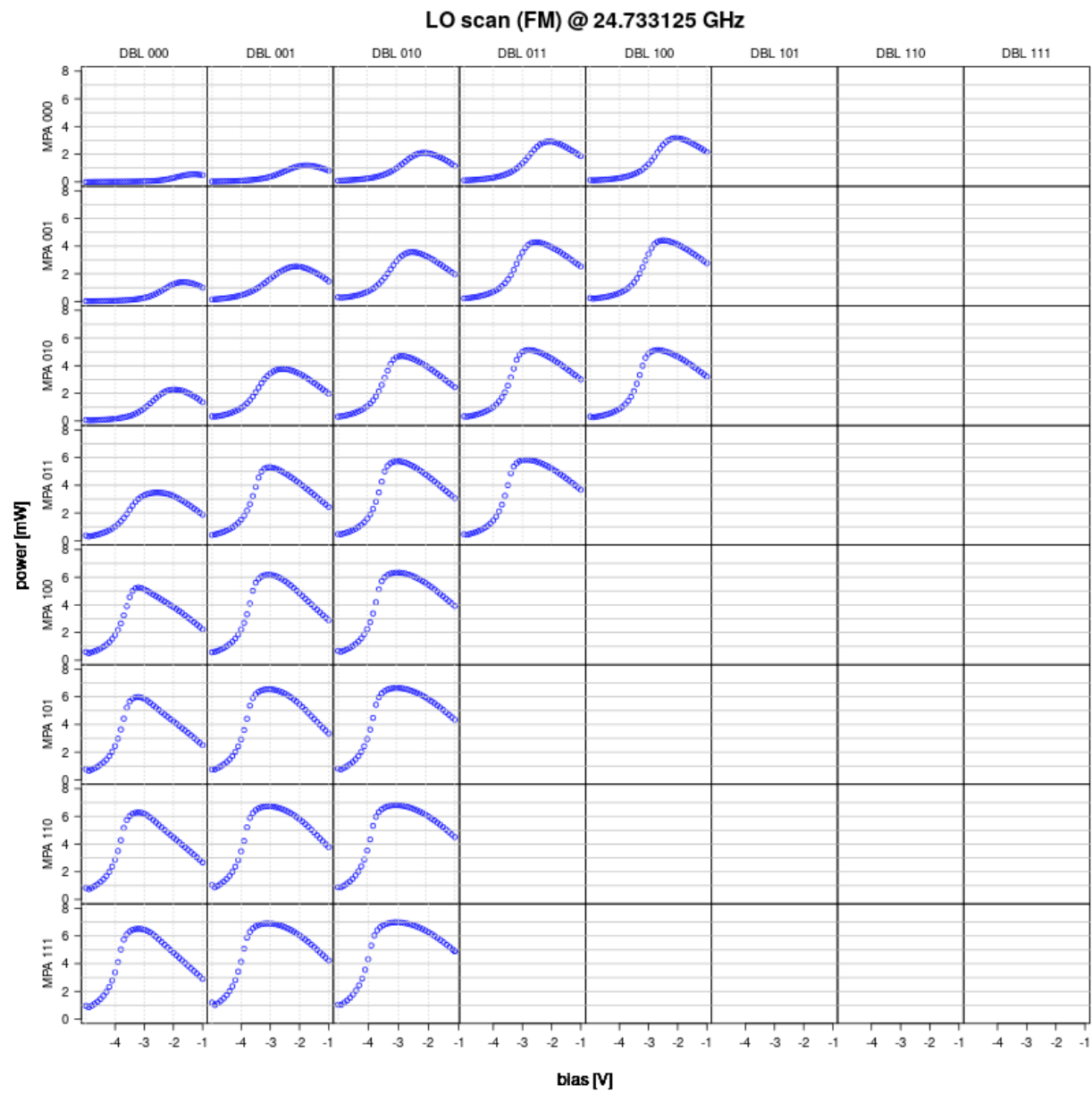
LO scan (FM) @ 24.5325 GHz

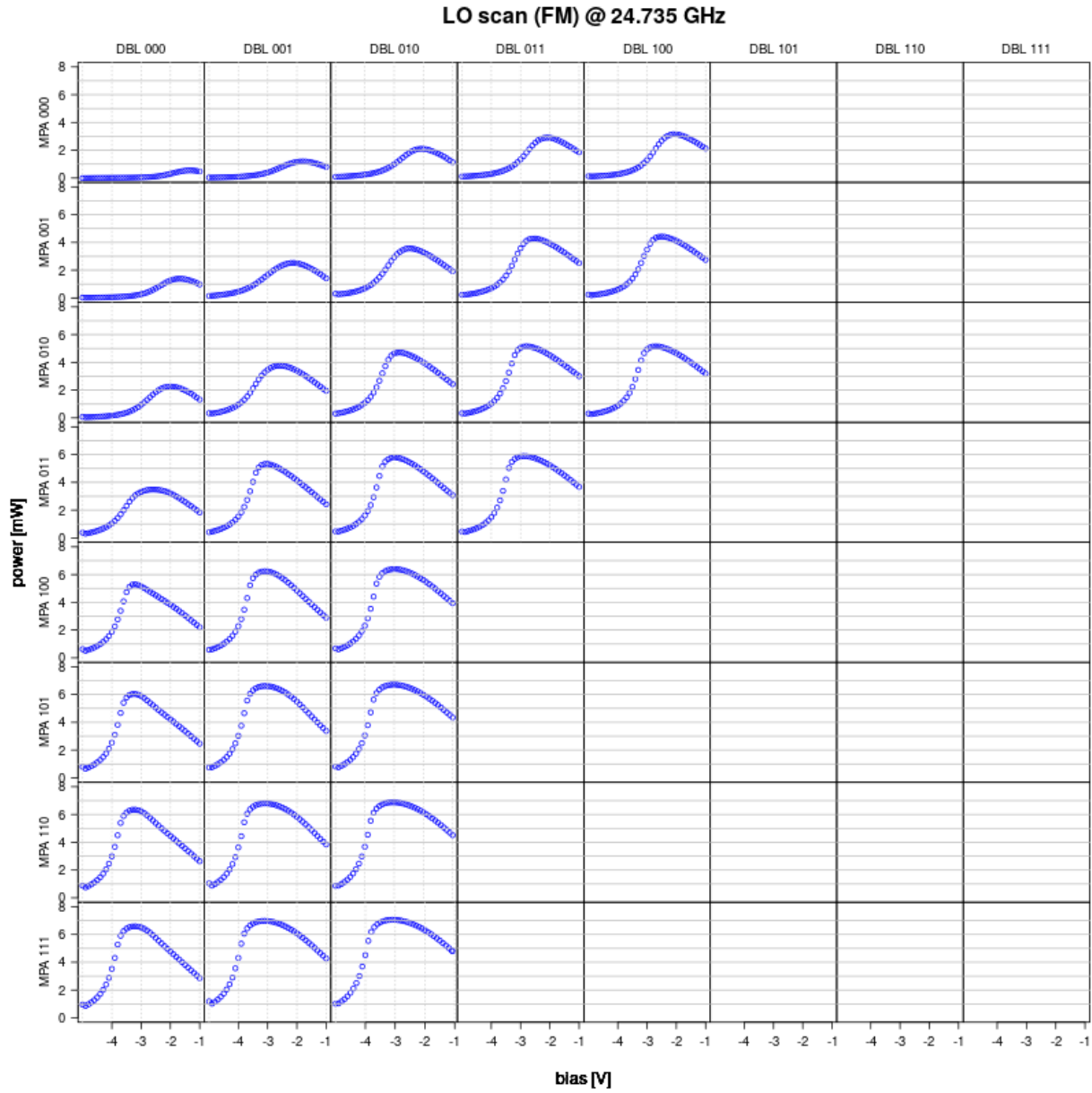


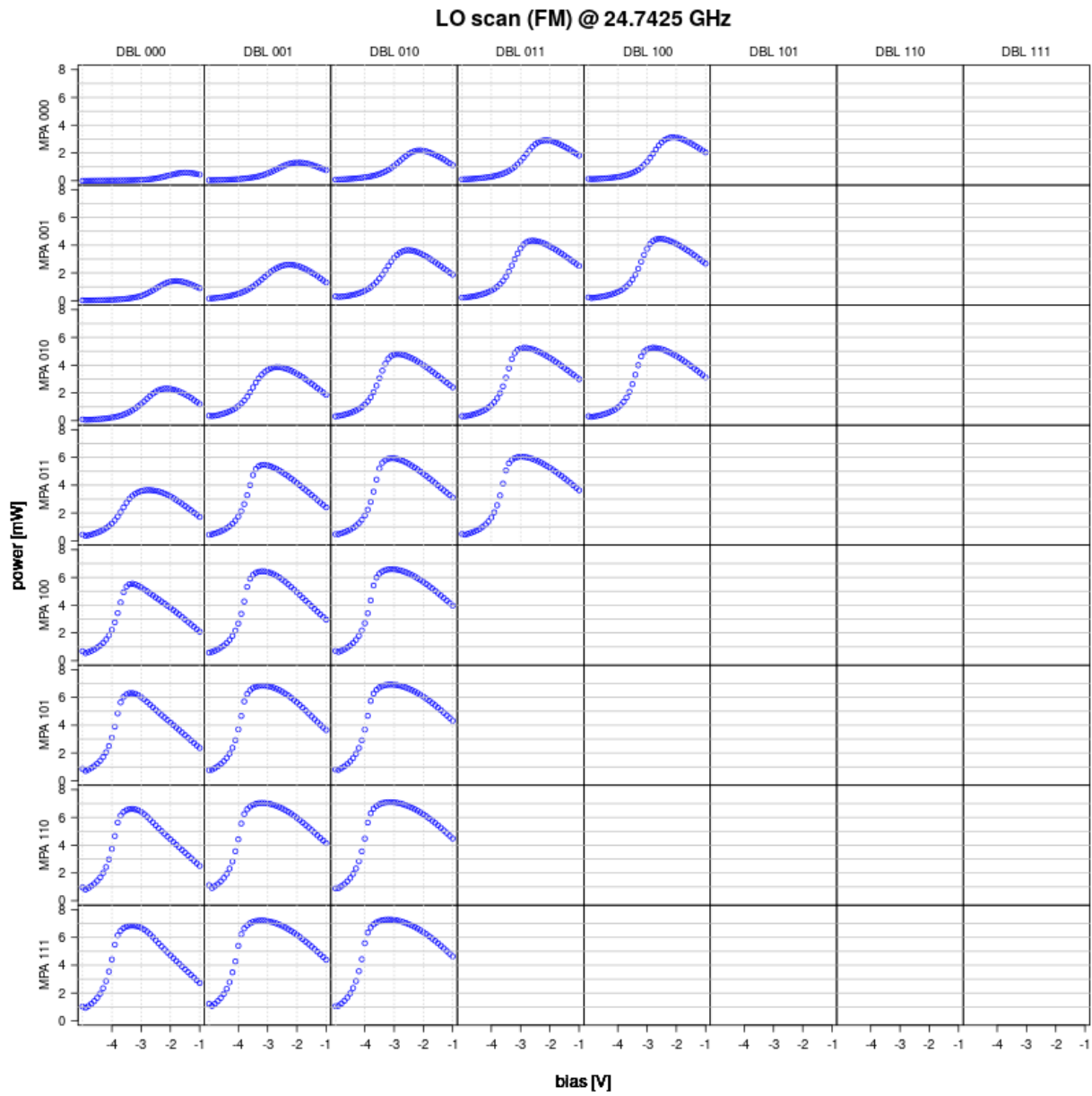


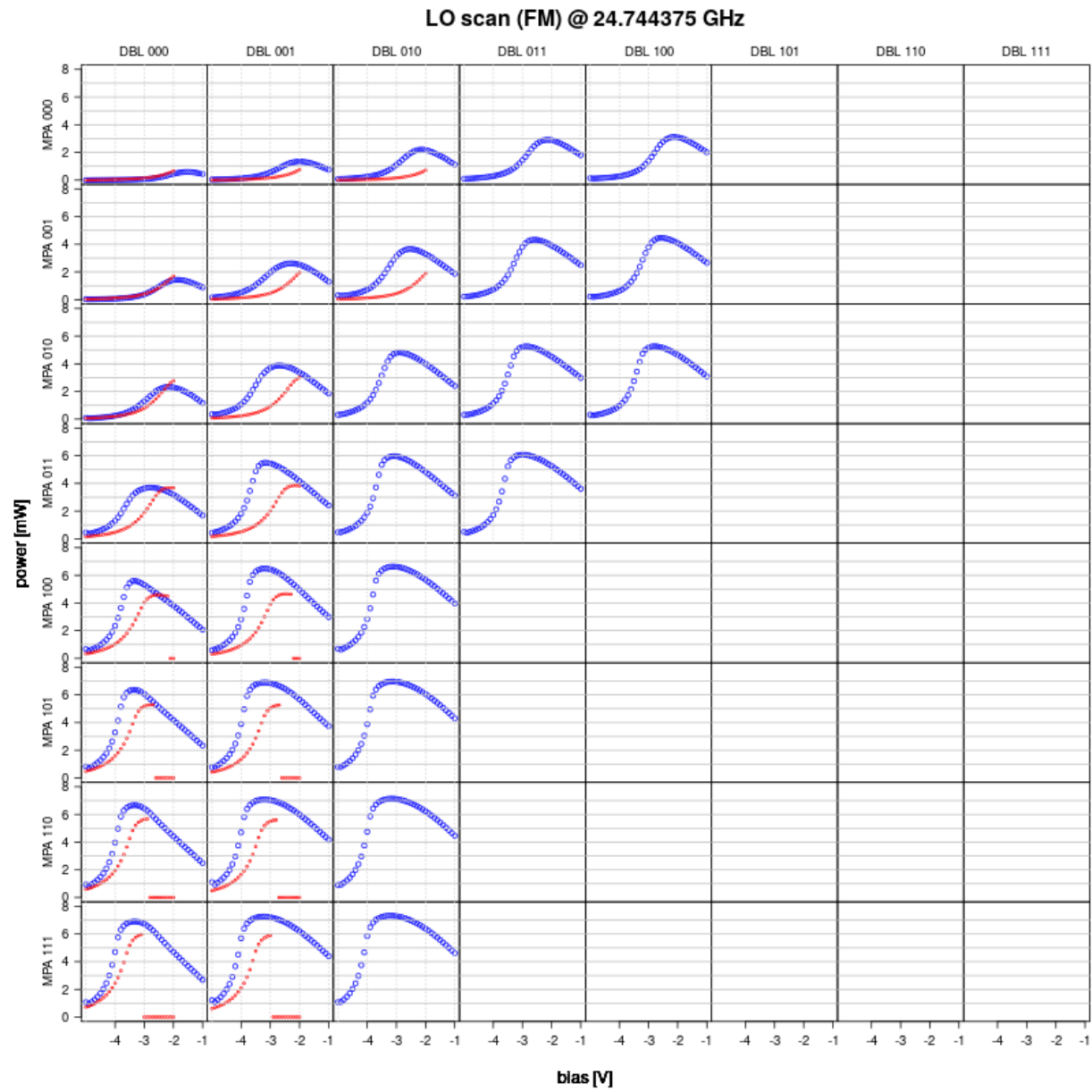


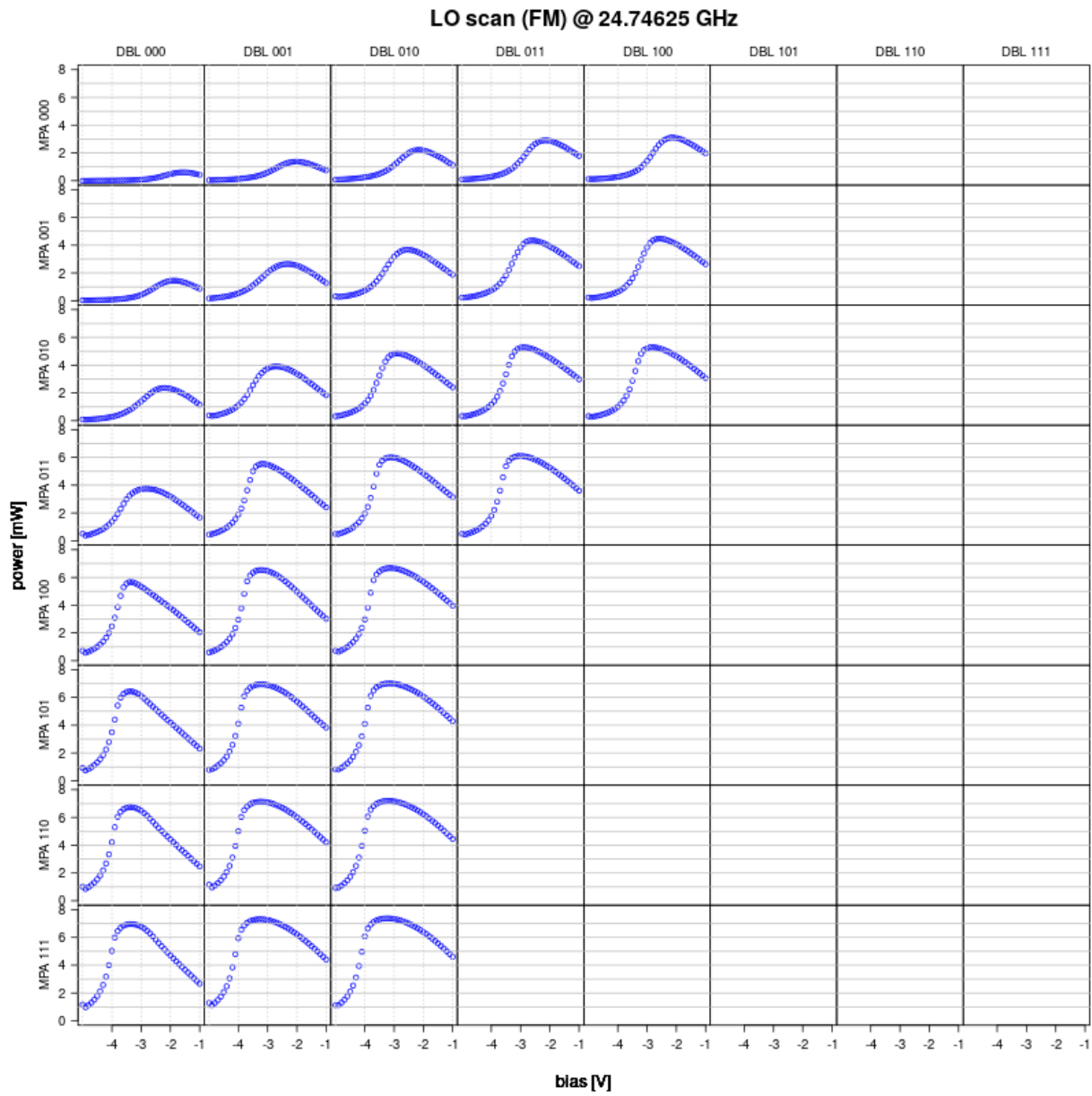


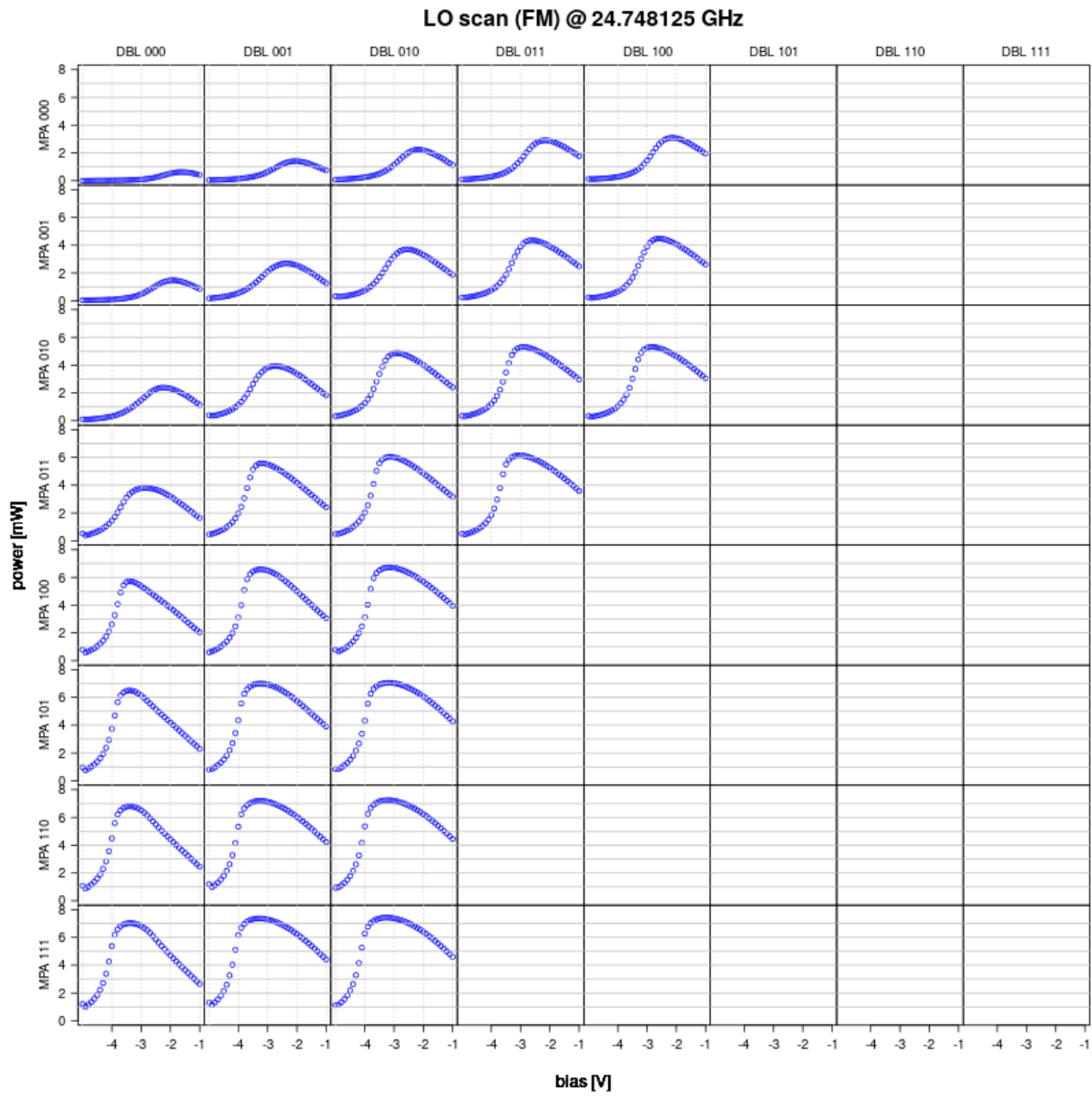


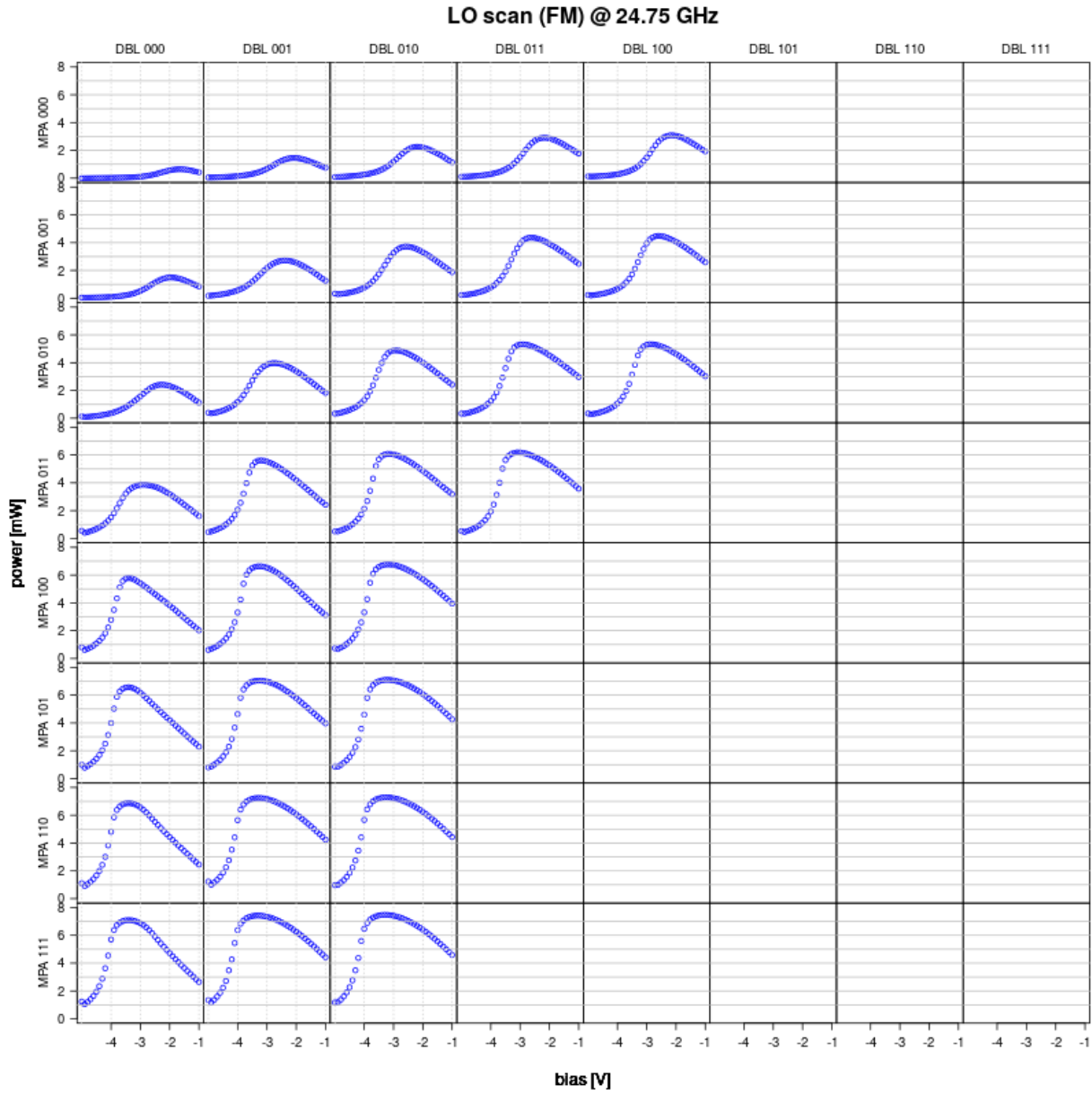


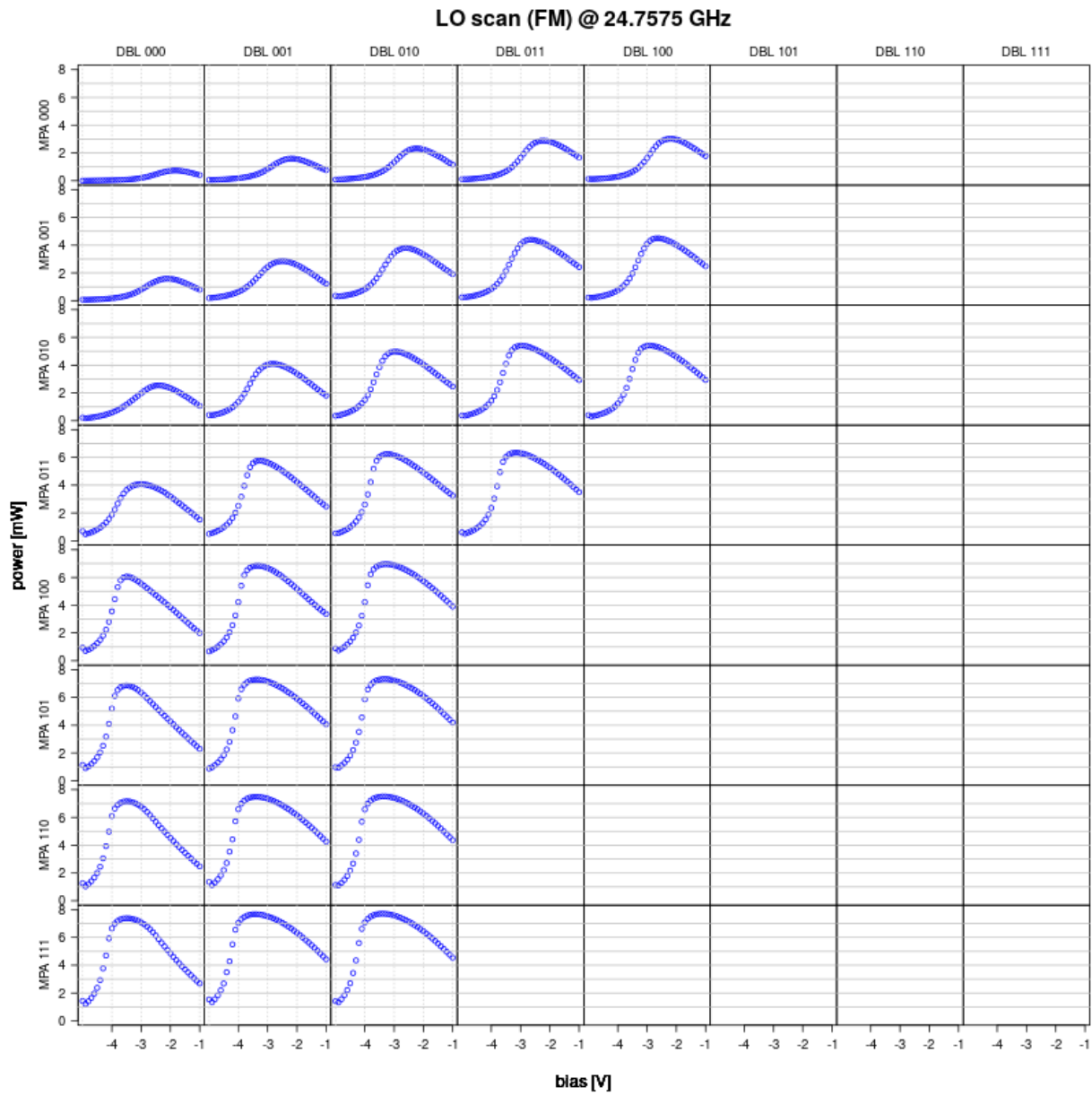






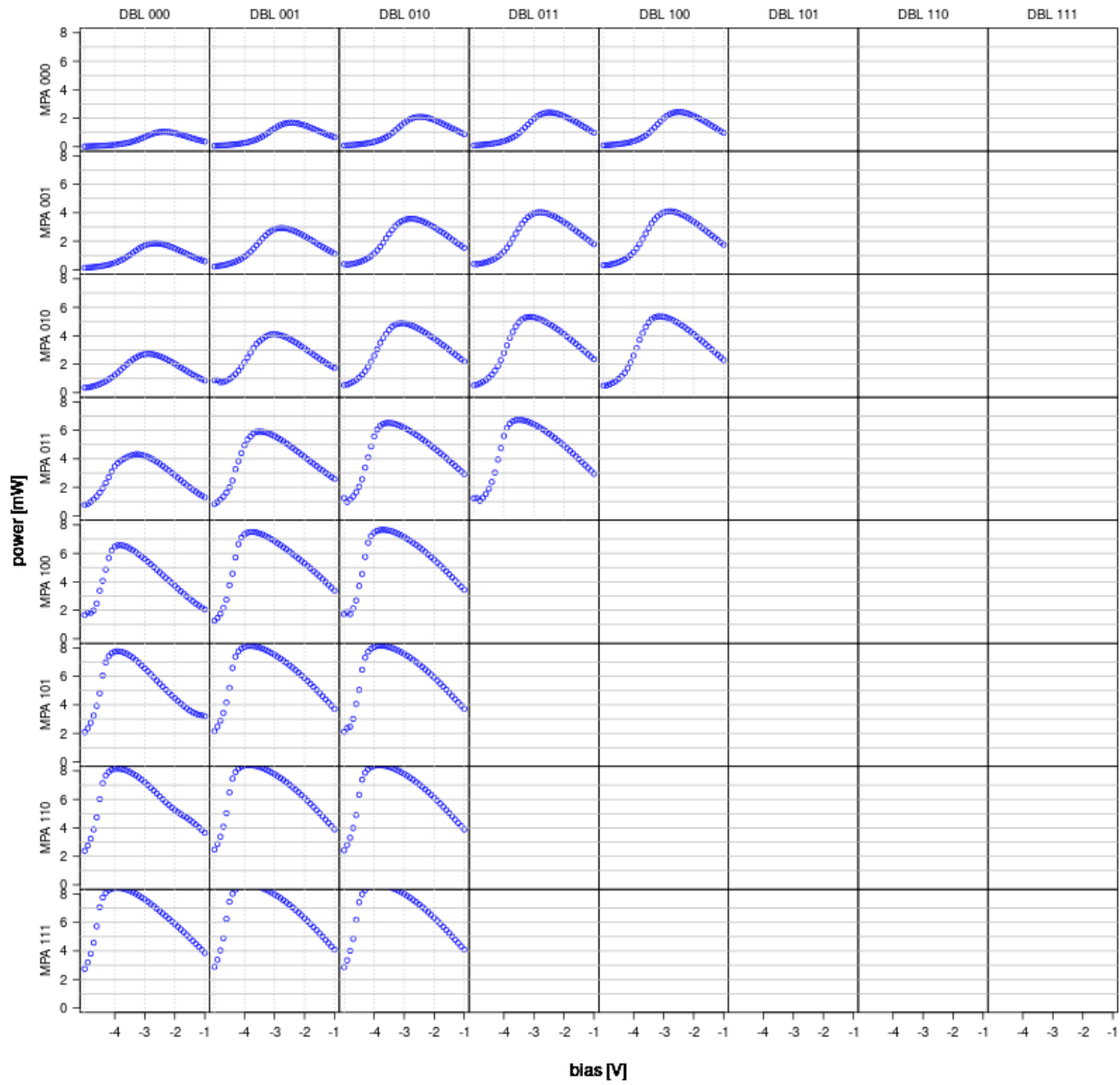


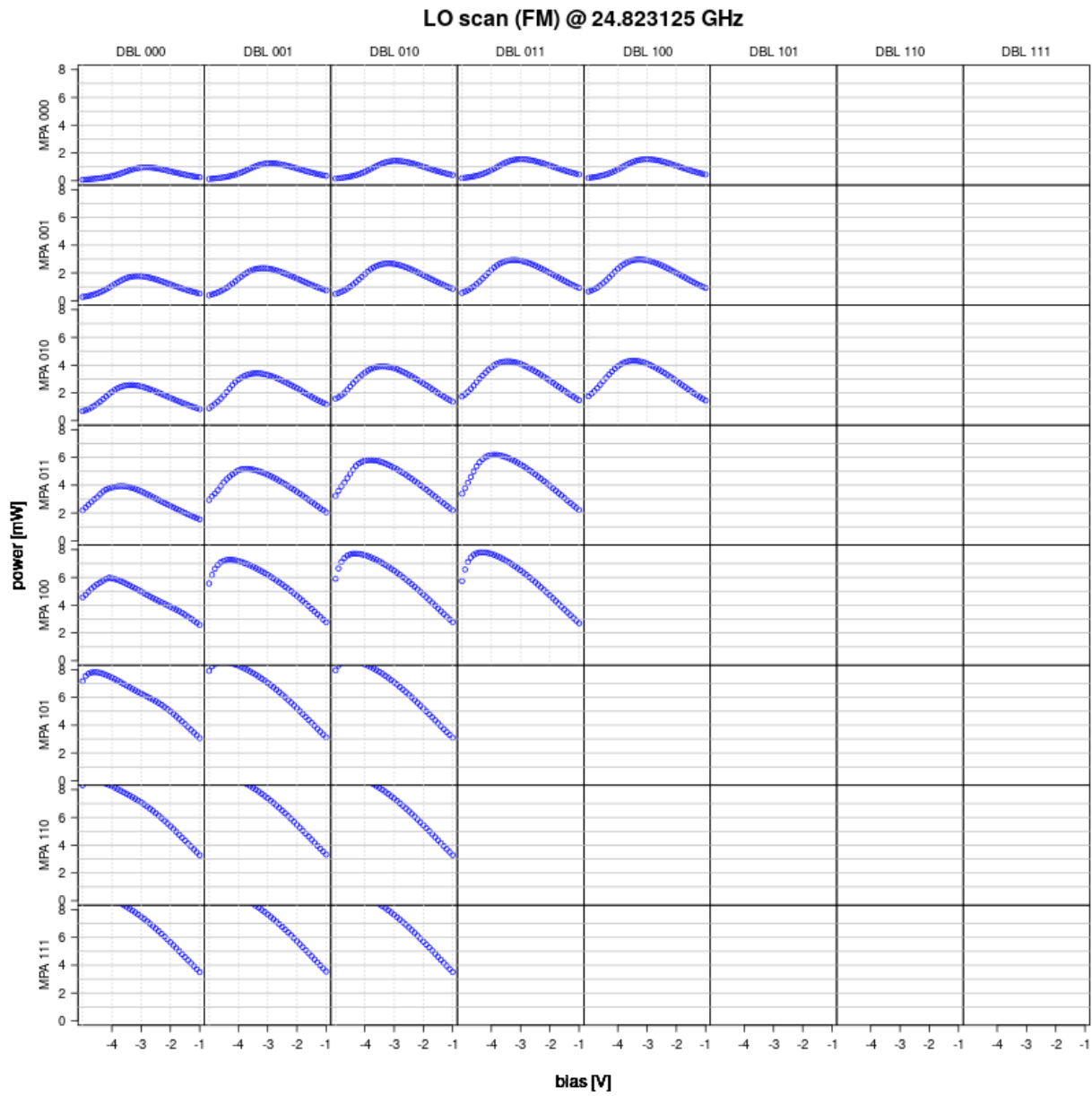


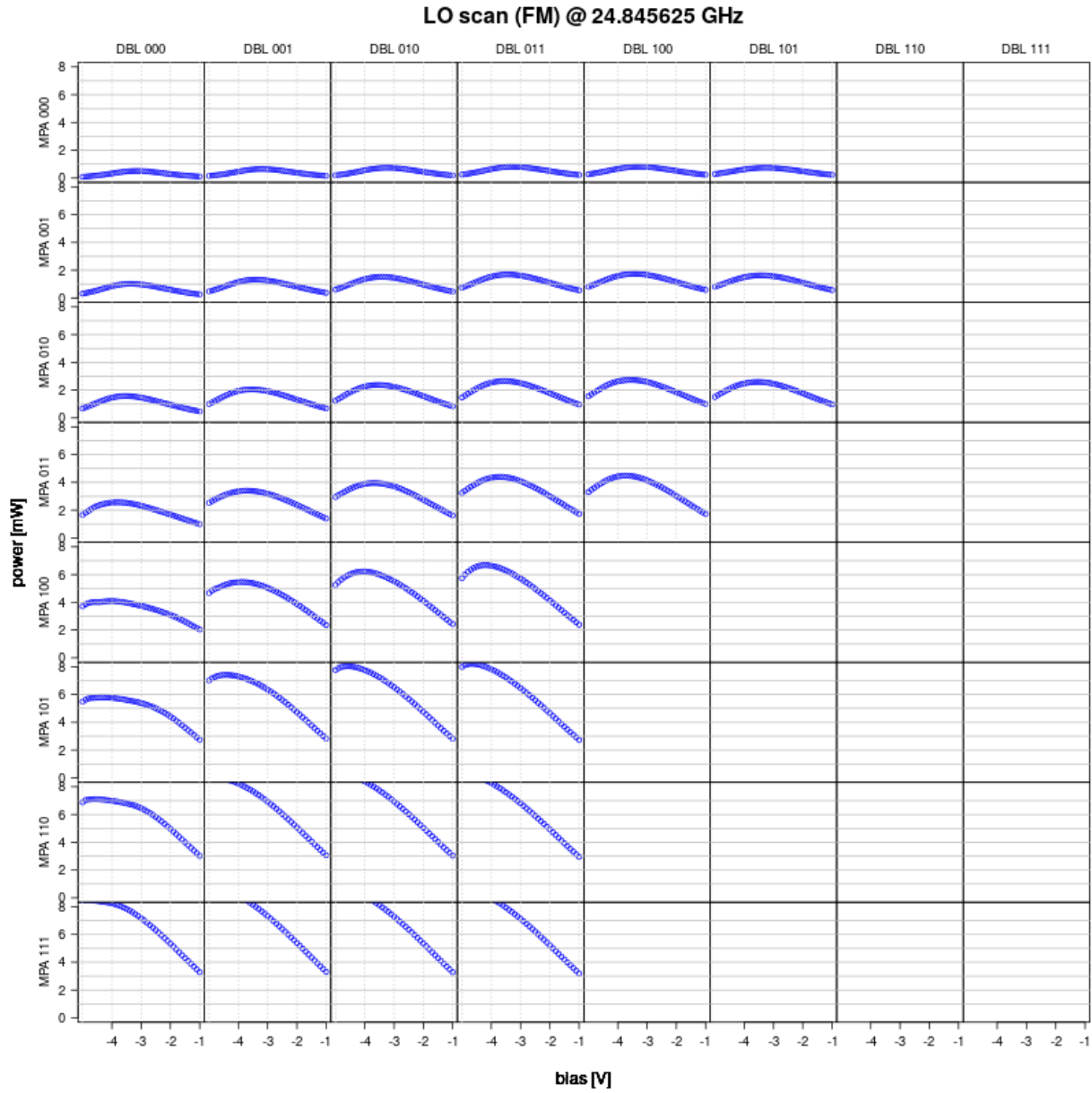


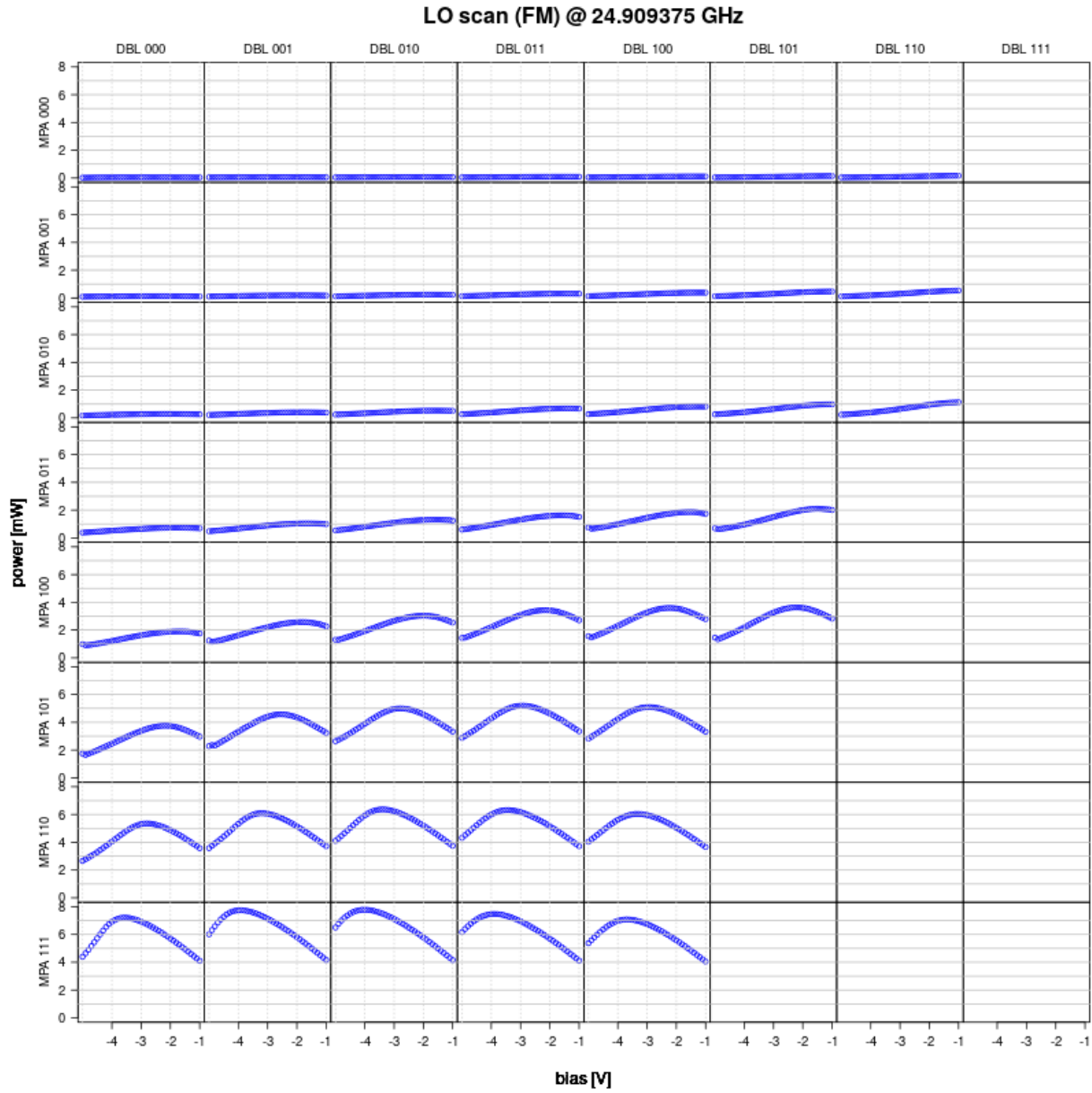


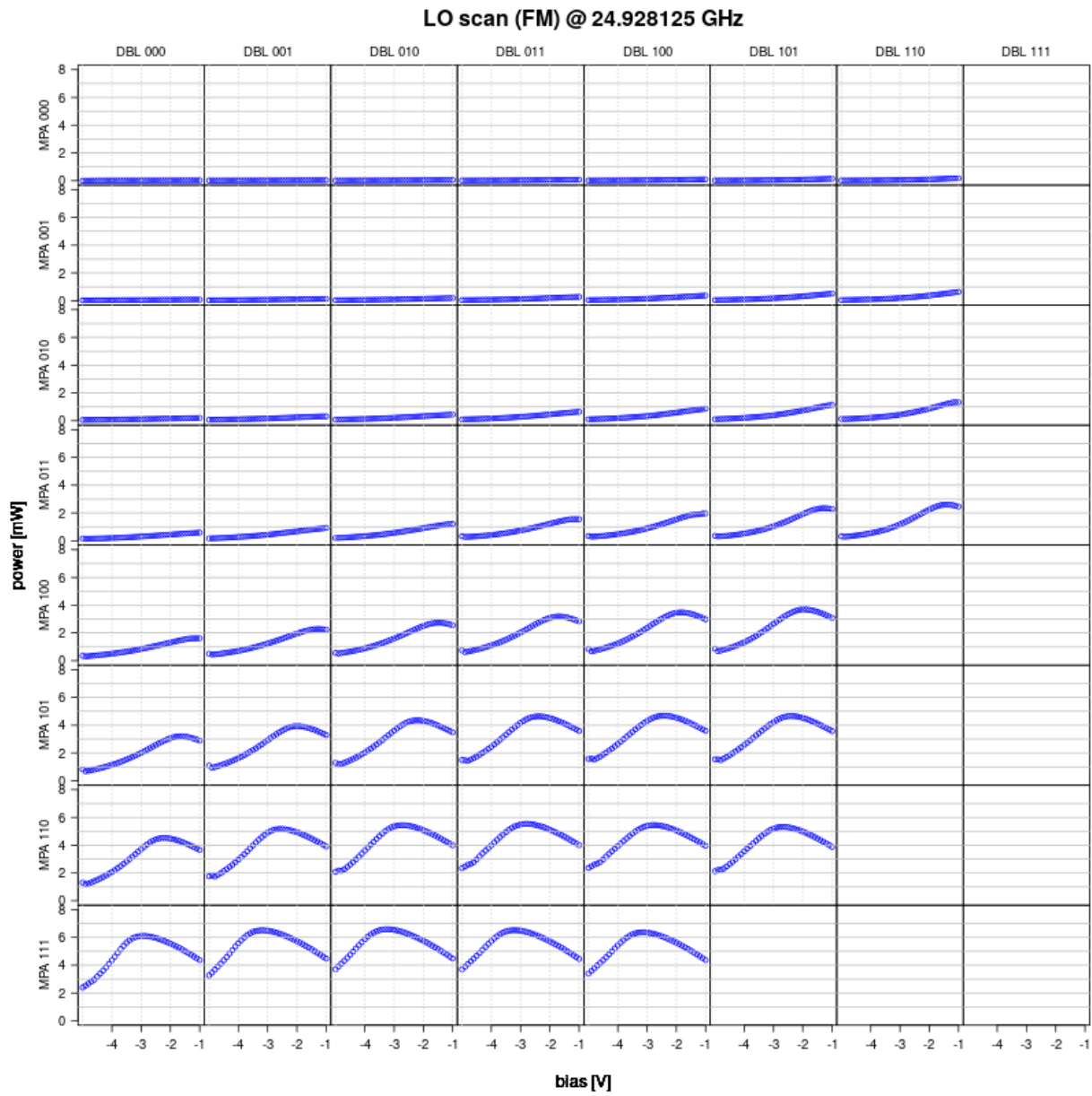
**LO scan (FM) @ 24.7875 GHz**

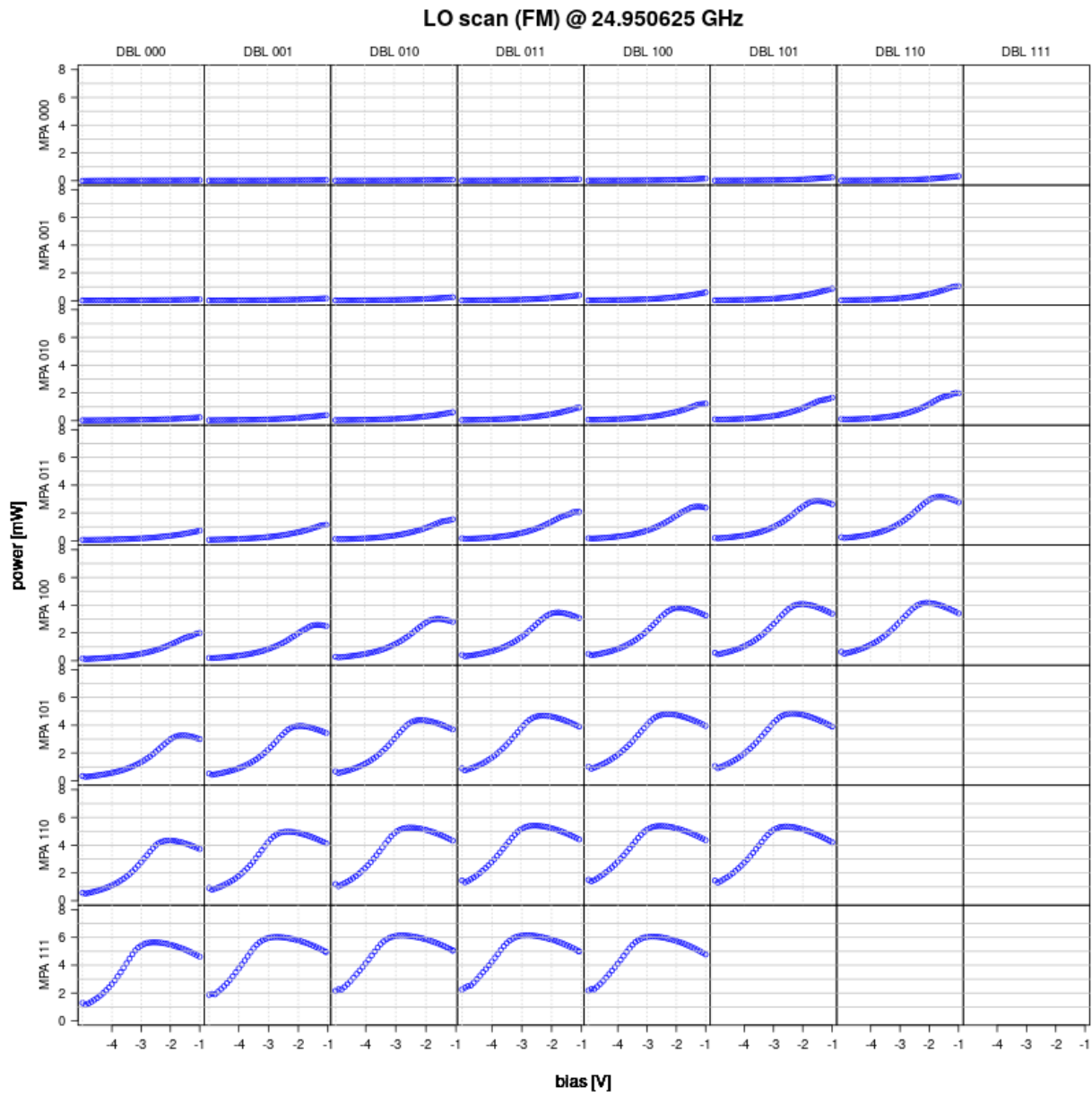




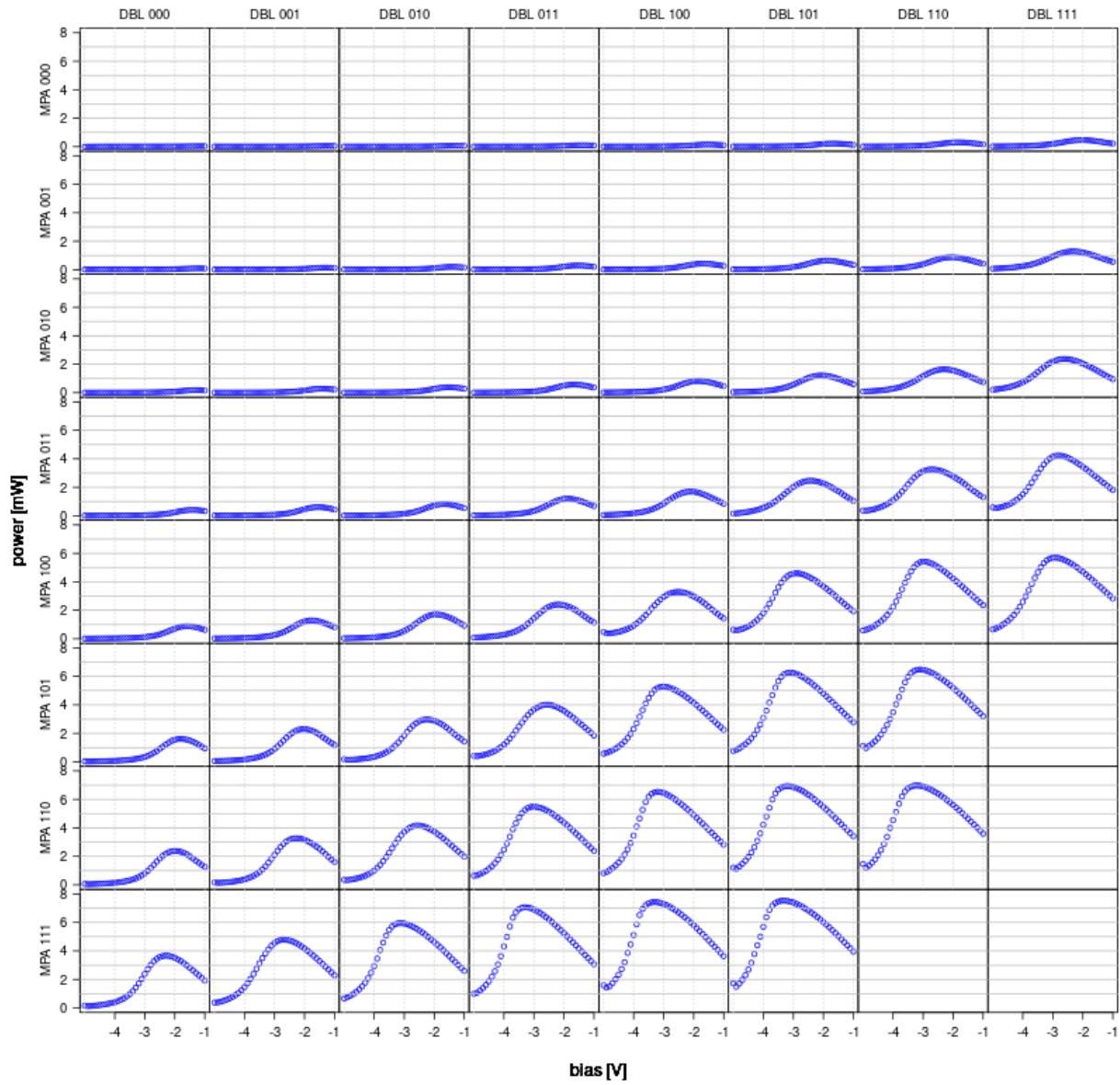


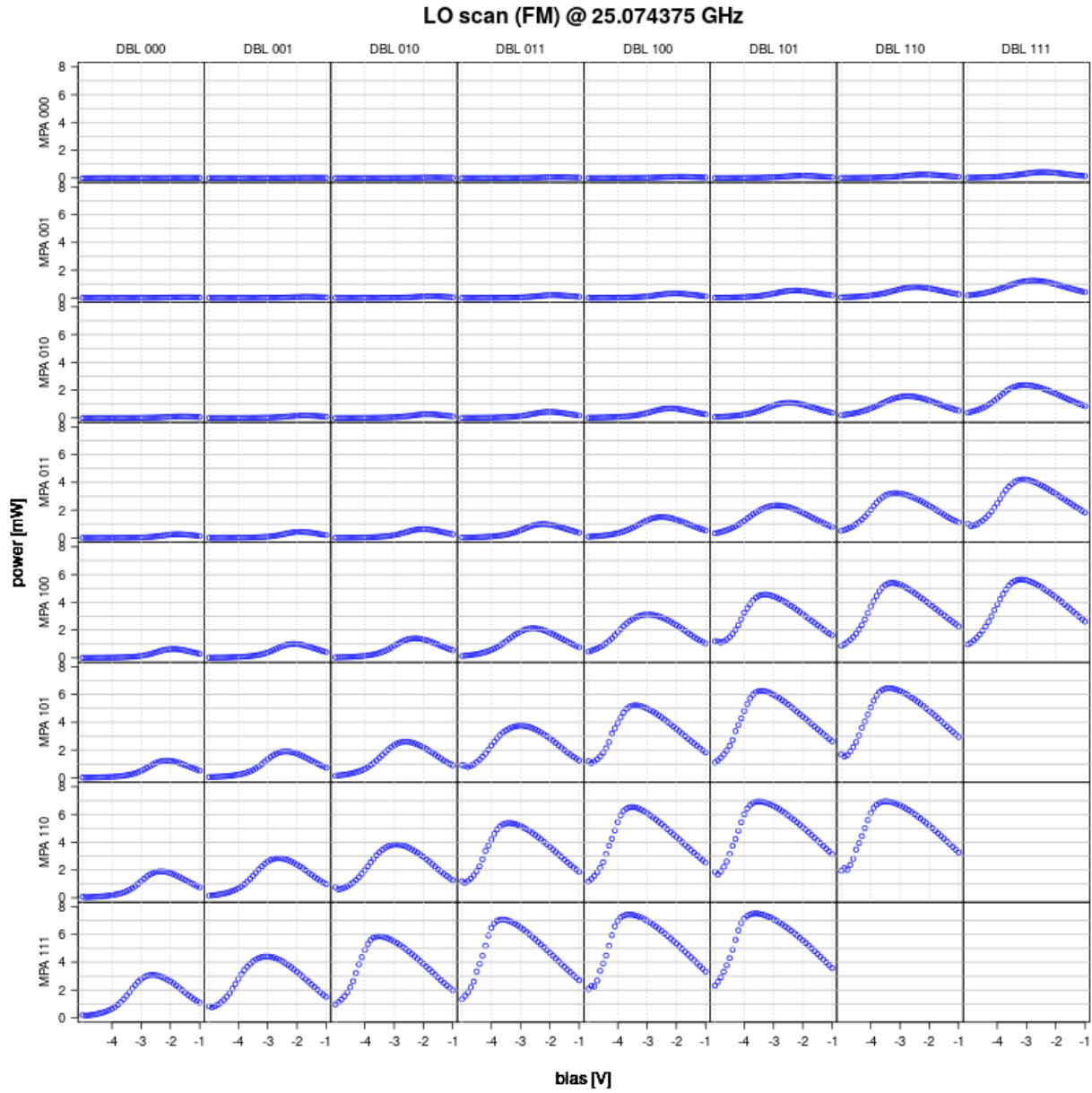






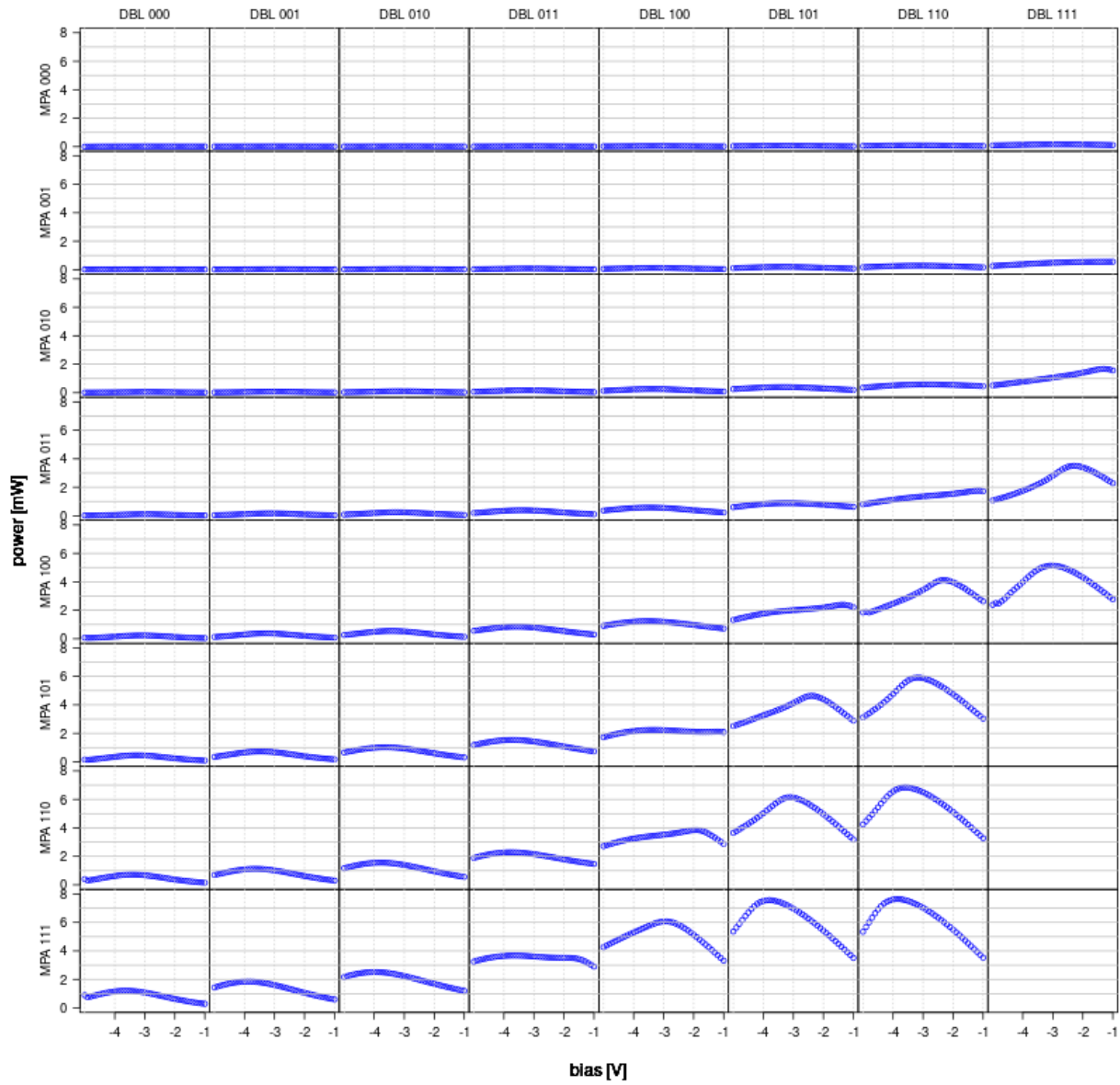
**LO scan (FM) @ 25.051875 GHz**

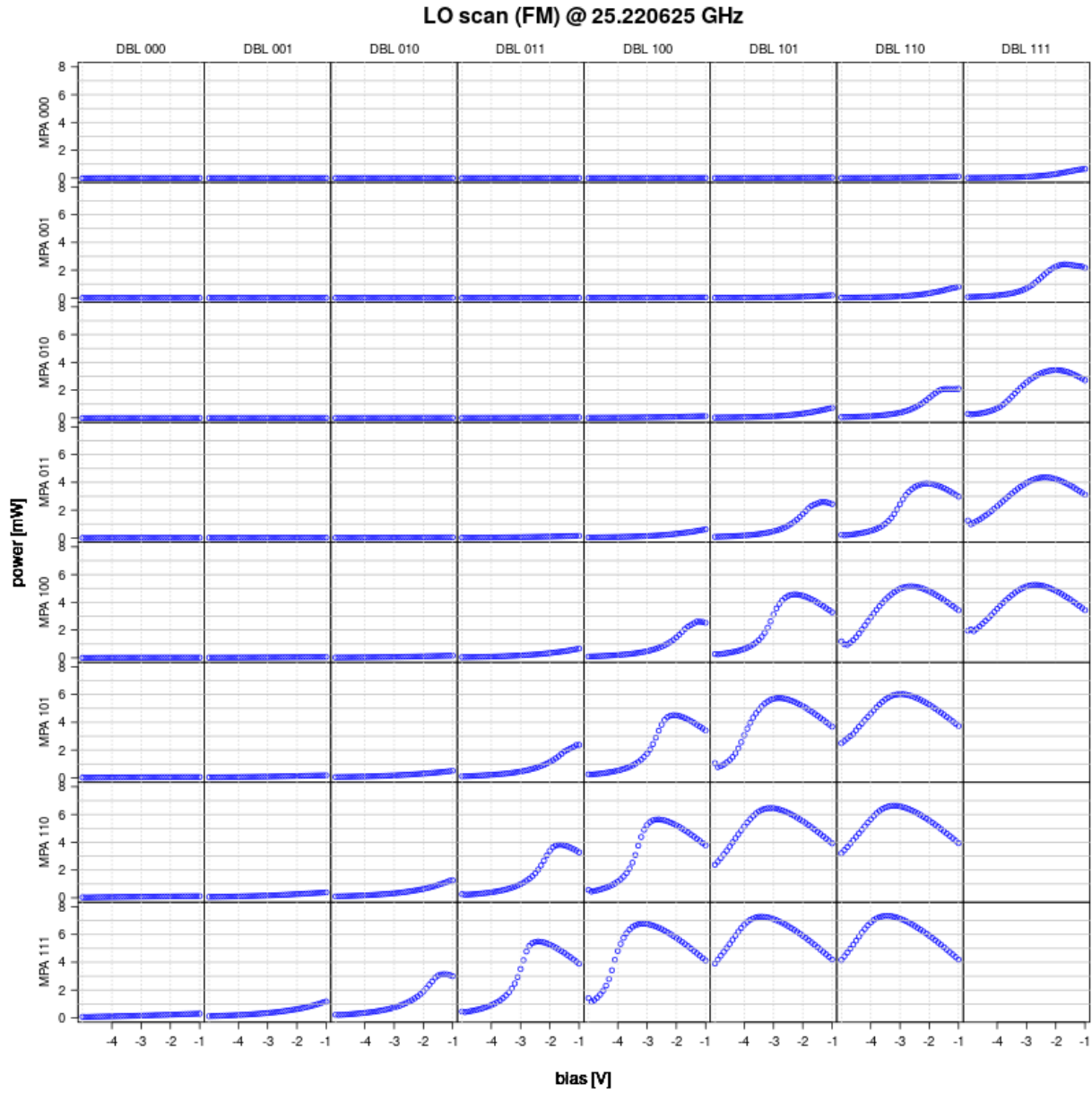


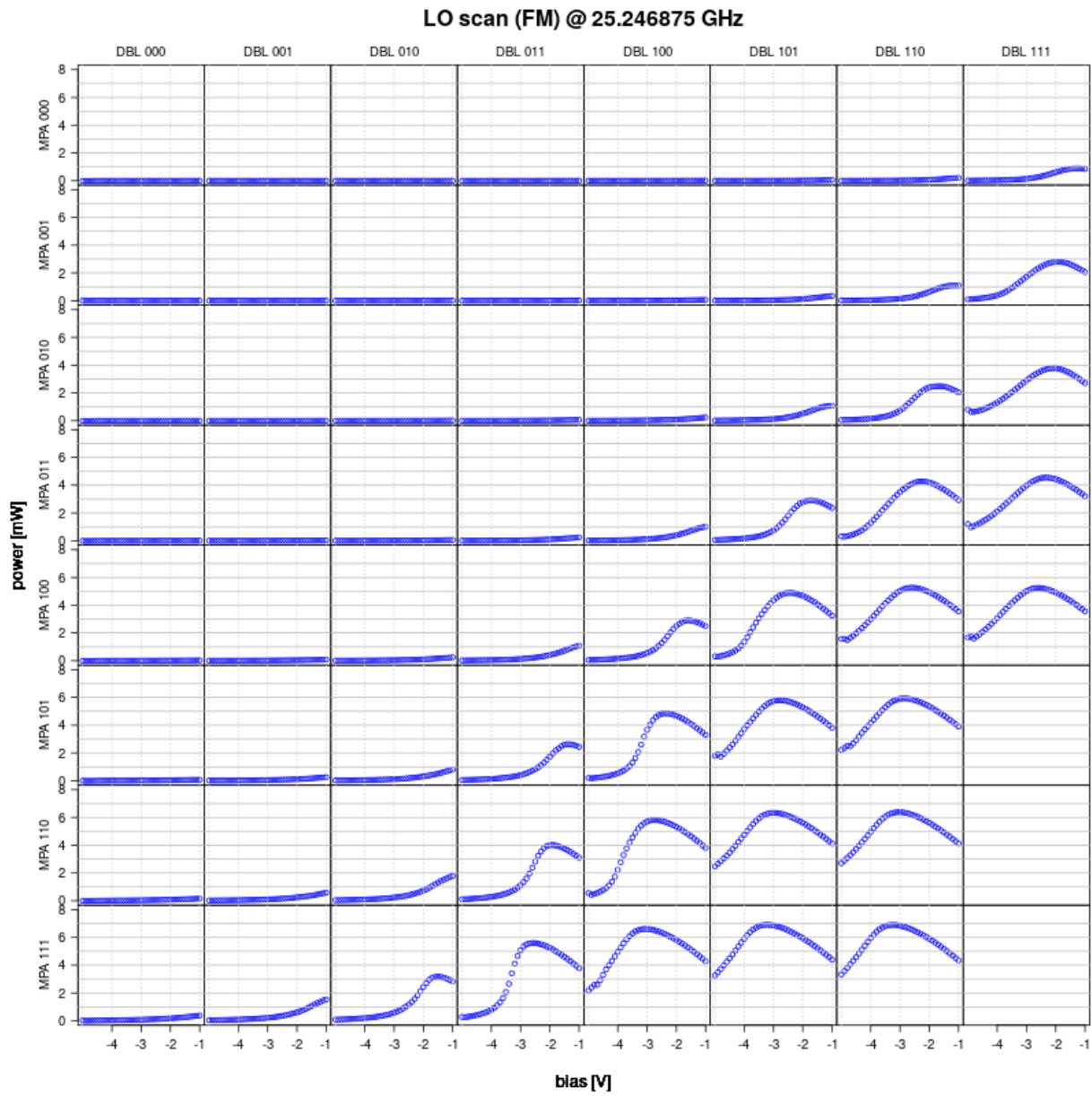


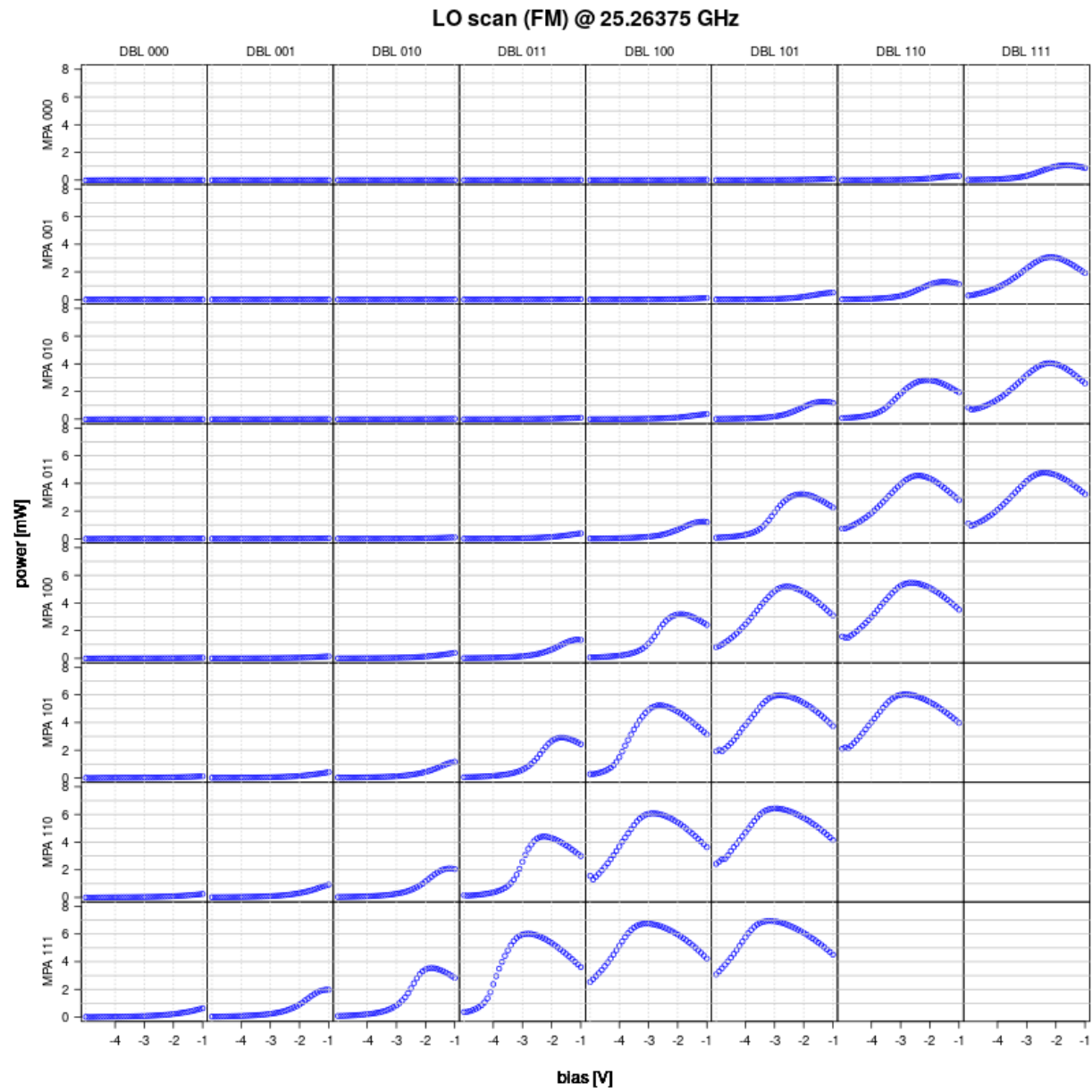


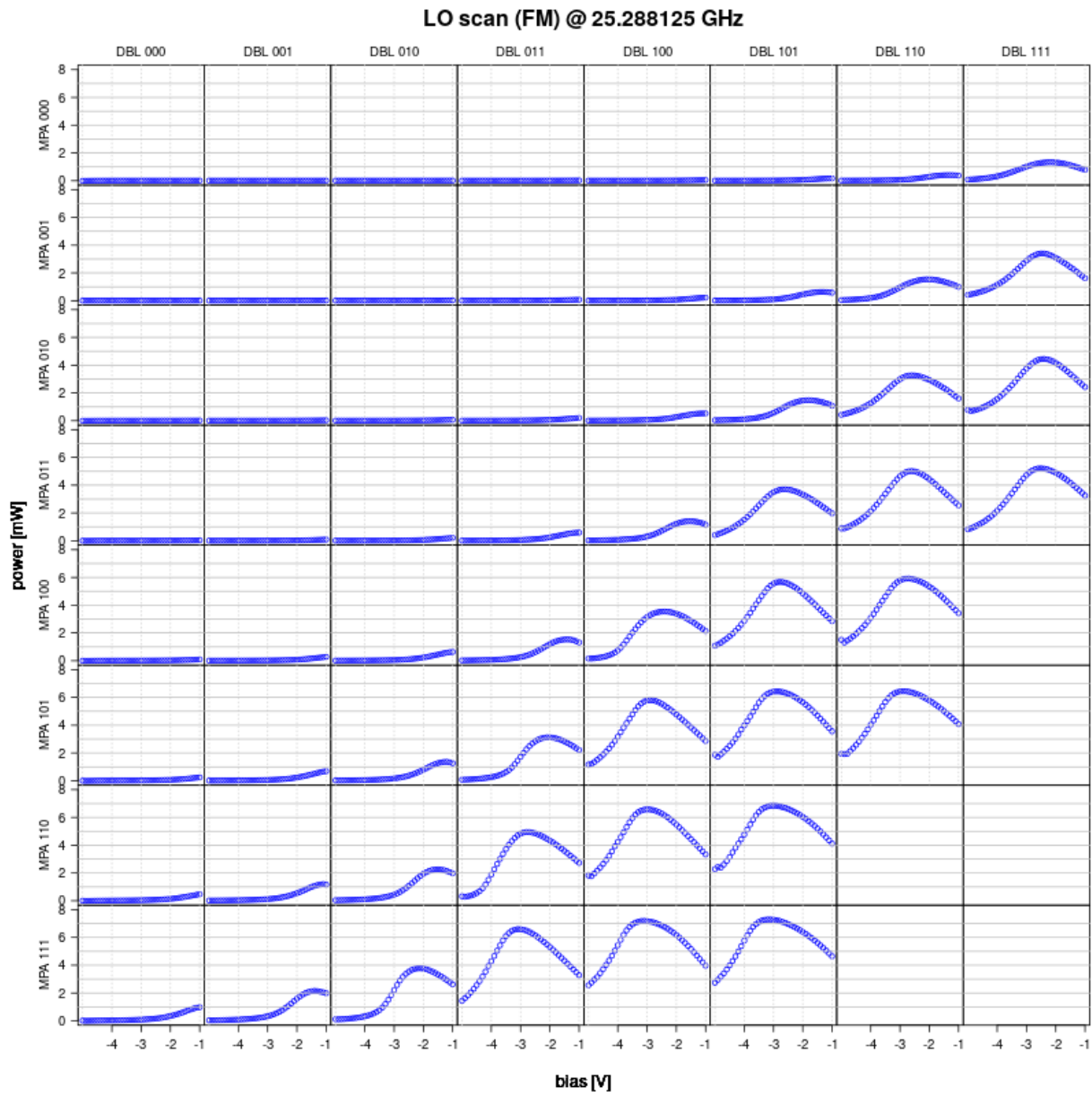
**LO scan (FM) @ 25.1475 GHz**

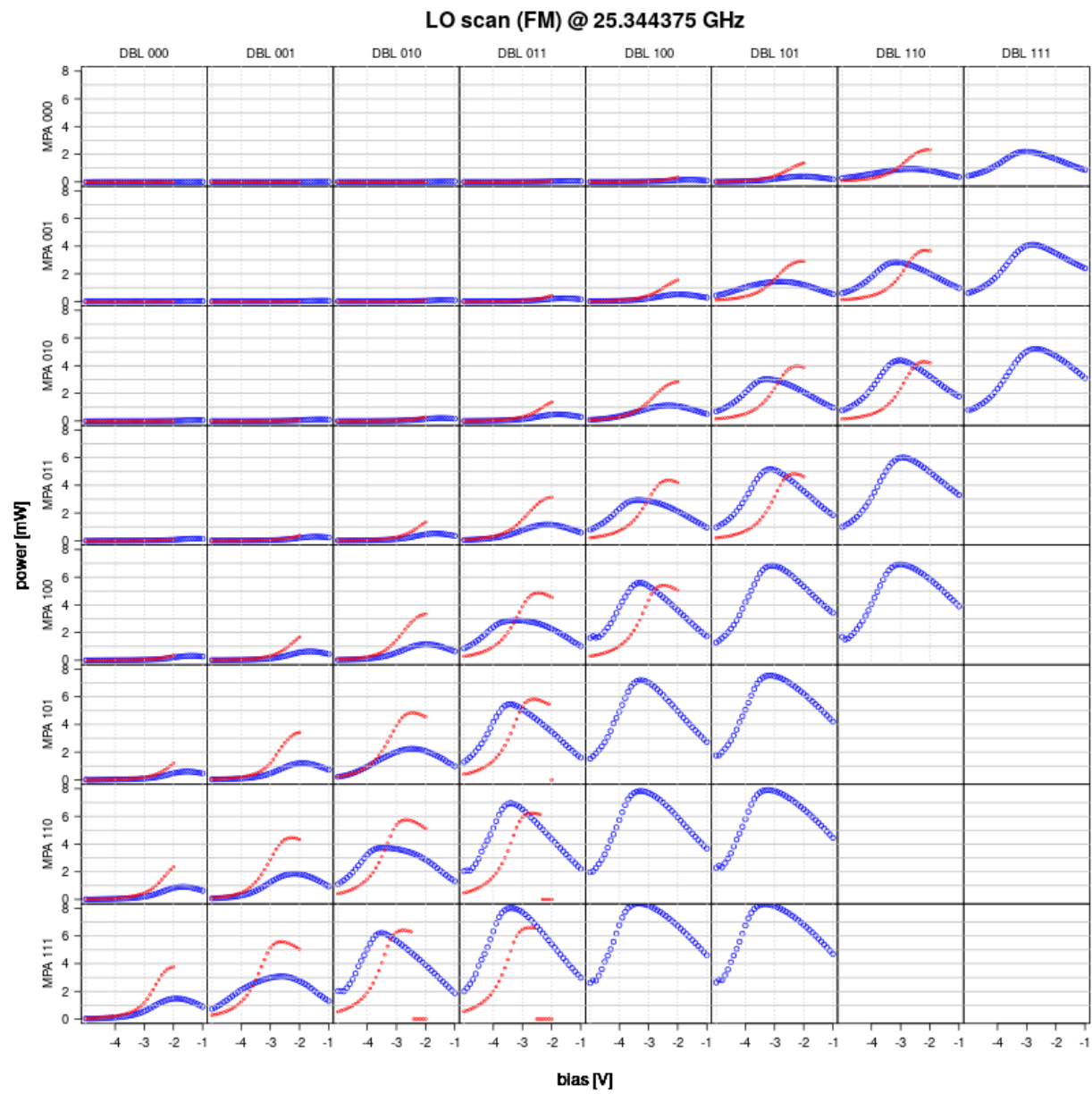




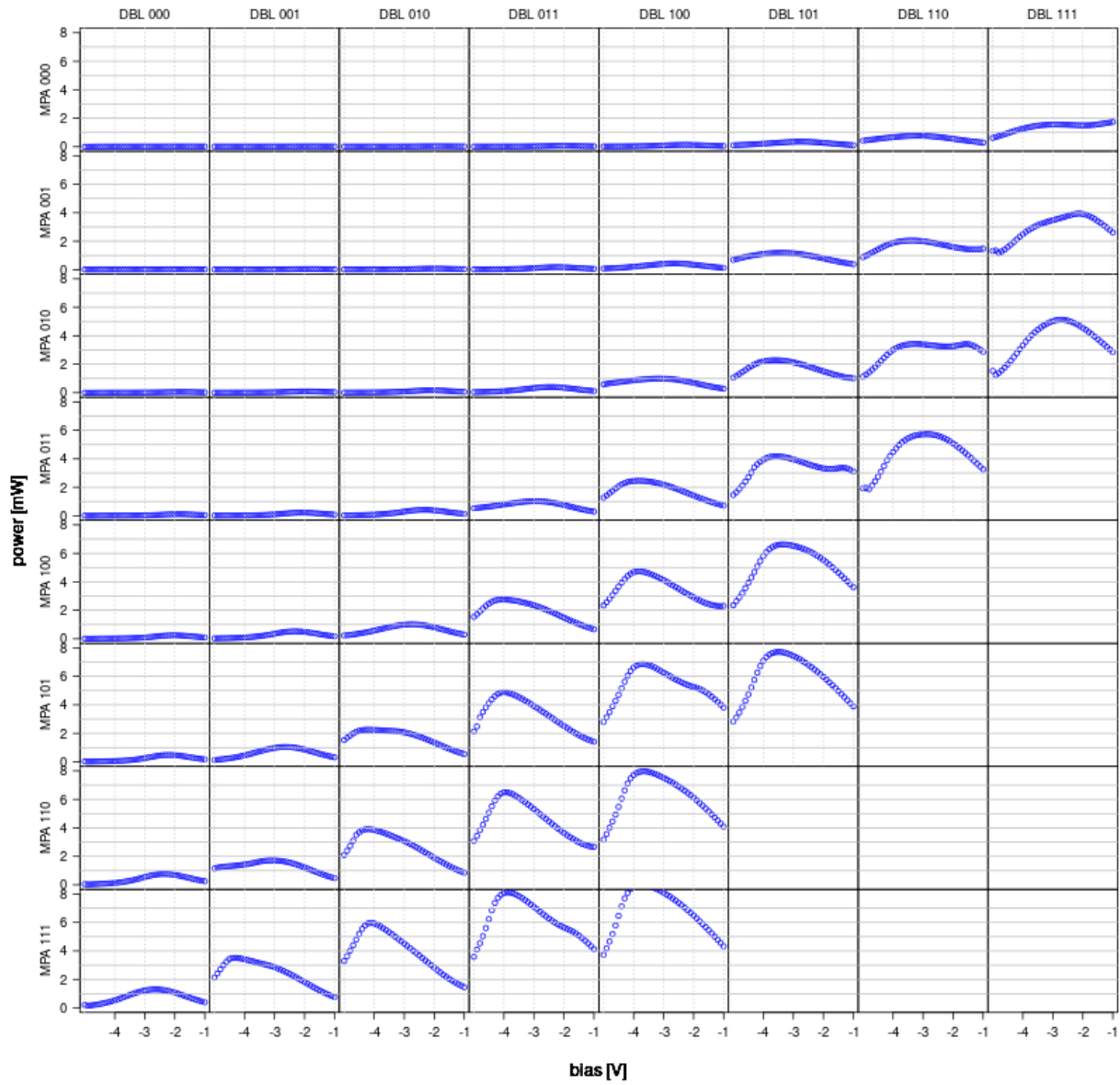


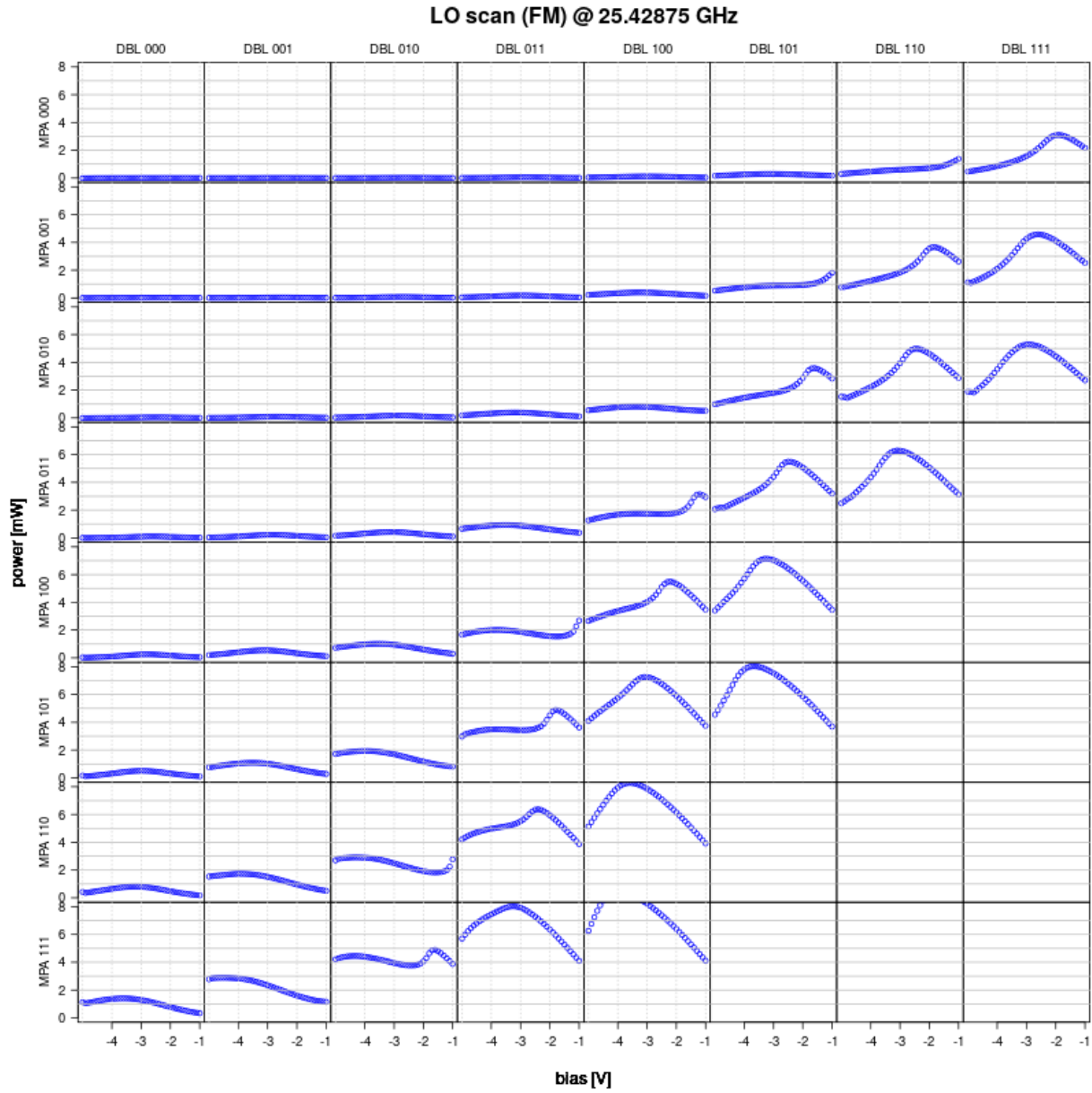




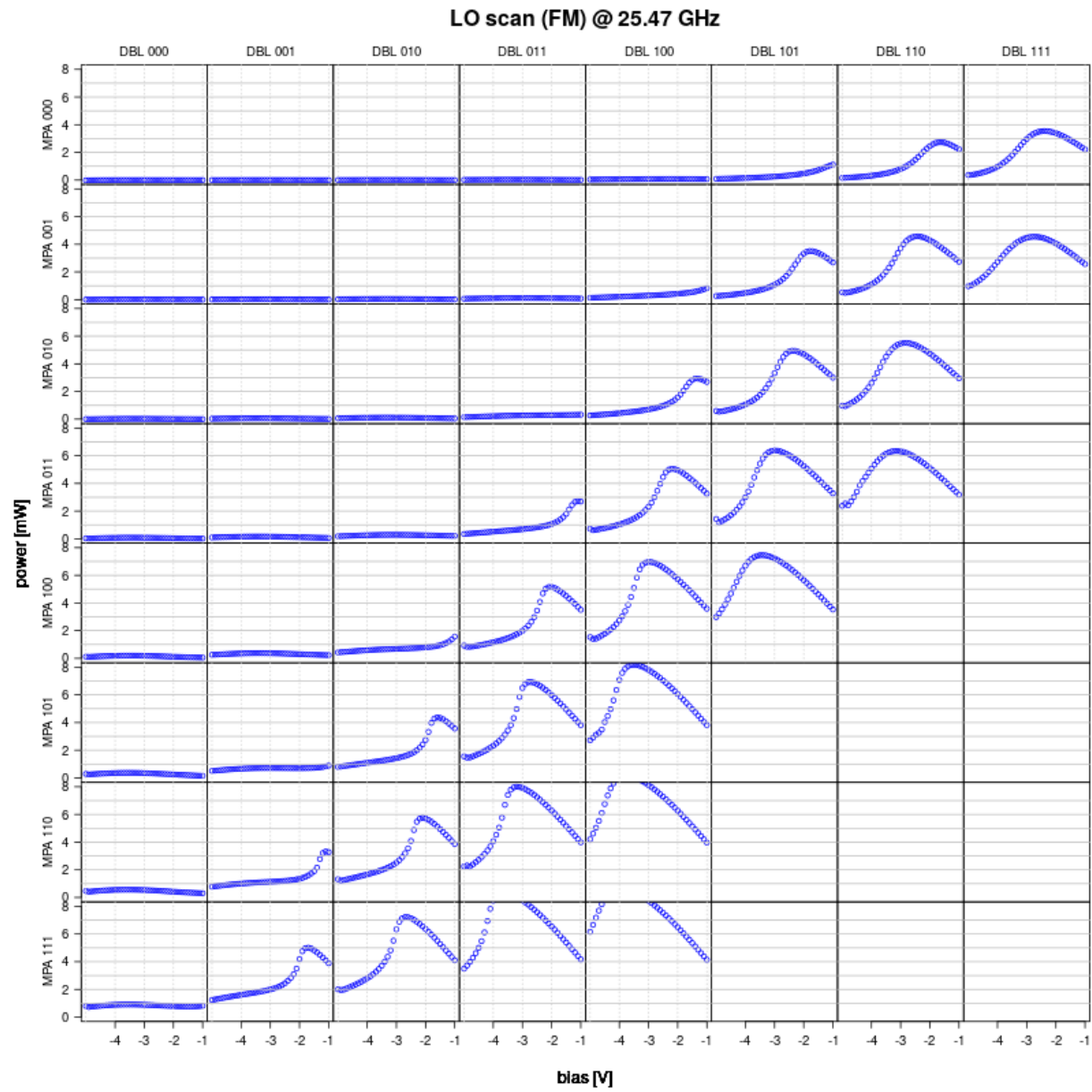


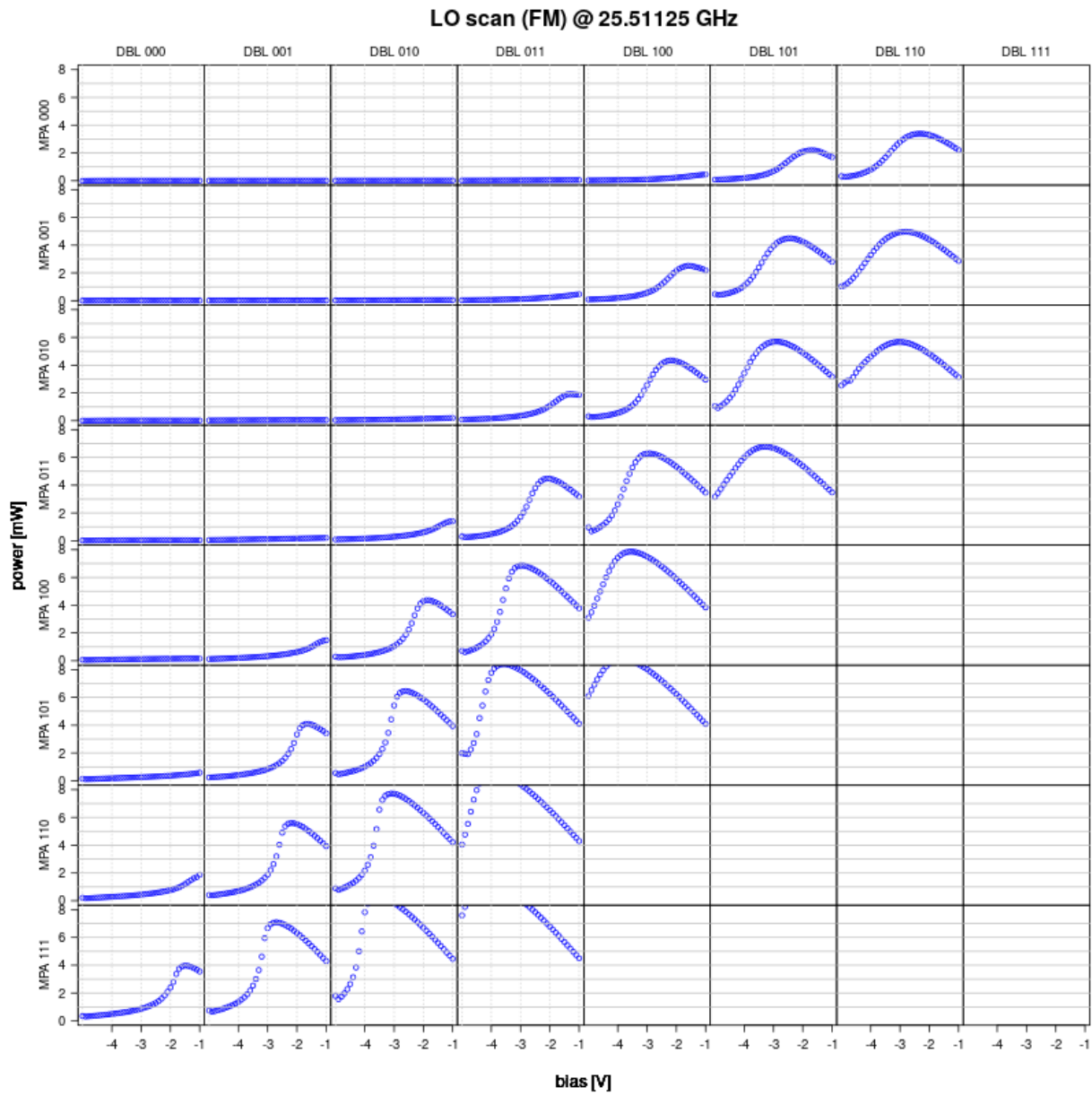
**LO scan (FM) @ 25.3875 GHz**

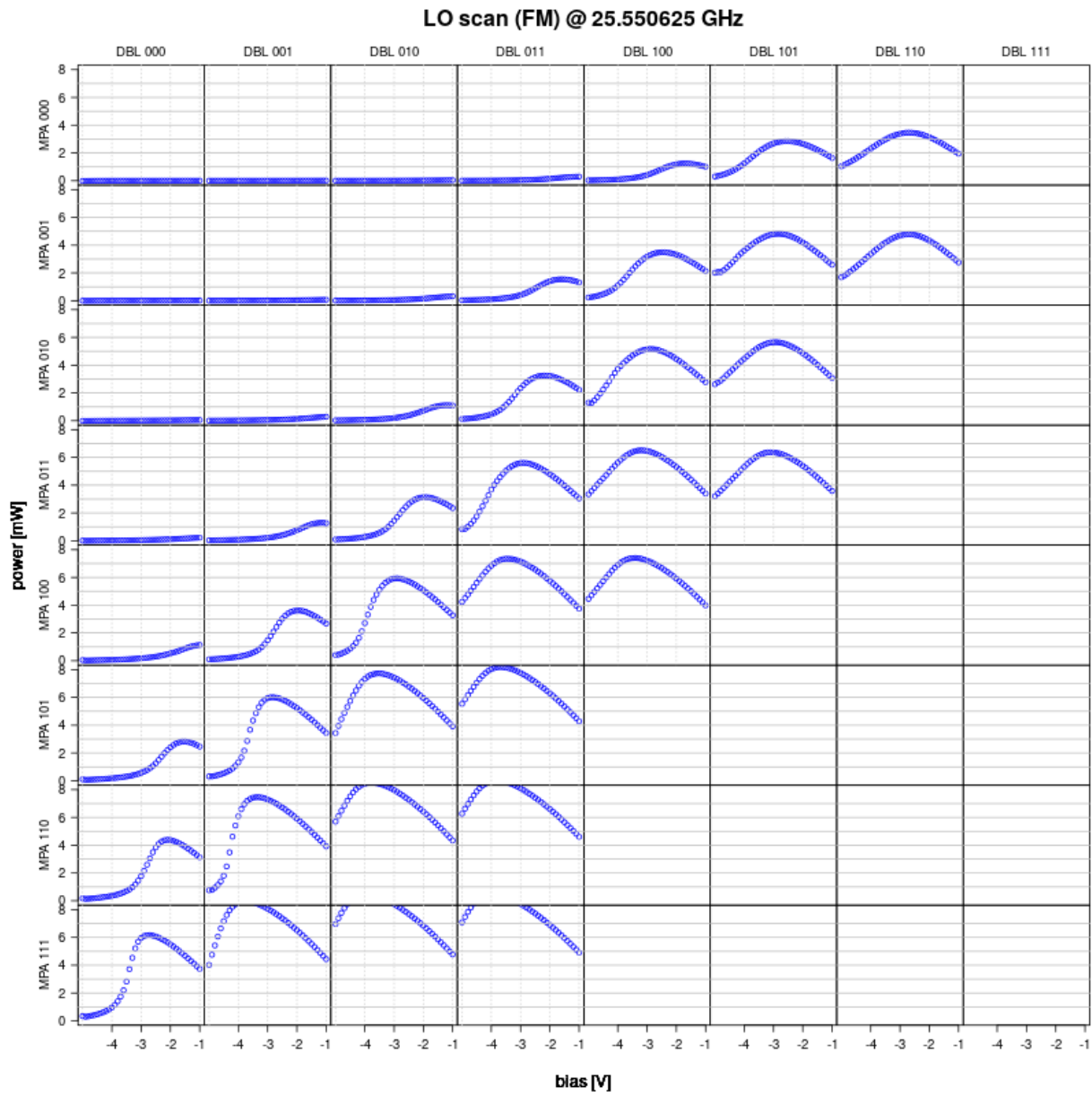


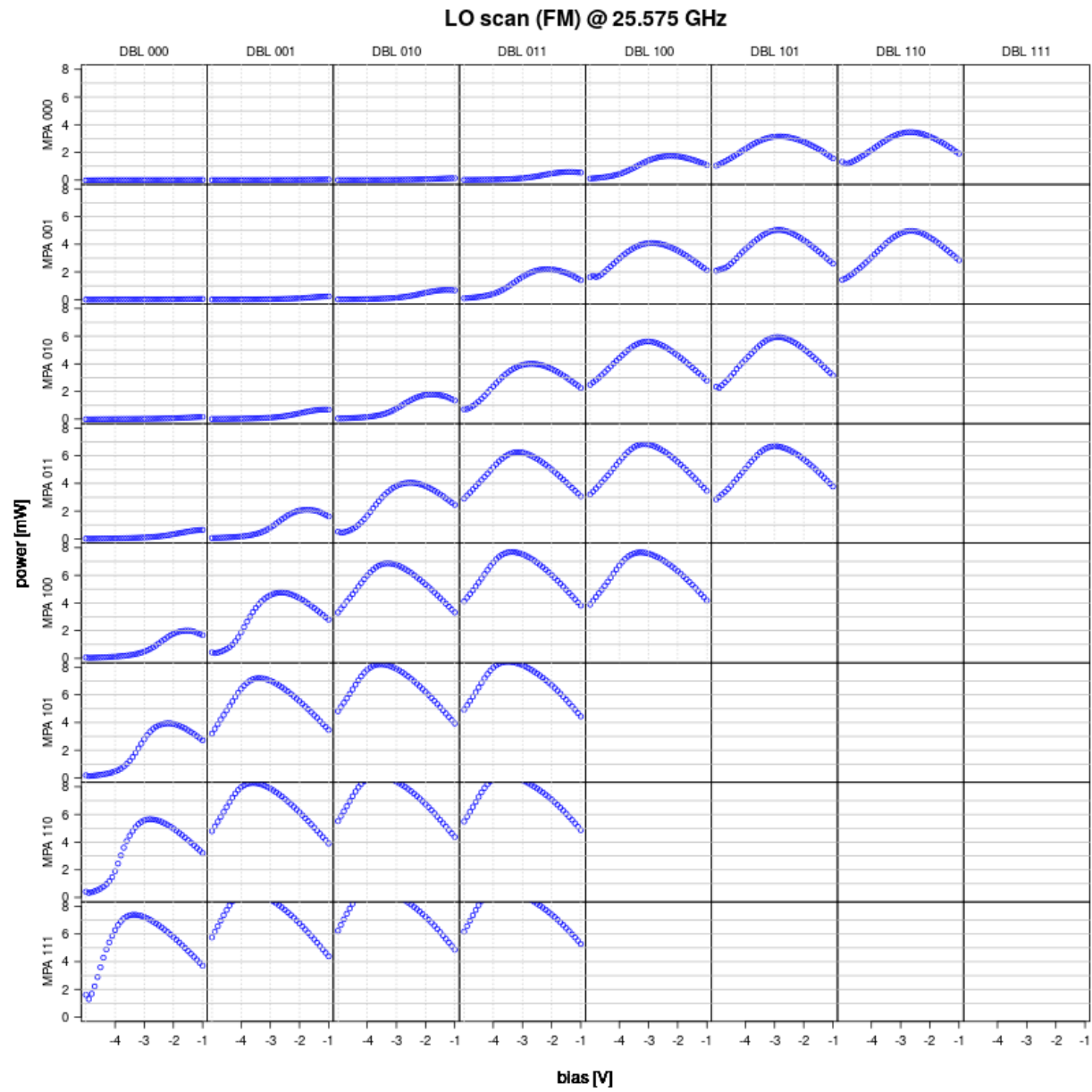


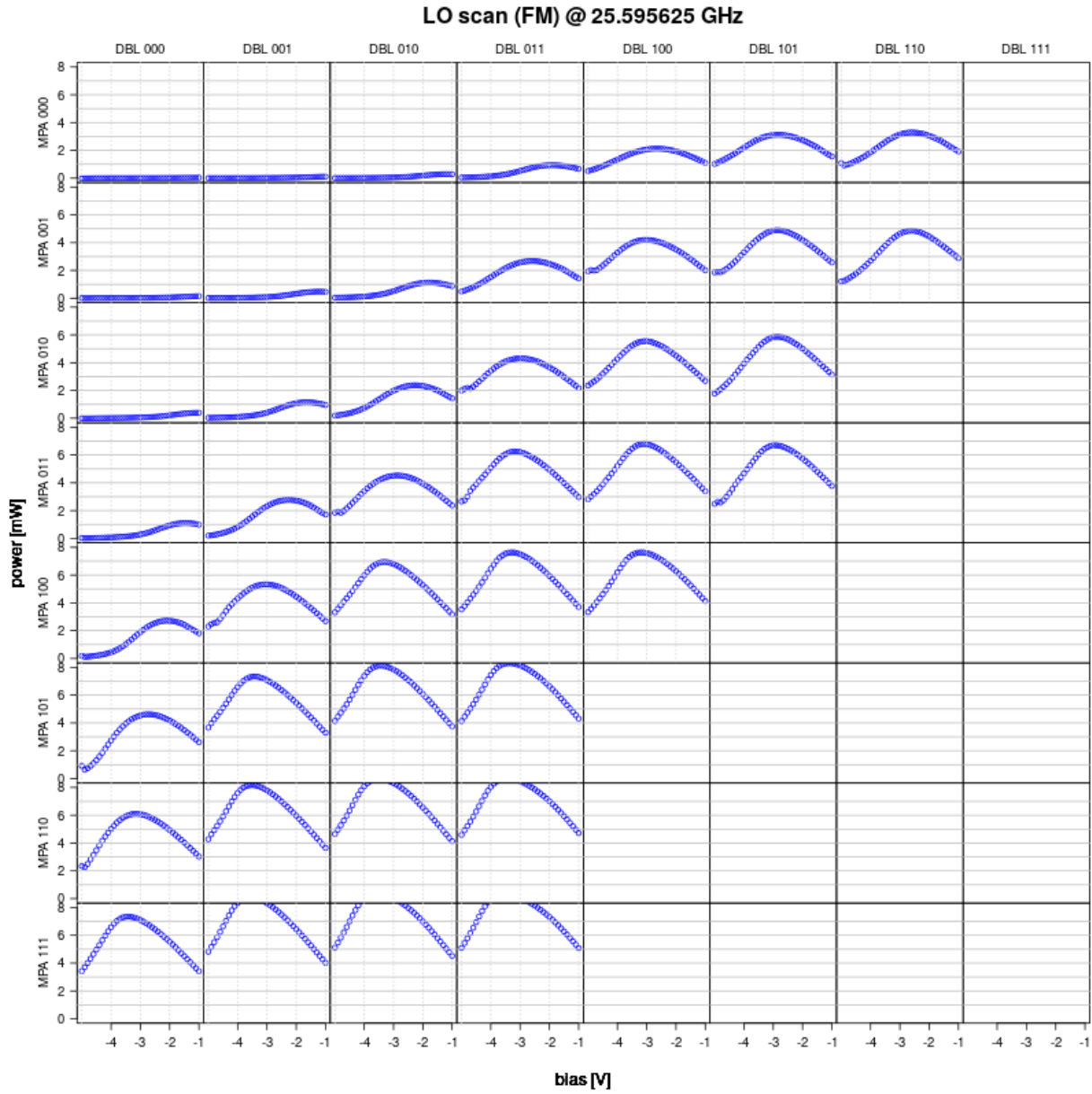


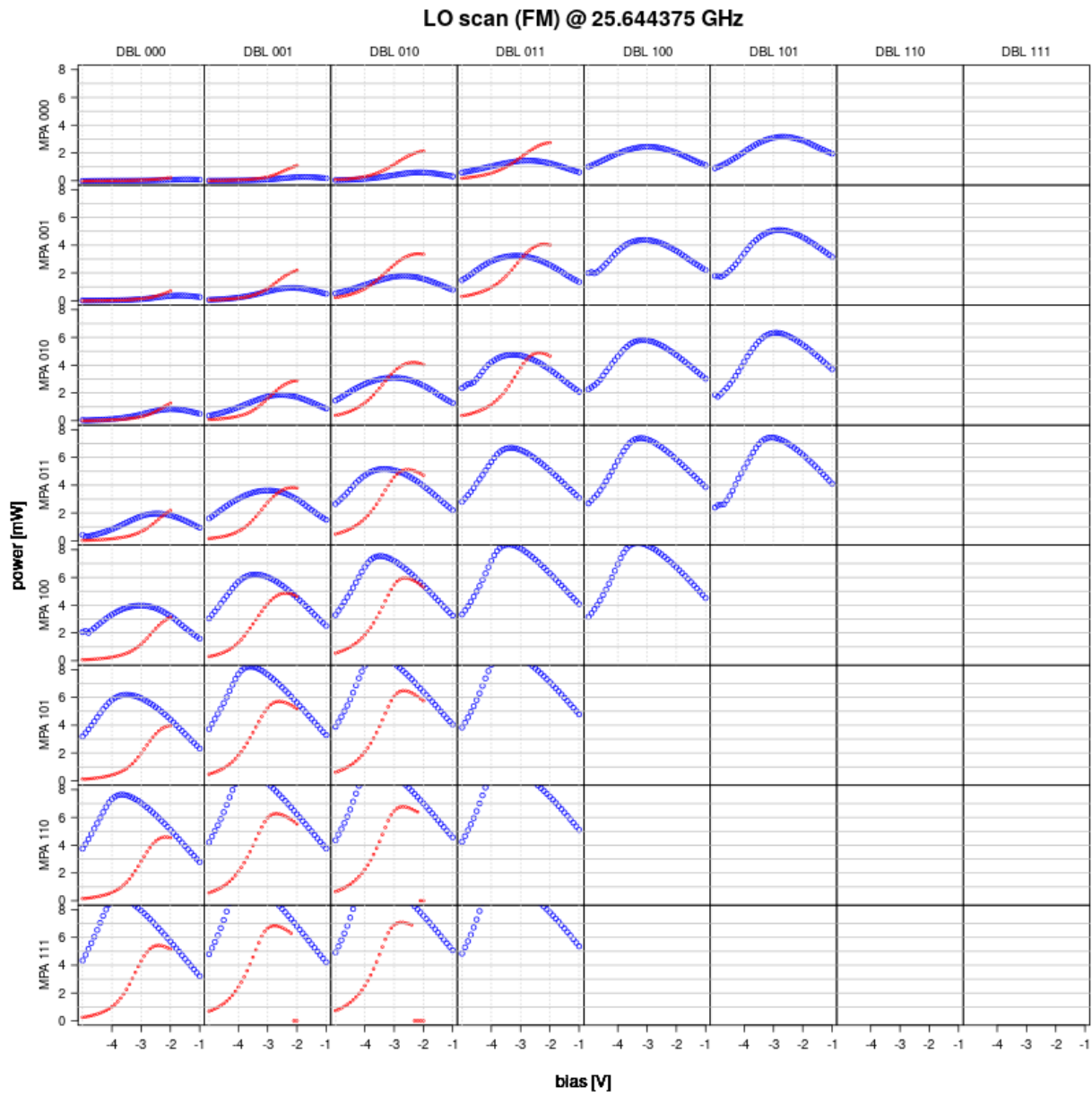


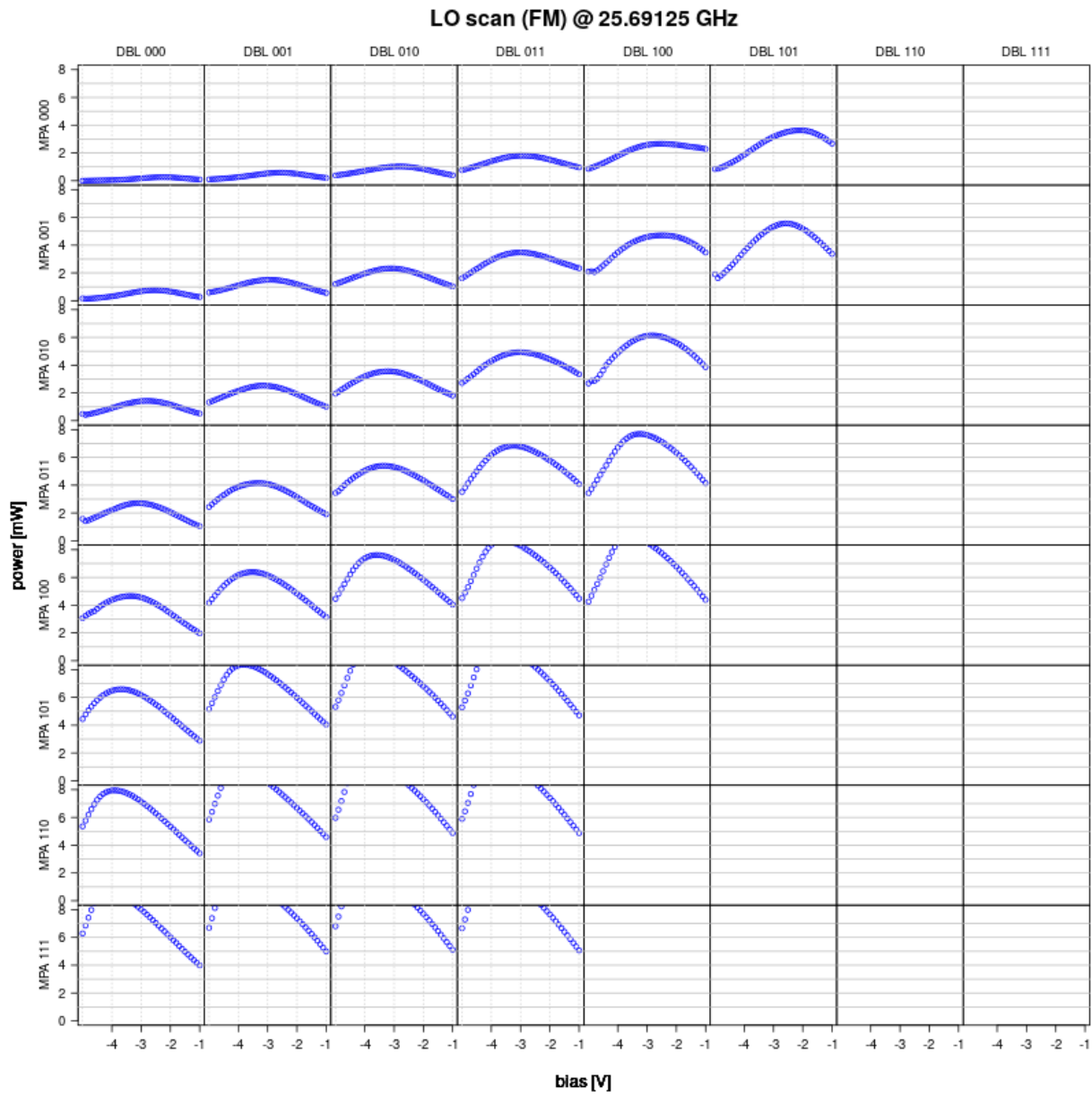


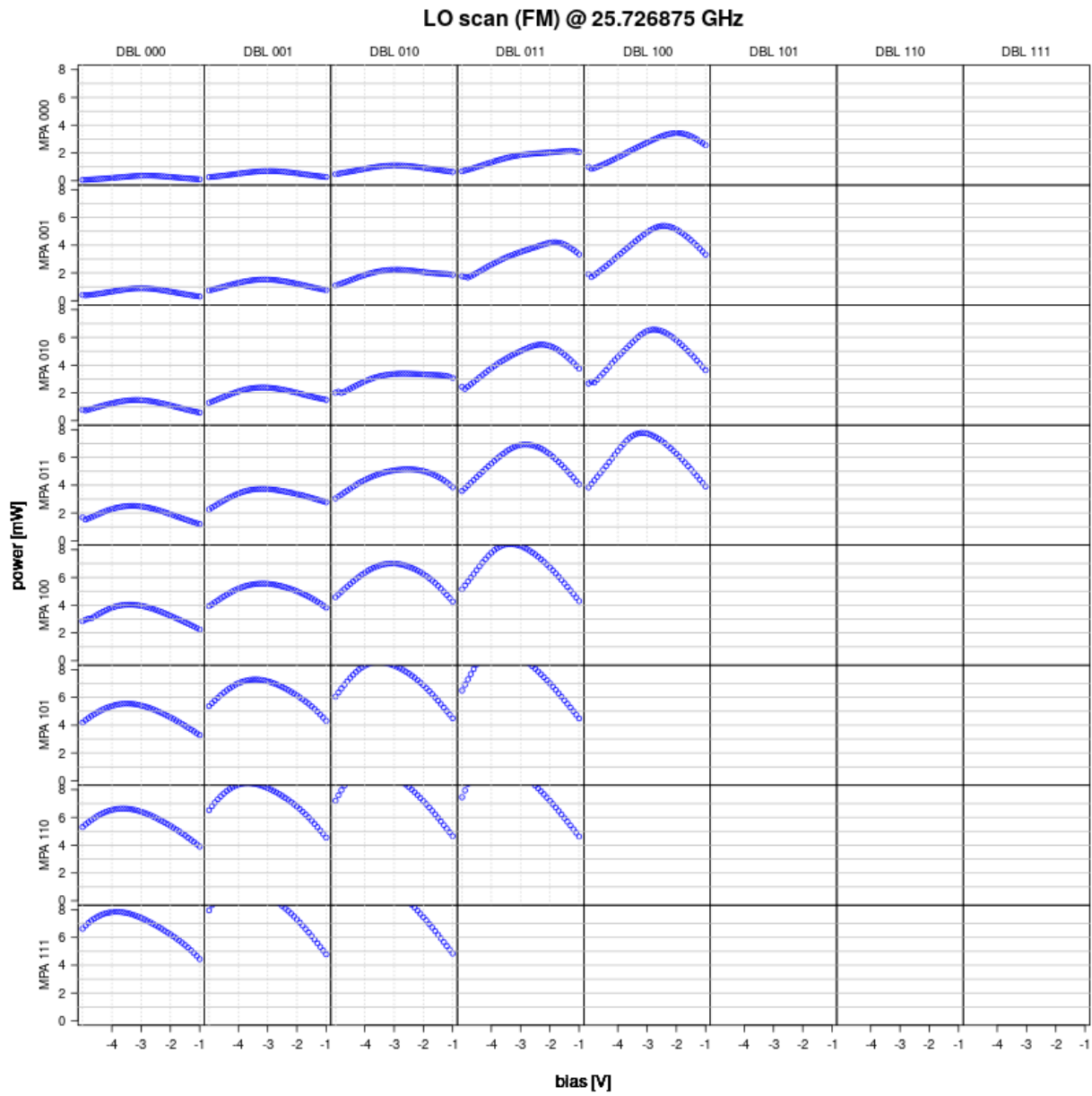




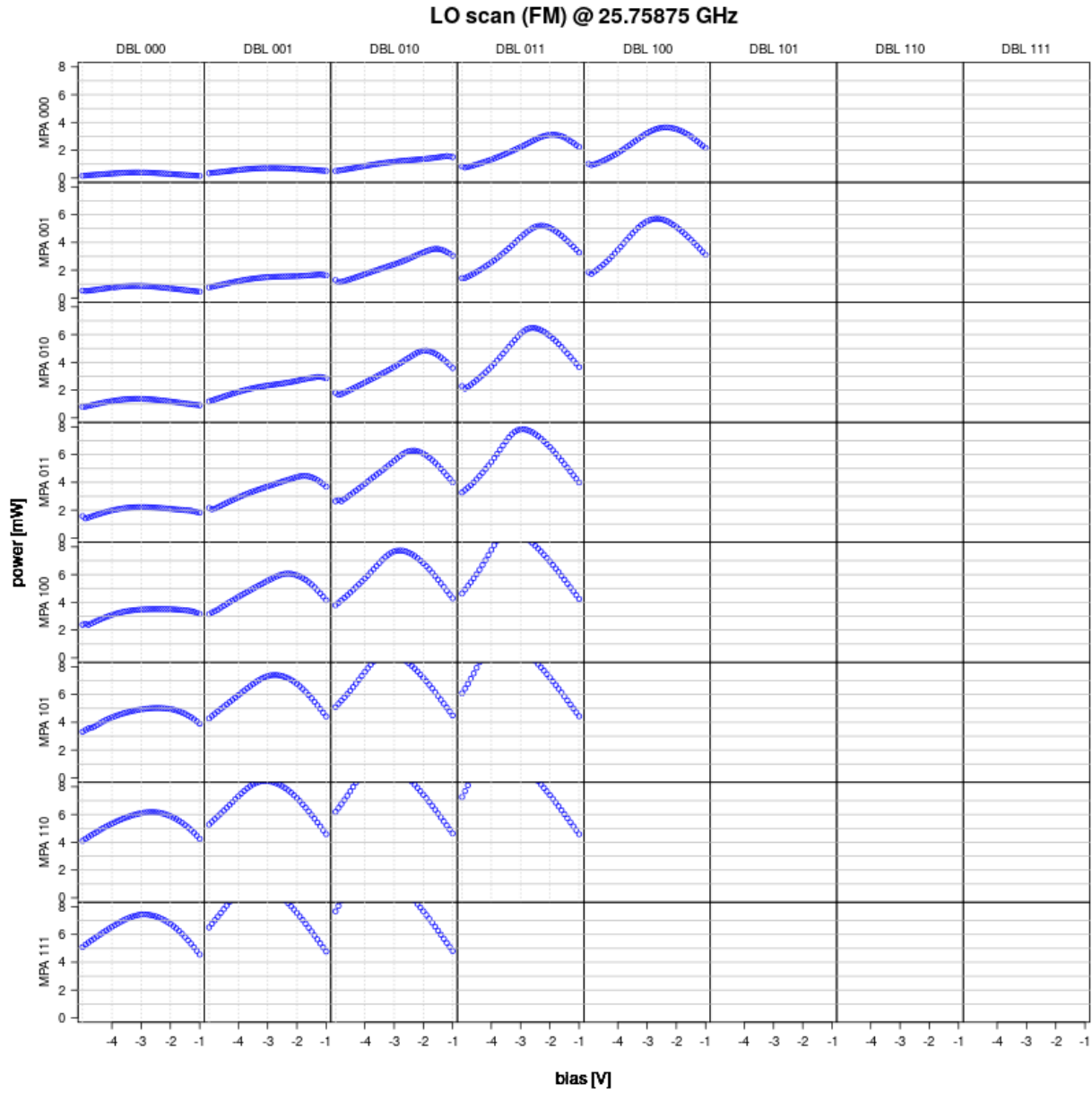


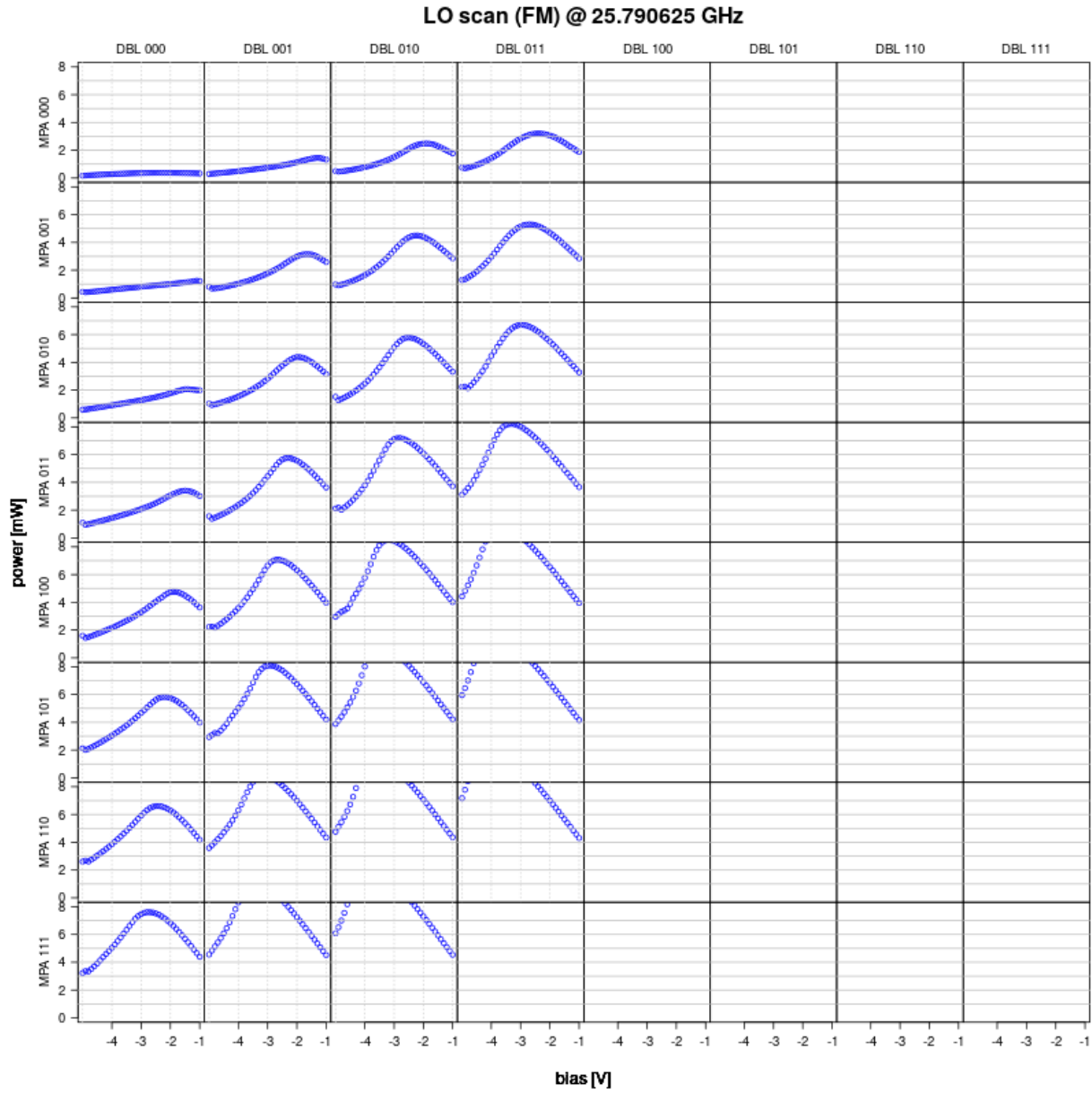


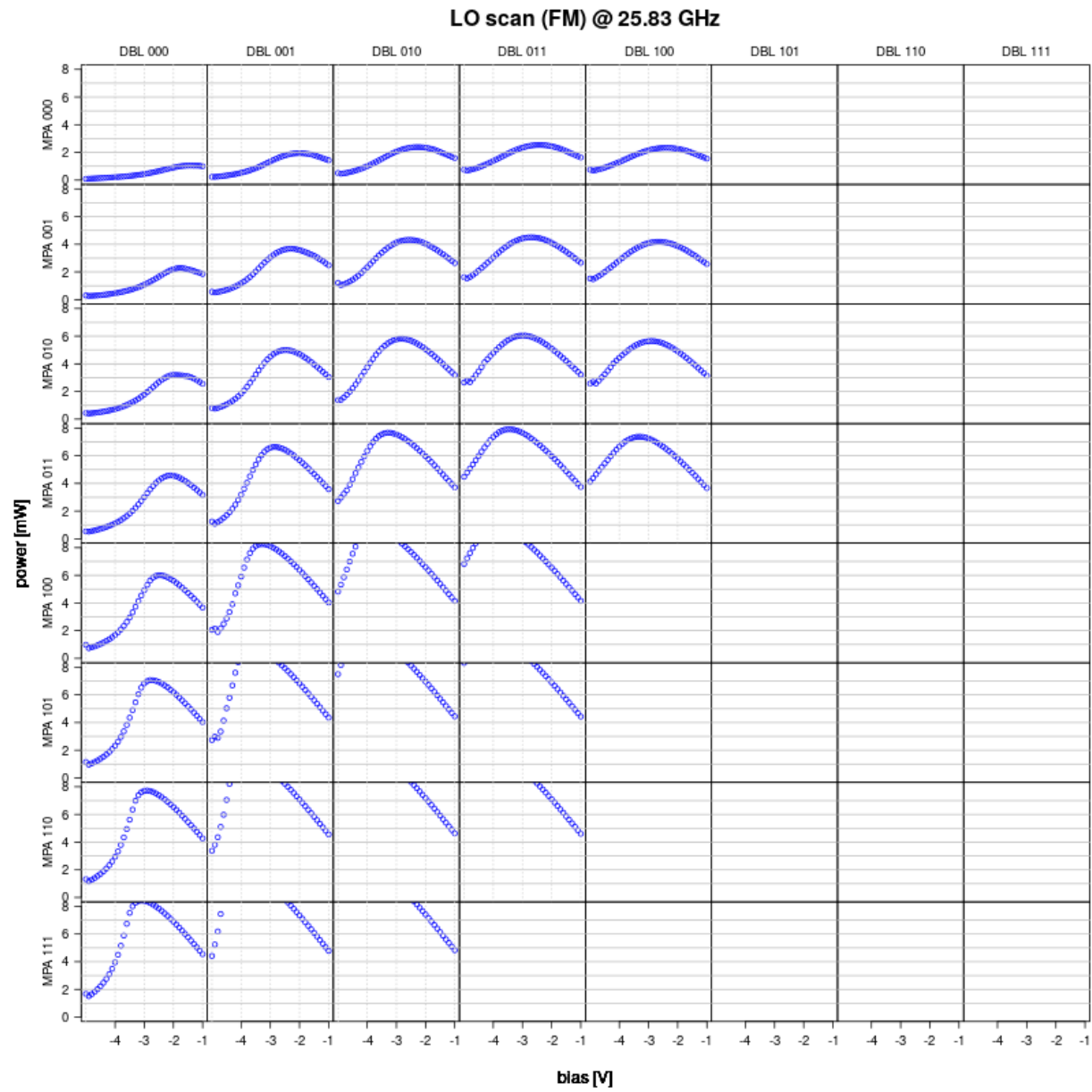


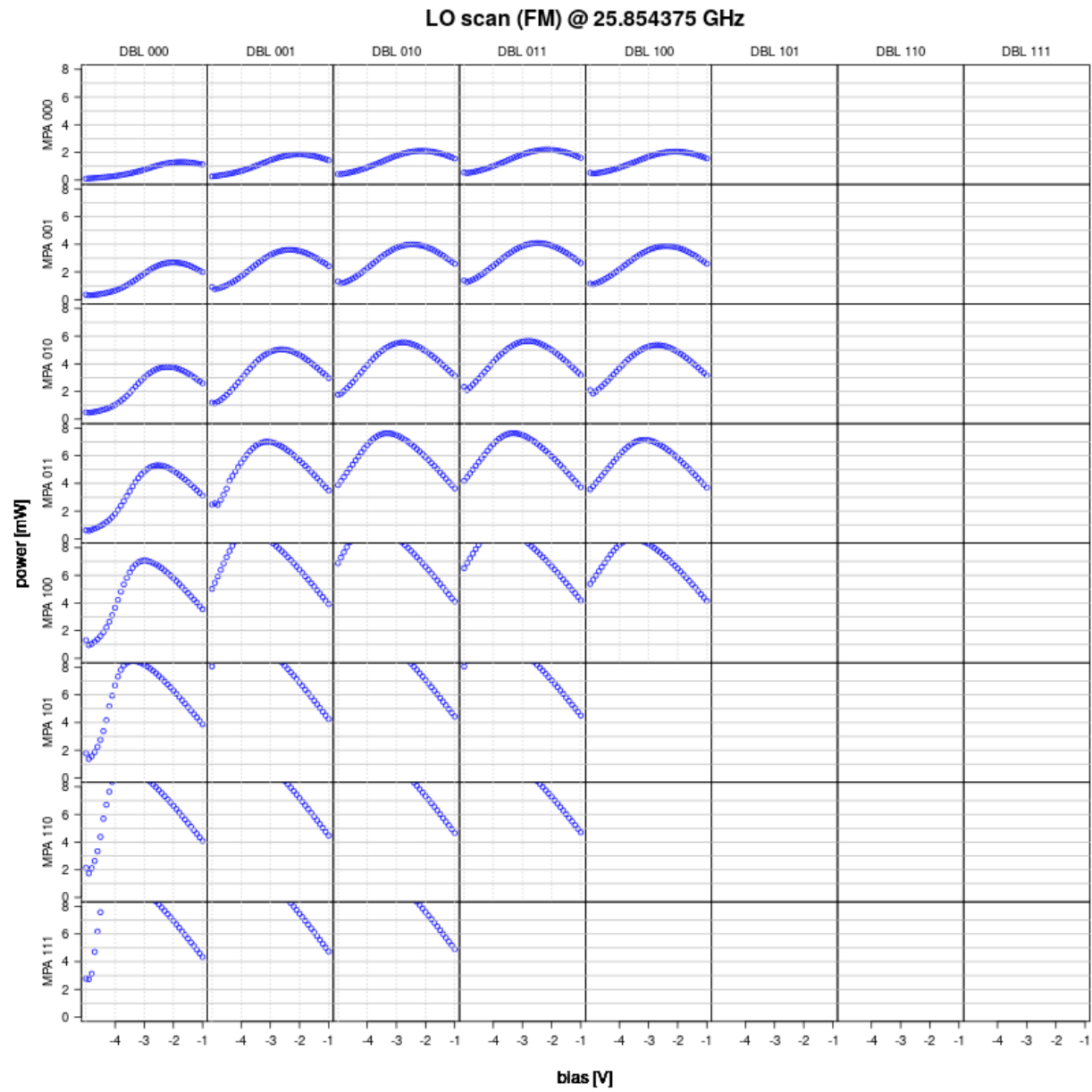


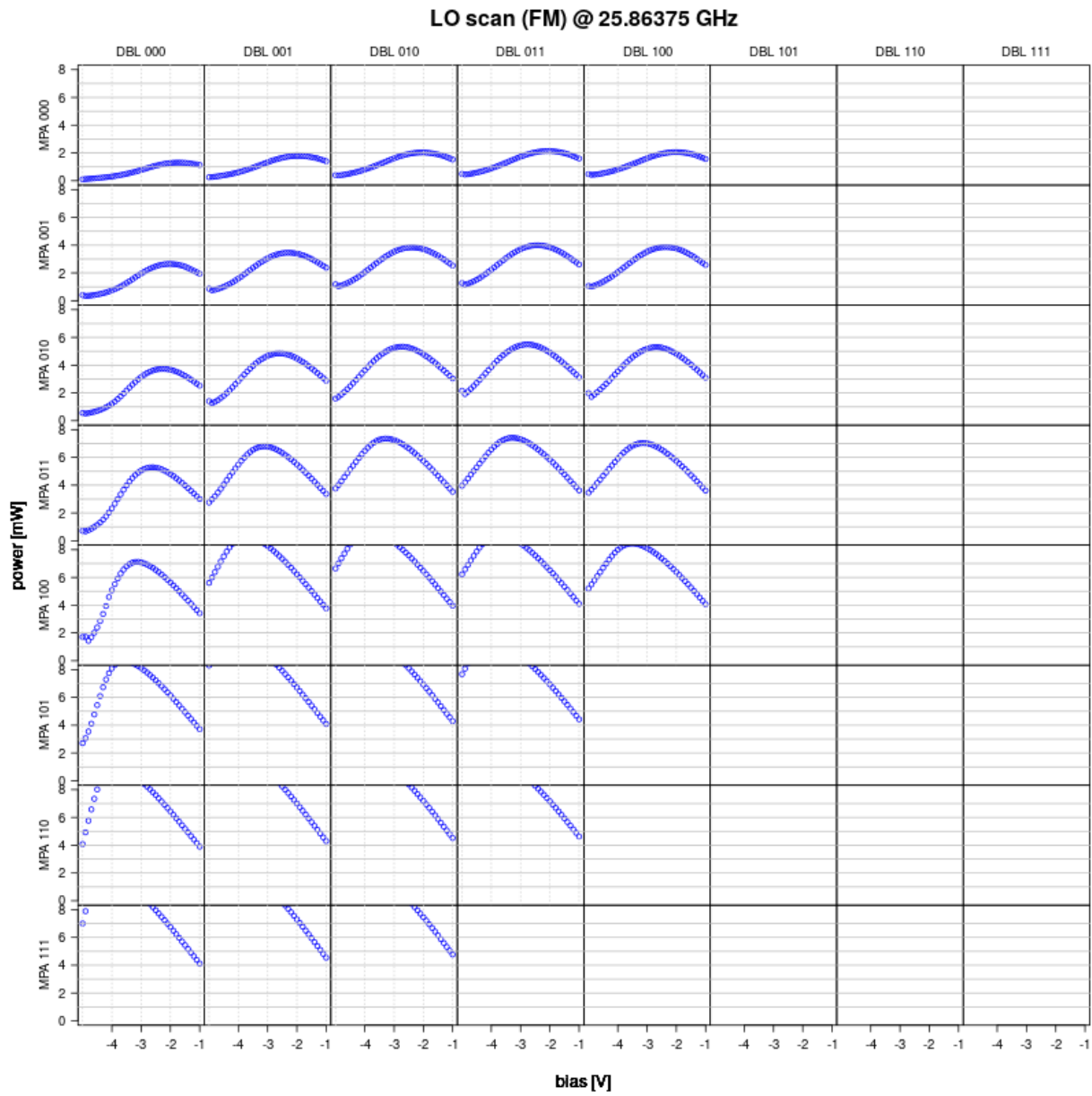


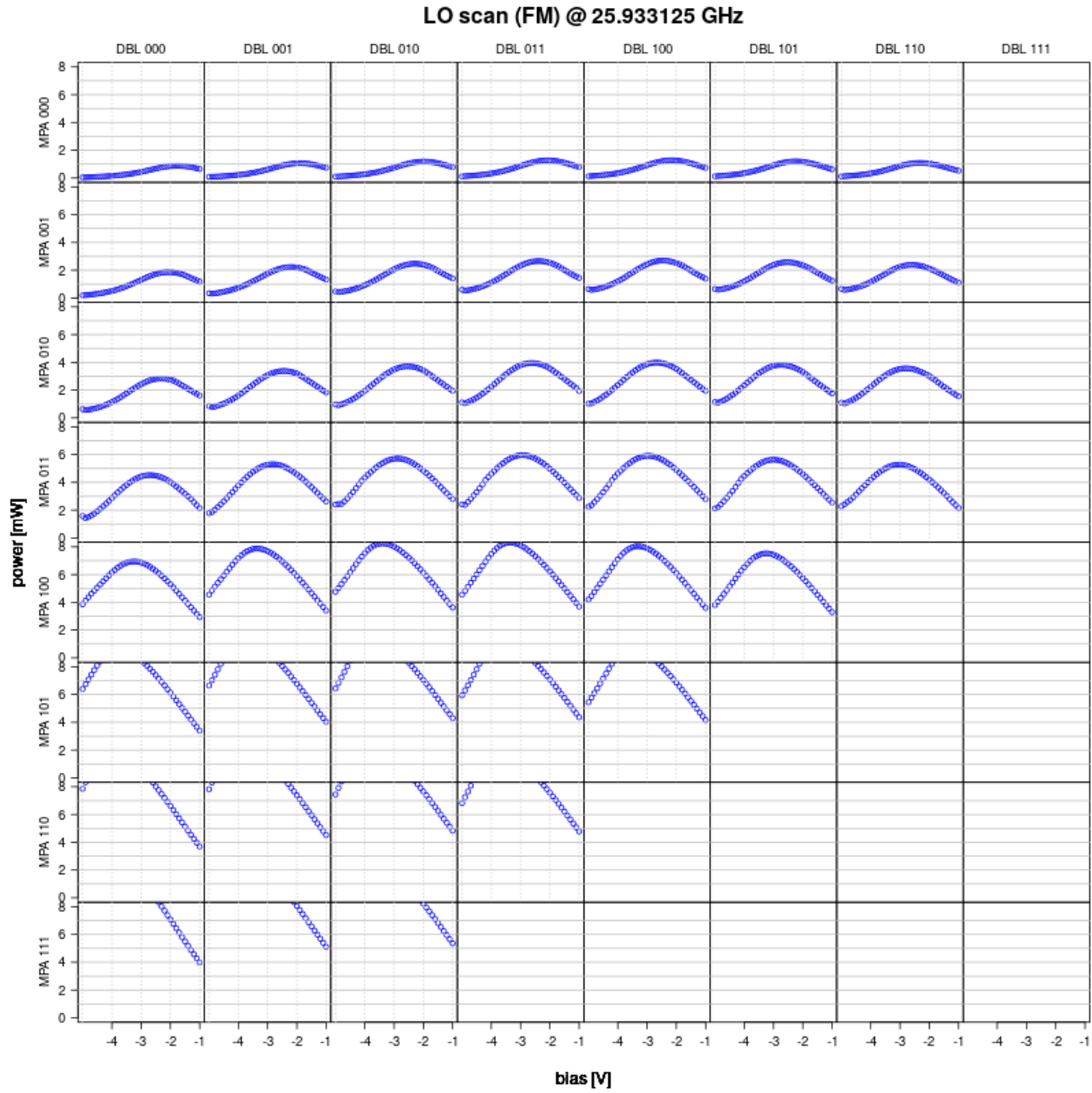


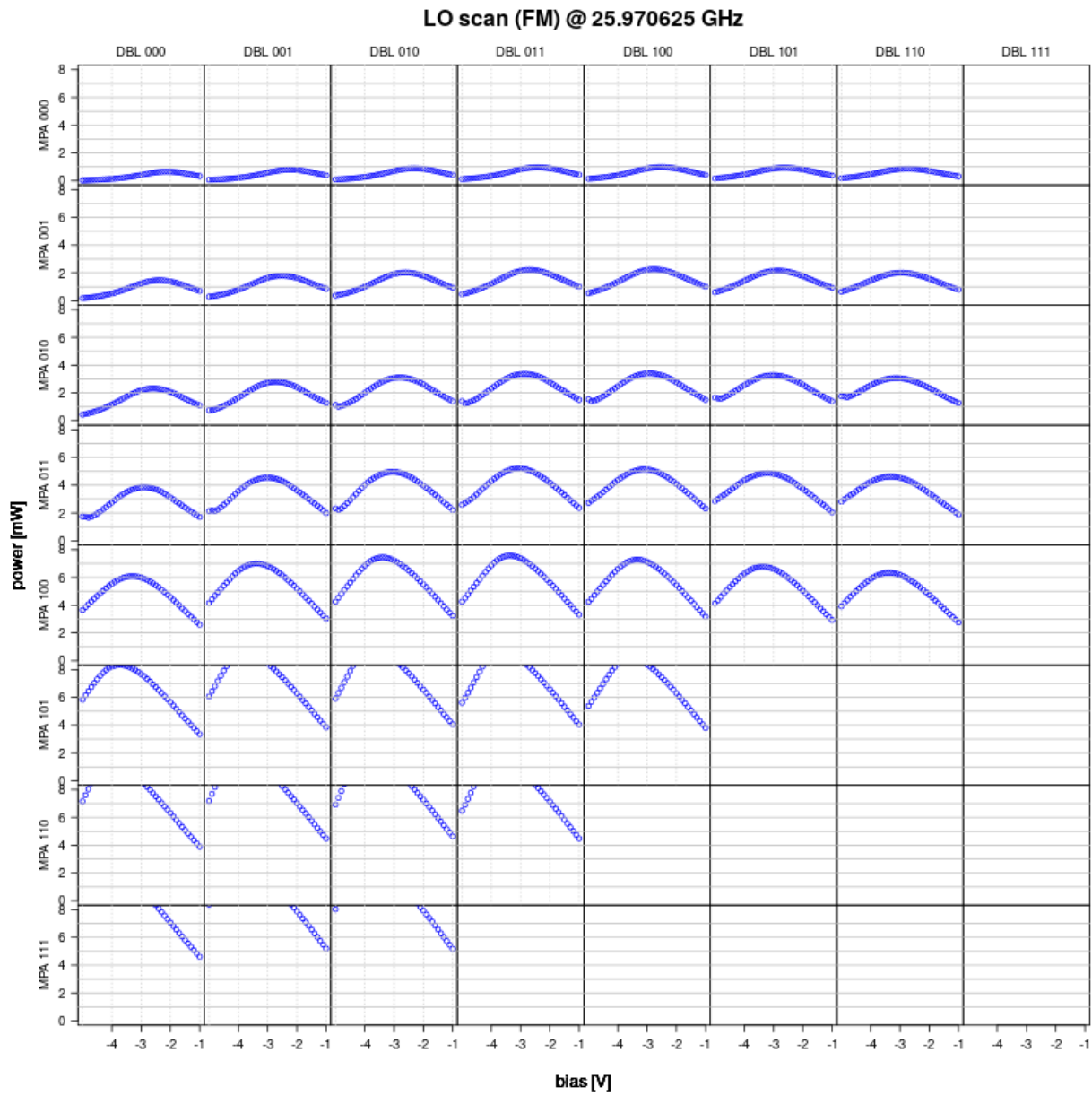


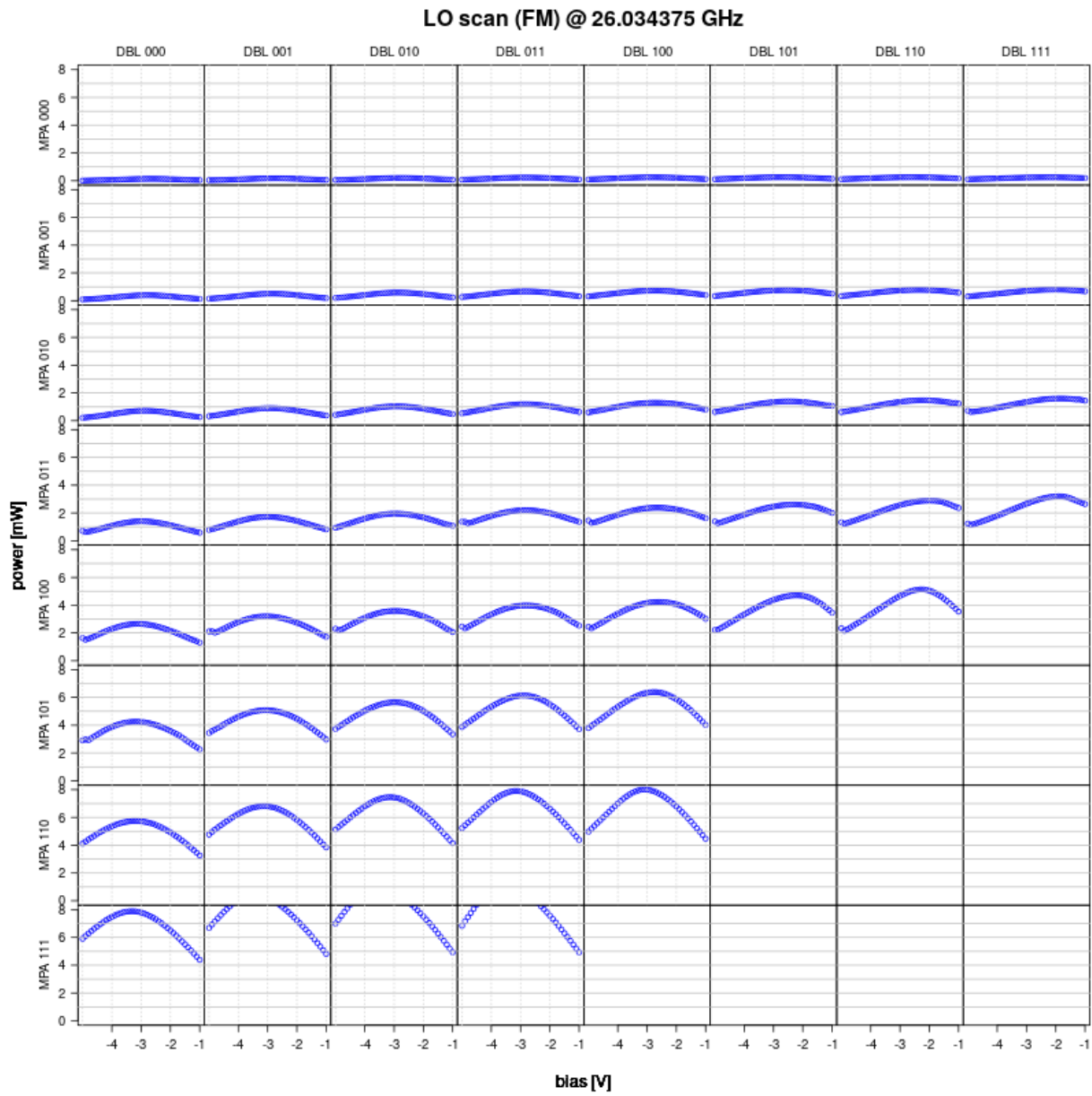




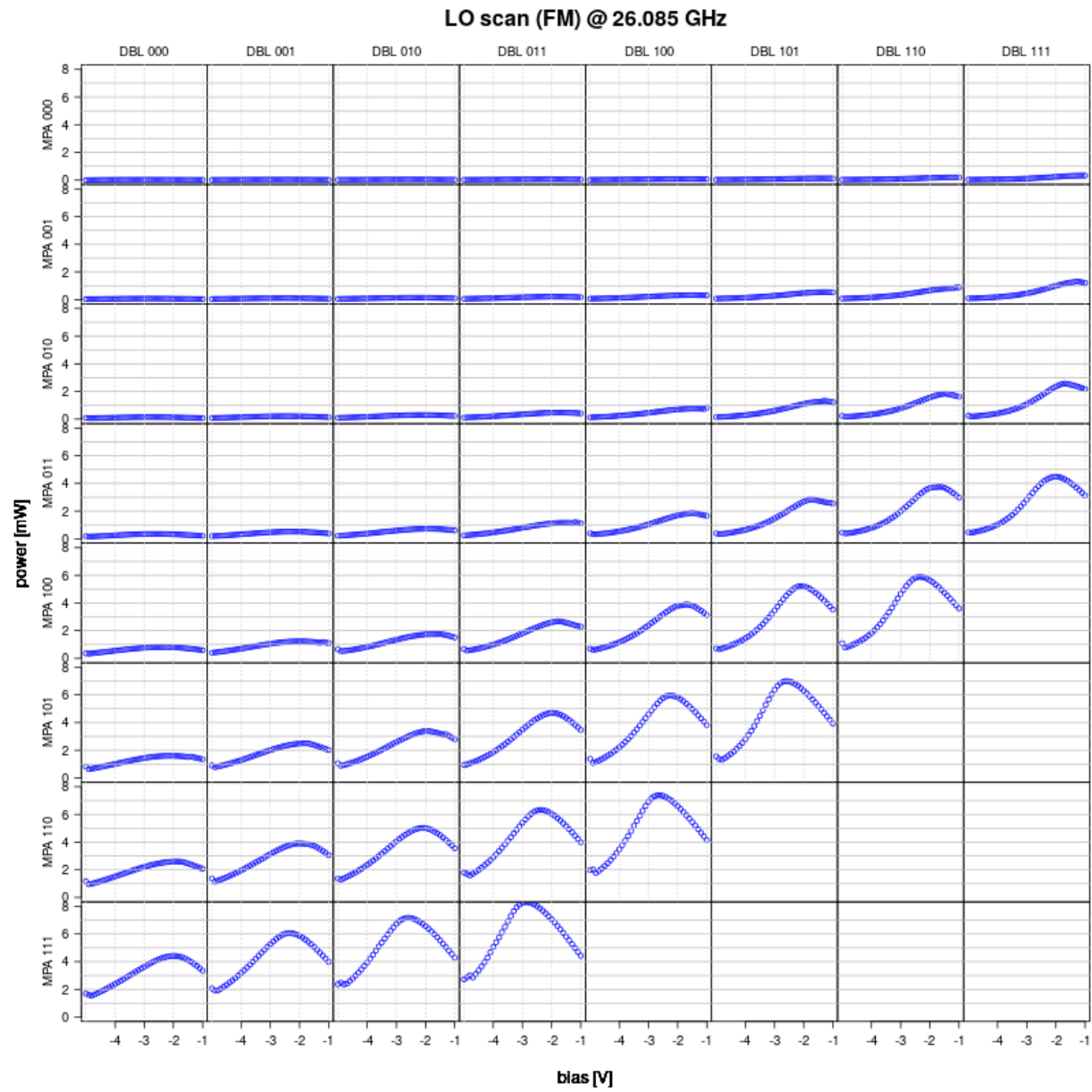




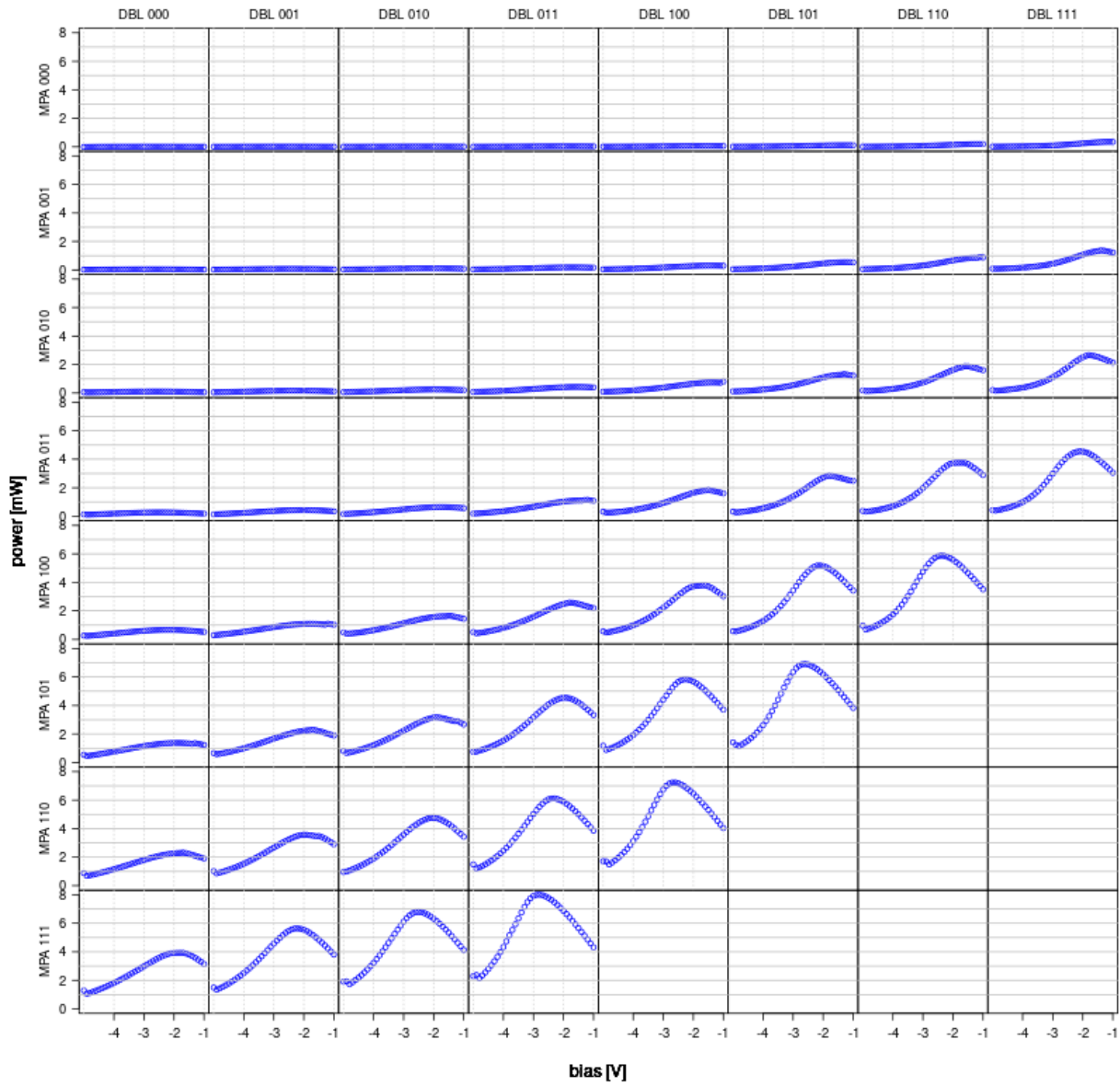


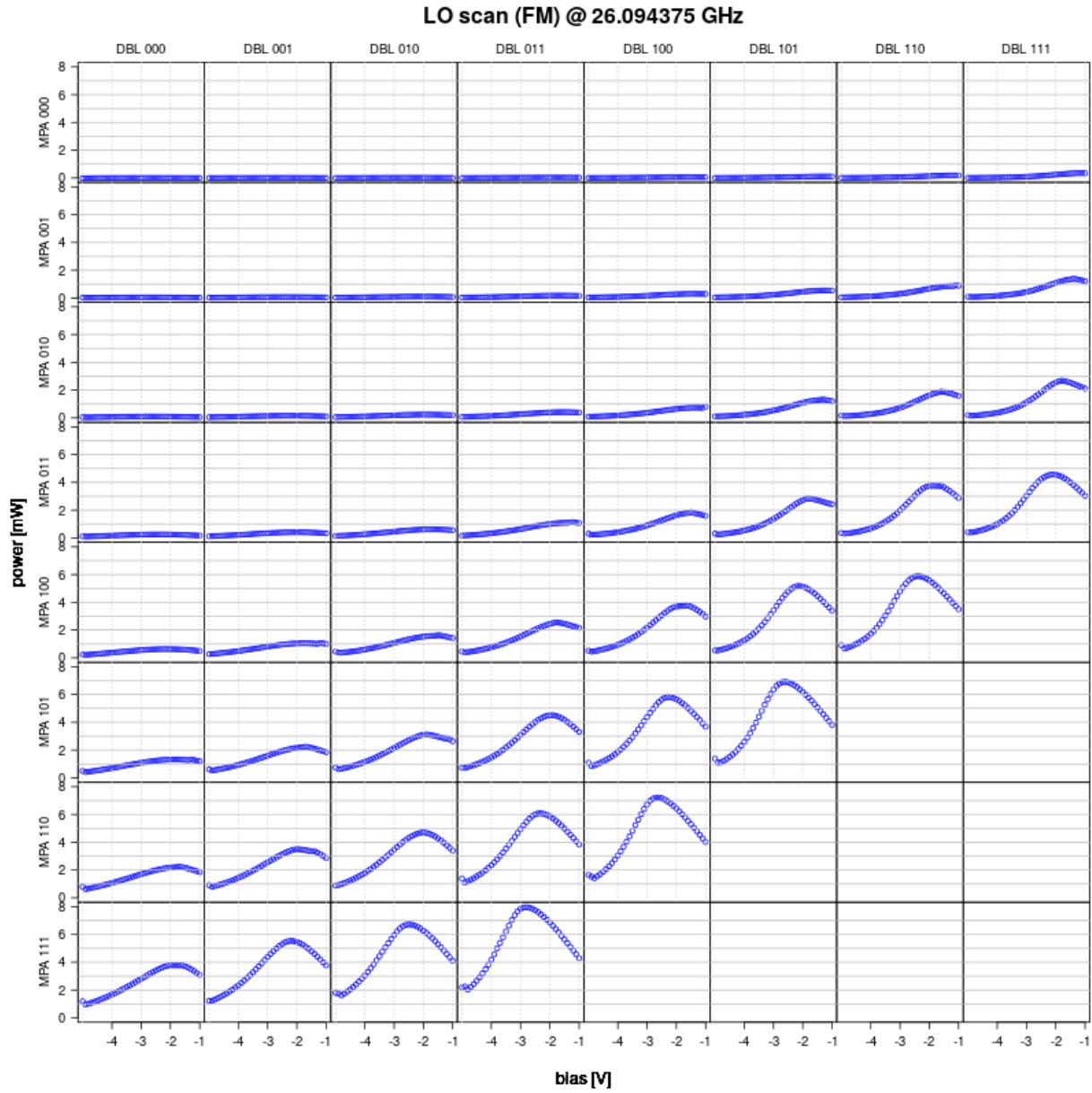


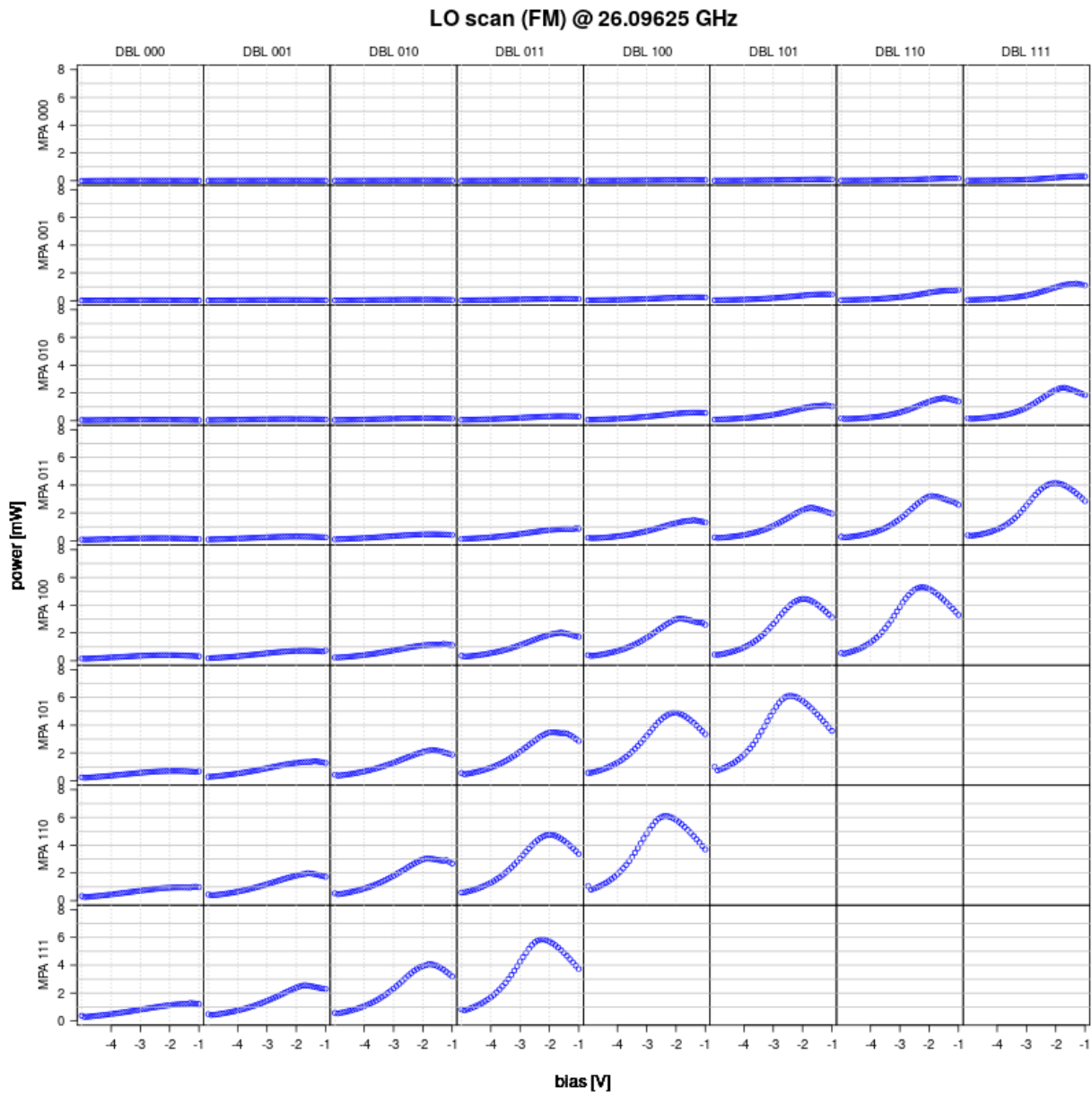


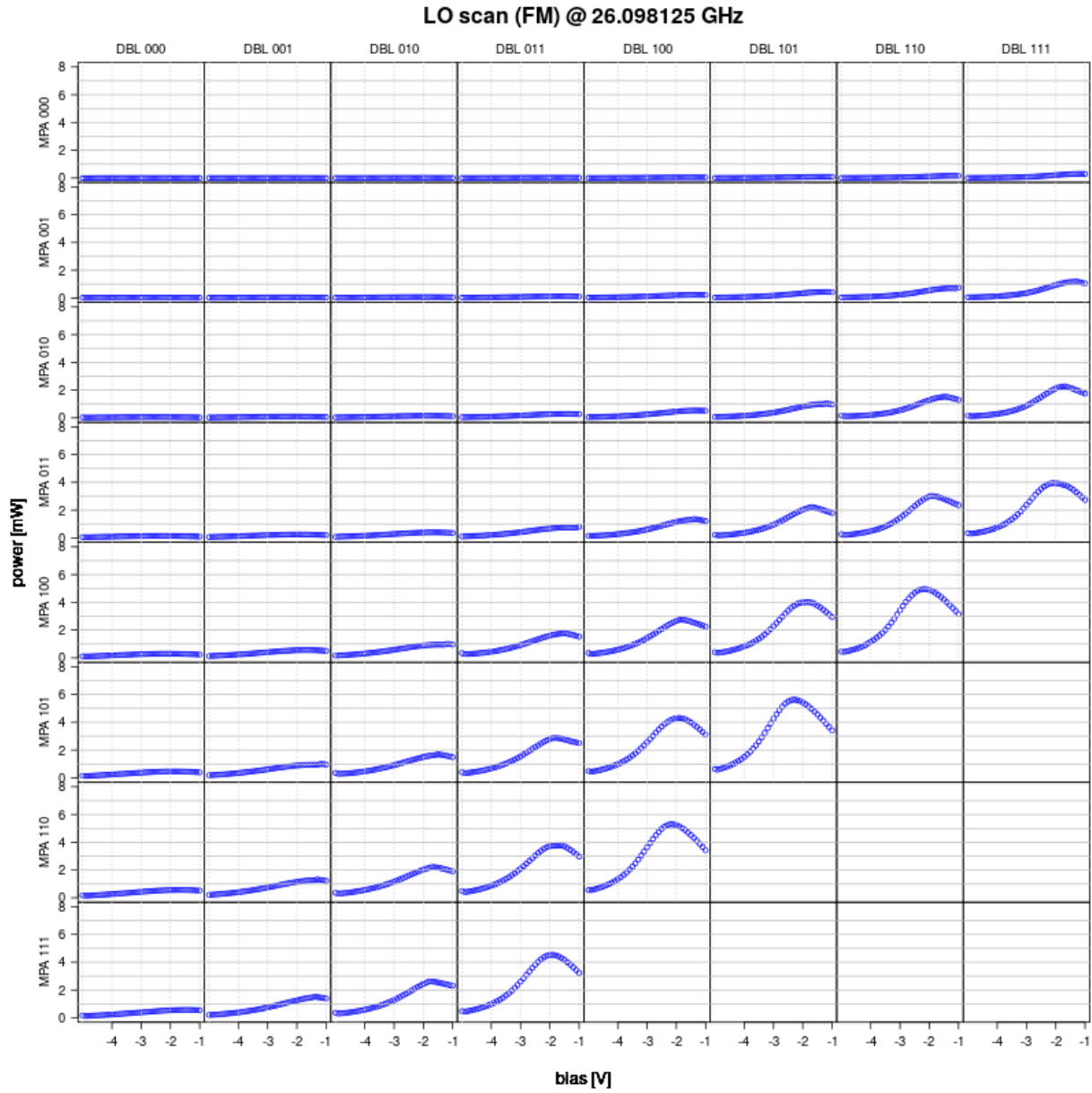


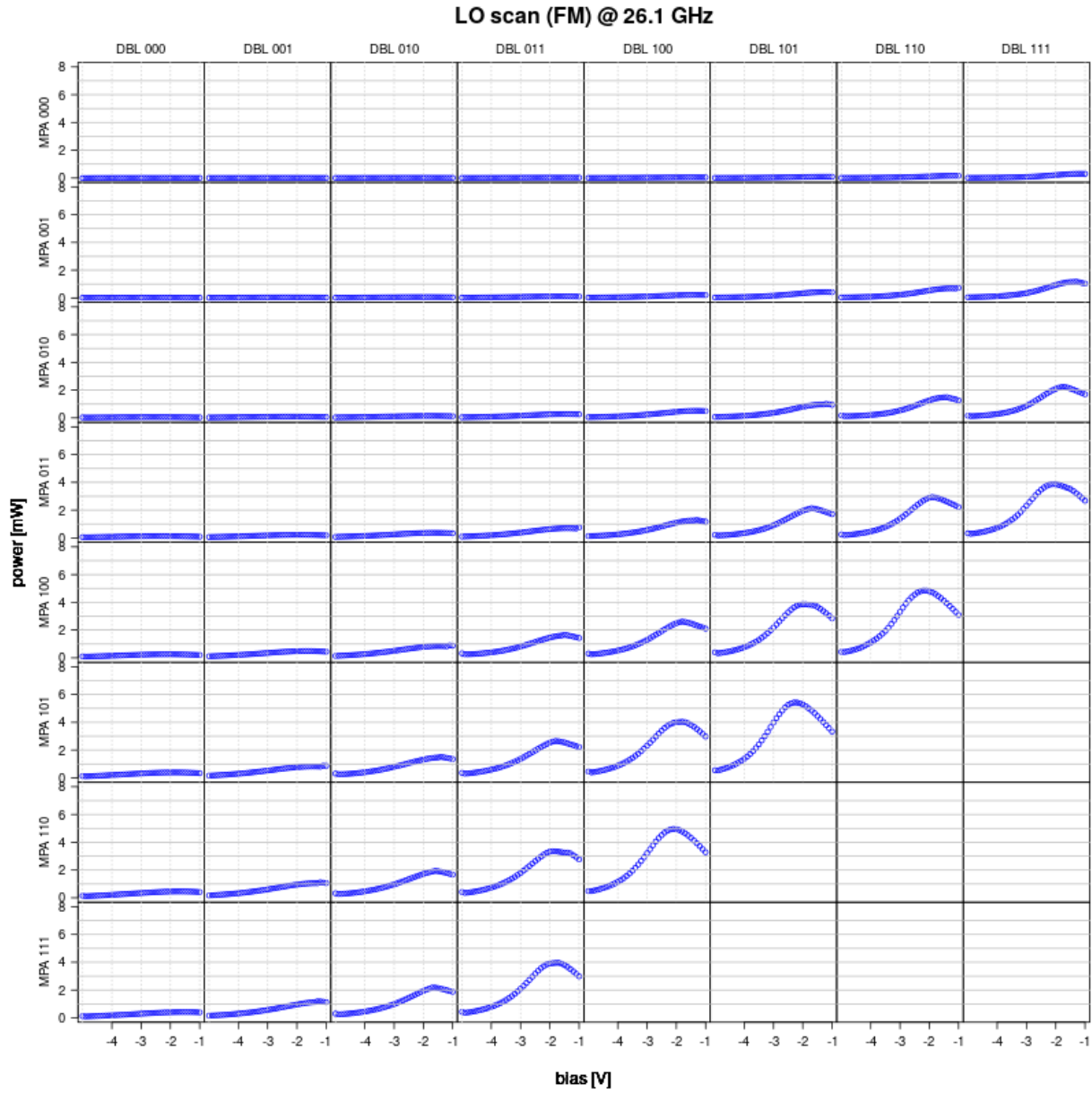
**LO scan (FM) @ 26.0925 GHz**



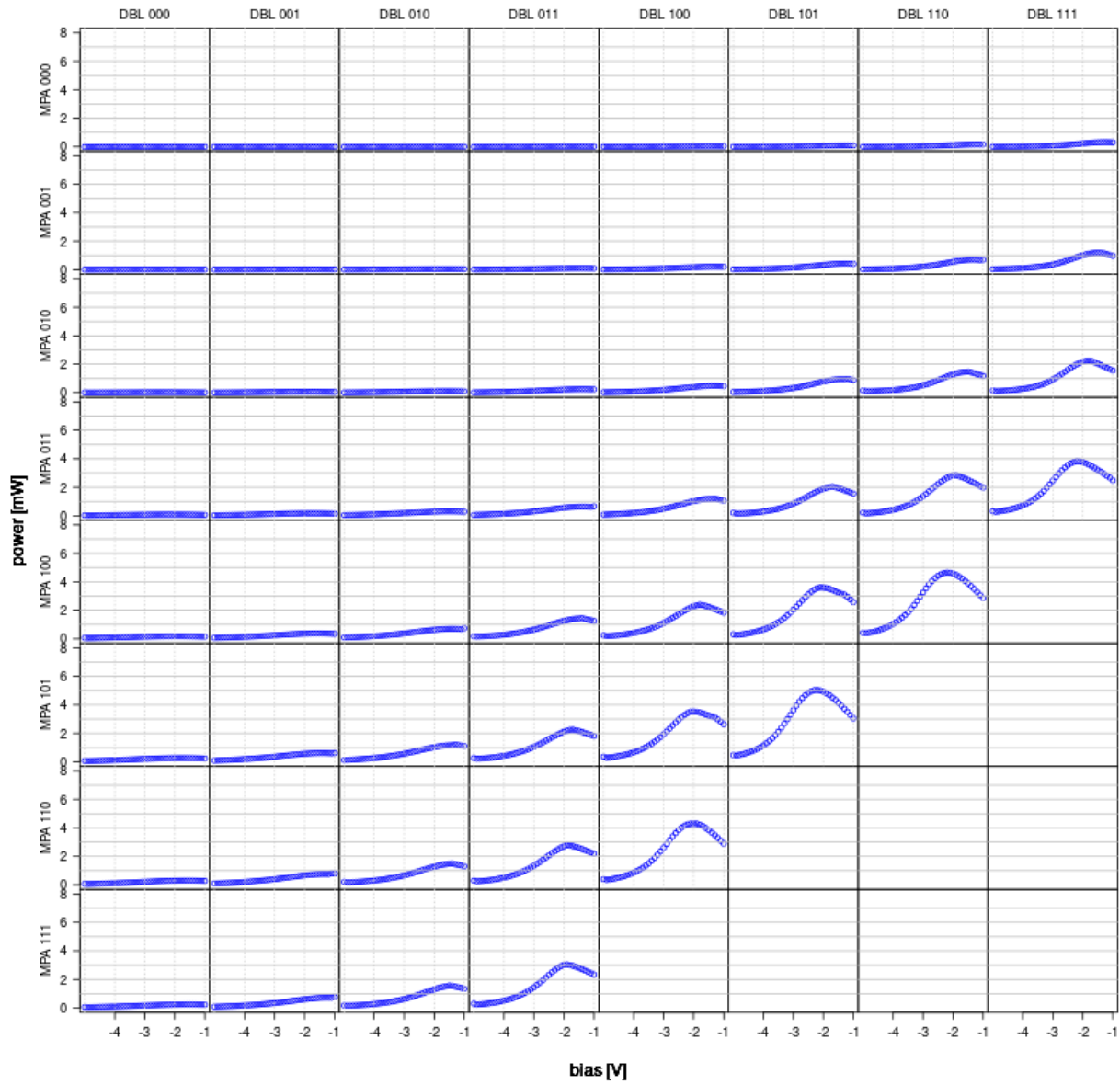


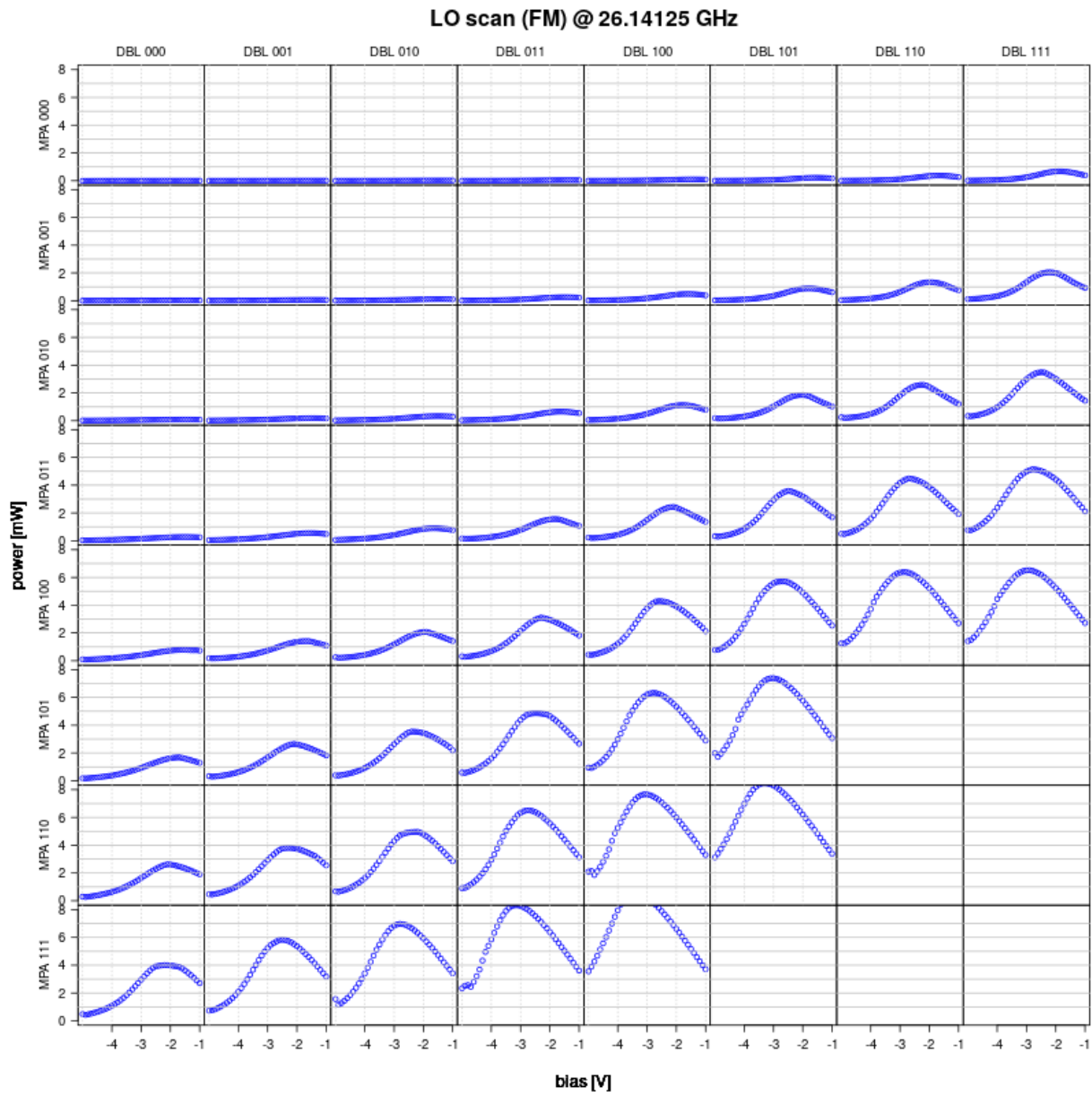




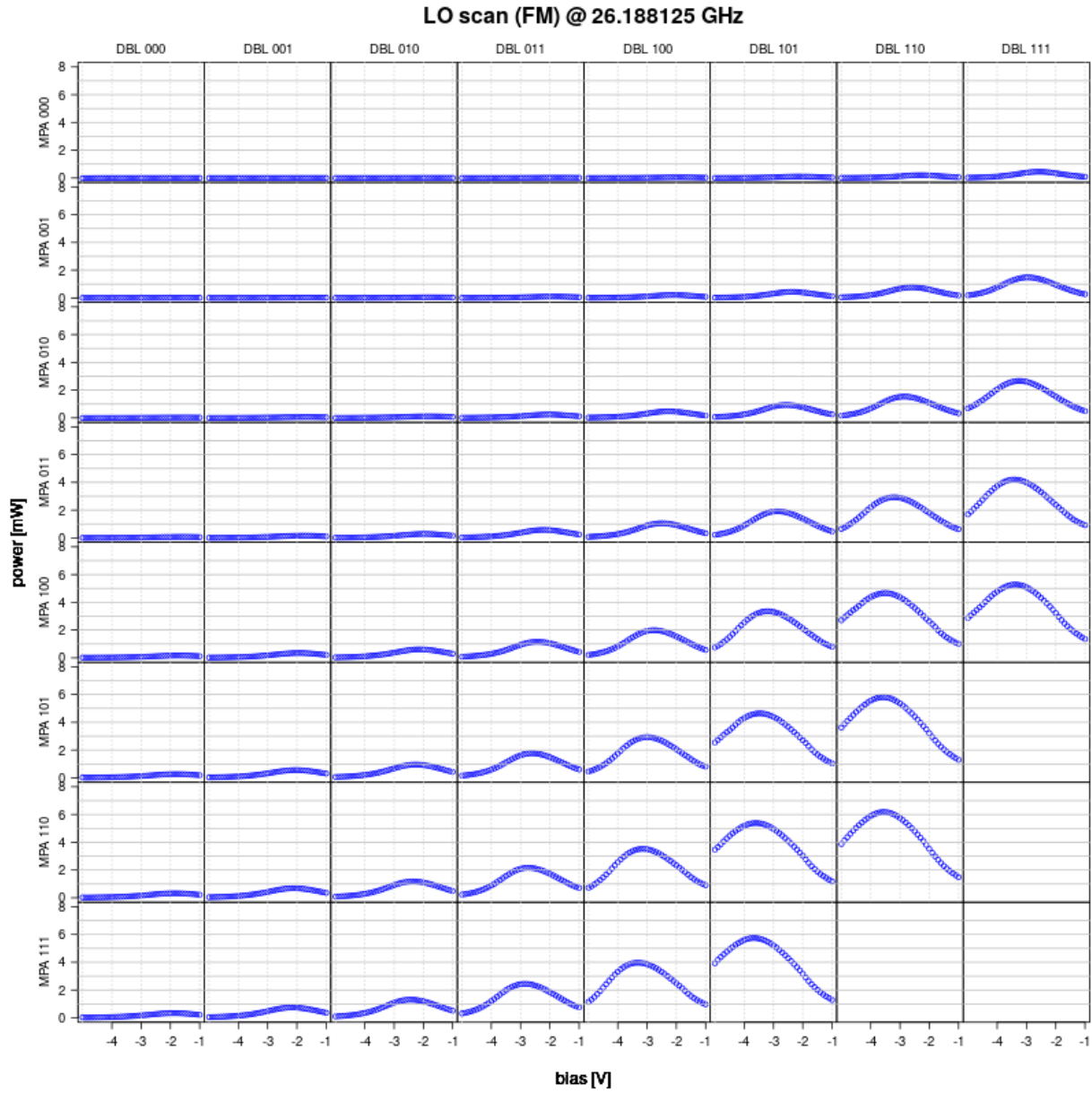


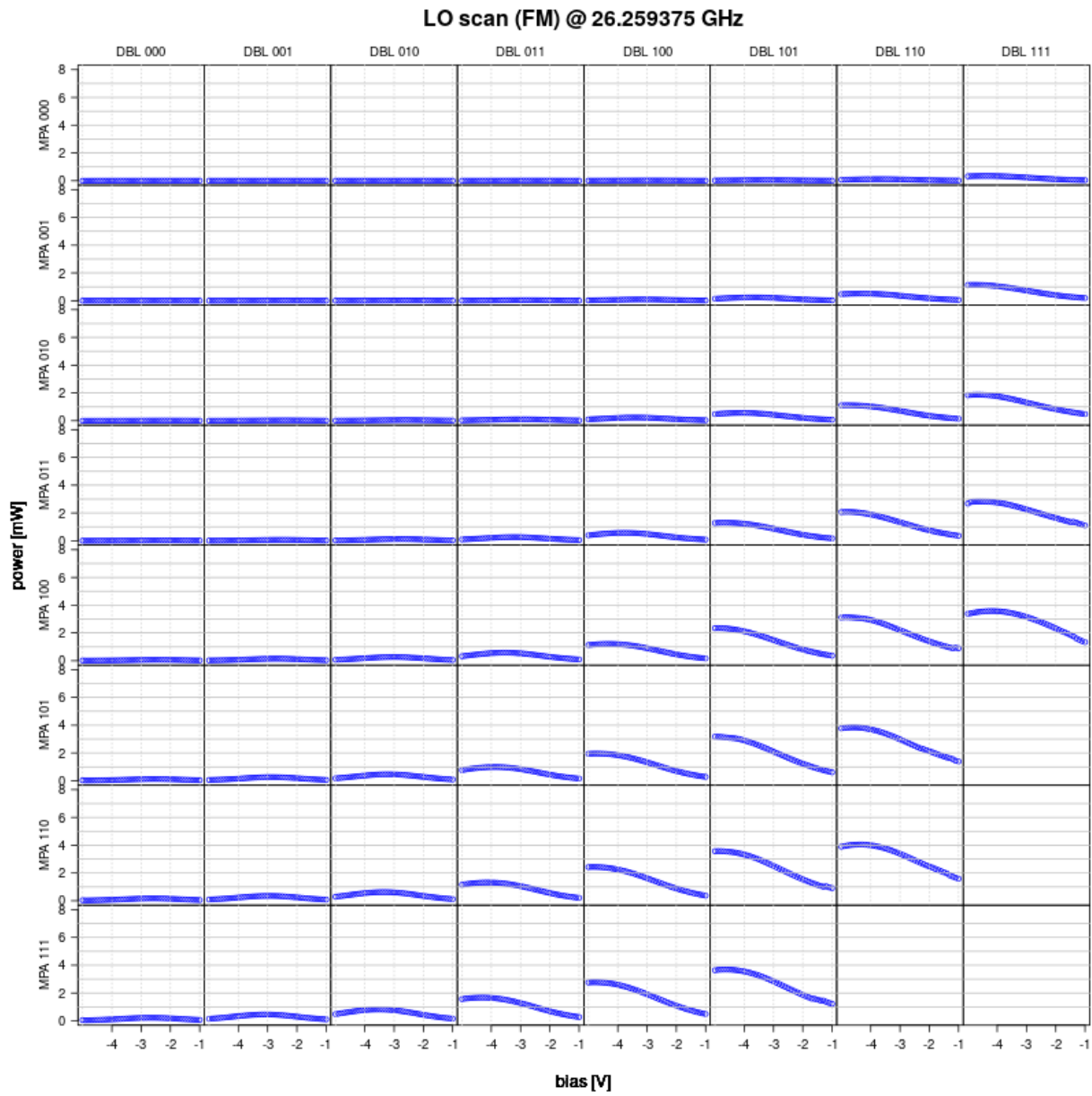
LO scan (FM) @ 26.1075 GHz

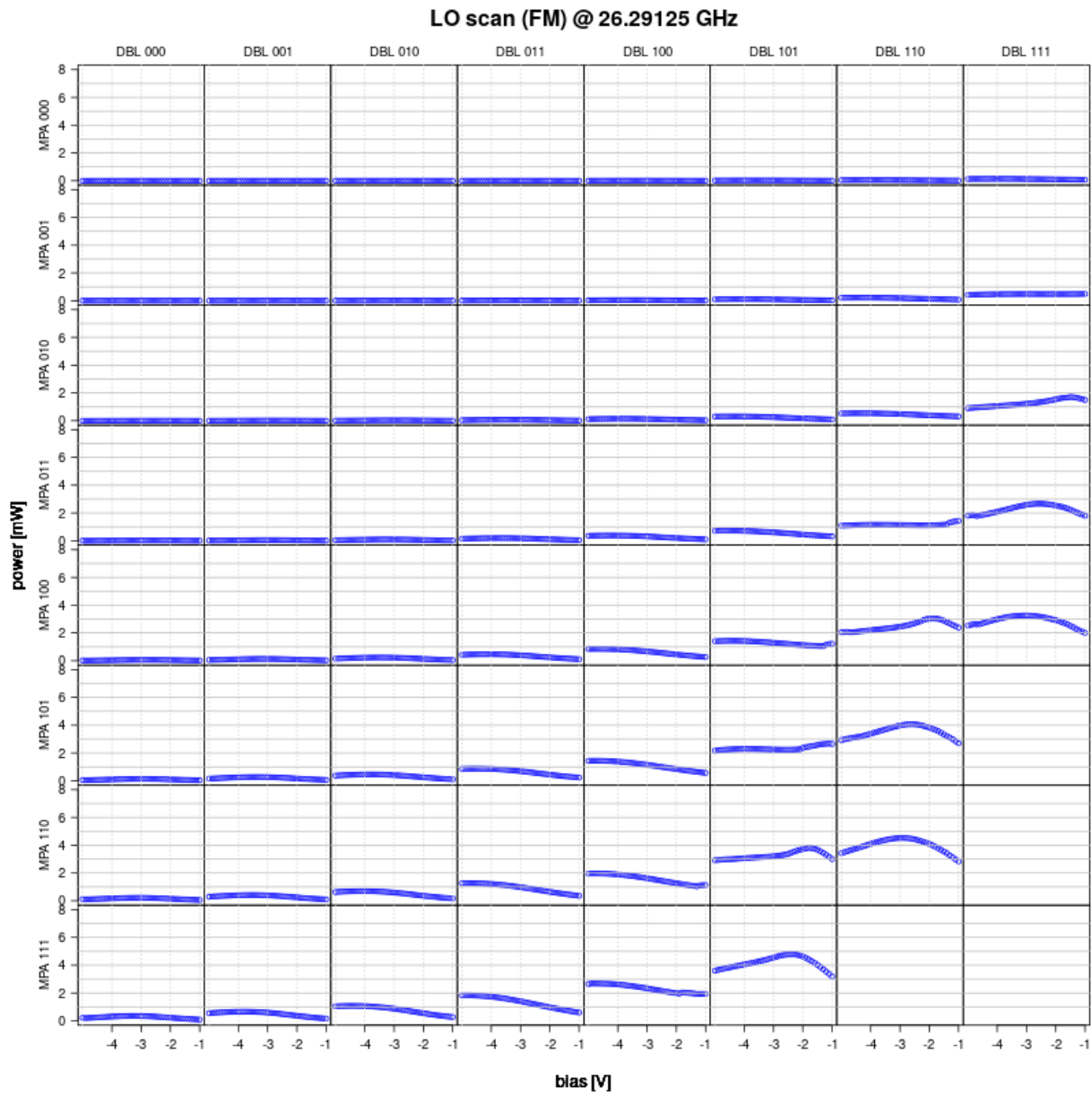


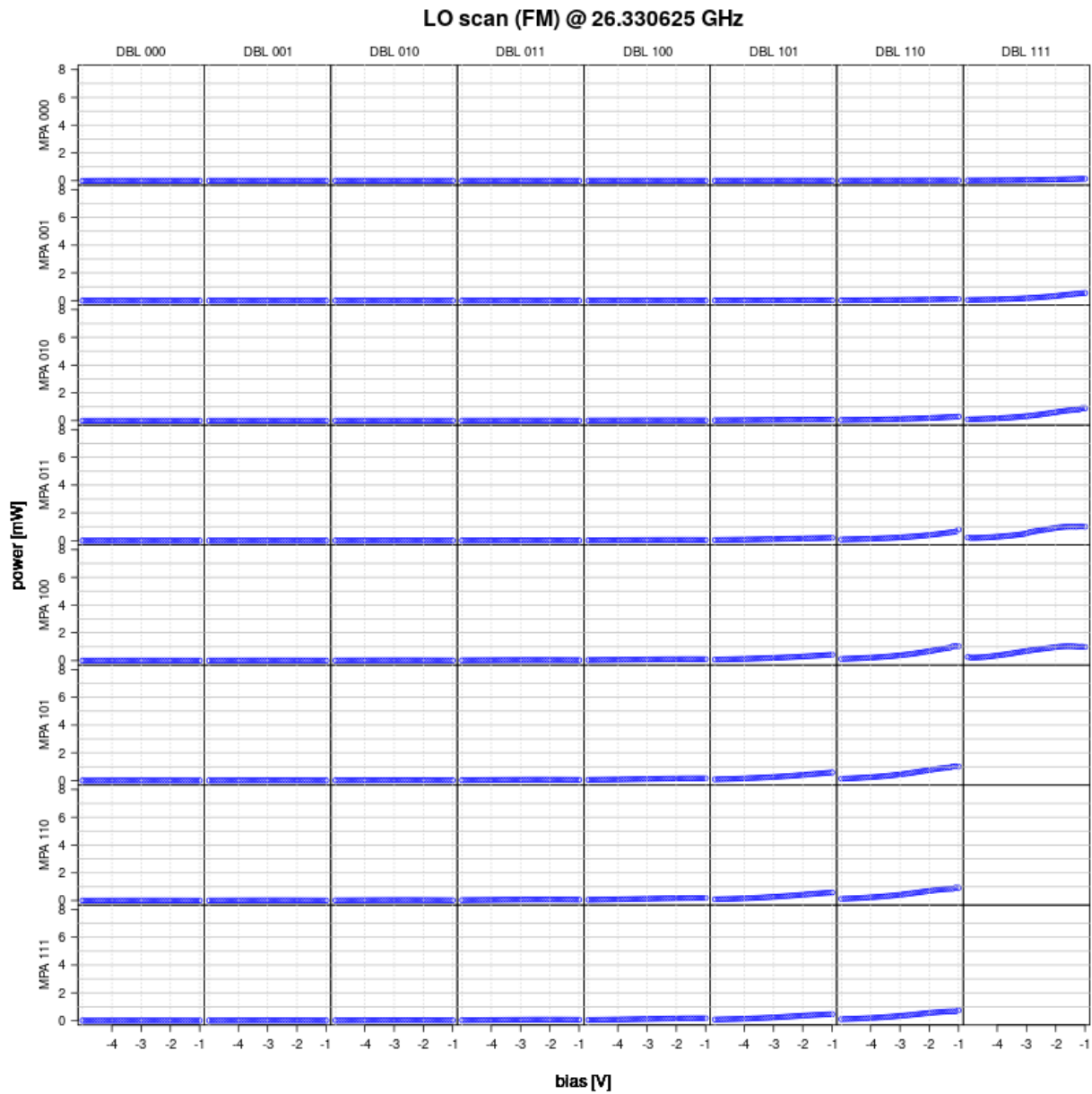












## C Python script

The data were taken in an automated fashion using the following script.

```
import sys
import os
import signal
import time
import logging
import numpy as np

import raspi
import vdi

from instruments.SMF100A import SignalGenerator
from instruments.Keithley2450 import SourceMeter
```

```
logging.basicConfig(level=logging.INFO,
                    format='%(asctime)s - %(name)s - %(levelname)s - %(message)s')
logger = logging.getLogger("LO-600GHz-FM")

# global variable to allow interrupt by Ctrl-C
done = False

def signal_handler(sig, frame):
    global done
    done = True

def keithleyToPM5(voltage):
    return voltage*20/10

def rf_power_levels(rf_power):
    a = np.geomspace(1, 100, 10)*(30+rf_power)/100*(-1)
    a = np.round(a + rf_power - a[0], 2)
    return a[::-1]

signal.signal(signal.SIGINT, signal_handler)

# setting up SMF100A spectrum generator
gen = SignalGenerator()
gen.set_rf(0)

# setting up Keithley 2450 source for drivinnng DBL
keithley = SourceMeter()
keithley.connect()

# setting up VDI power meter
pm5 = vdi.VDI()
pm5.set_zero()
#pm5.set_range(20.0e-3, False)

# setting up Keithley 2450 for analog readout of the VDI power meter
pm5_keithley = SourceMeter('192.168.56.45')
pm5_keithley.connect()

# setting up RasPi GPIO
mpa_gpio = raspi.BiasSetter(raspi.RPGMPA)
doubl_gpio = raspi.BiasSetter(raspi.RPGDOUBLER)

# Load file with calibration data
a = np.loadtxt('LO_scan_calib_table_FM.txt', delimiter=',')

rf_freq = a[:, 0]
rf_power = a[:, 1]
mpa = np.arange(8)
dub = a[:, 2:]

Vmin = -5.
Vmax = -1.
Vstep = .1
Vscan = np.arange(Vmin, Vmax+Vstep, Vstep)
Ilim = 4.8e-3
keithley.setCurrent(Ilim*1e3)

K = 3
for i in range(len(rf_freq)):
    if done:
        break
    print('Press Ctrl+C to interrupt frequency loop')
    logger.info("This is frequency #{} out of {}".format(i, len(rf_freq)))
    # Set SMF100A generator power level (rf_power) and frequency (rf_frequency)
    gen.set_freq(rf_freq[i])
```

```
logger.info("RF frequency is set to {}".format(rf_freq[i]))
pm5.set_zero()

# Switch on RF generator
gen.set_rf(1)
for power_levels in rf_power_levels(rf_power[i]):
    logger.info("RF power is set to {:.3f}".format(power_levels))
    gen.set_power(power_levels)
    time.sleep(.2)

saveFileName = time.strftime(
    '%y%m%d-%H%M%S')+'_'+str(rf_freq[i])+'_'+str(rf_power[i])+'_L0_scan_FM.dat'

# Set MPA TTL levels (set to 4 because one of the TTL does not work)
for mpa_idx in range(8):
    mpa_gpio.setBias(mpa_idx)

# Set DOUBLER TTL levels
for dbl_idx in range(int(dub[i, mpa_idx])+1):
    doubl_gpio.setBias(dbl_idx)

# Scan through voltages. The voltages must be NEGATIVE!
for Voltage in Vscan:
    keithley.setVoltage(Voltage)
    dbl_voltage = keithley.getVoltage()
    dbl_current = keithley.getCurrent()

    if dbl_current > Ilim:
        keithley.setVoltage(Vmin)
        logger.warning(
            "Current too high ({:1.3e} A). The voltage is set to min!".format(
                dbl_current))

        with open(saveFileName, "a") as out:
            tm = time.strftime("%Y-%m-%d_%H-%M-%S", time.gmtime())
            out.write("{}, {:.6f}, {:.2f}, {:.0f}, {:.0f}".format(
                tm, rf_freq[i], rf_power[i], mpa_idx, dbl_idx))
            out.write("{:.2f}, {:.3e}".format(
                dbl_voltage, dbl_current))
            out.write("{:.0f}, {:.0f}\n".format(0, 0))
            break

    time.sleep(0.3)
    power = np.zeros(K)
    for ii in range(K):
        power[ii] = keithleyToPM5(pm5_keithley.getVoltage())
        time.sleep(.1)

# Printing results on the screen
print('Freq = {:.6f} GHz / MPA = {:.0f} / DBL = {:.0f} / P = {:.3f}+{:.4f}
      mW/ DBL_V = {:.3f} V/ DBL_I
      = {:.13e} A'.format(
        rf_freq[i], mpa_idx, dbl_idx, np.mean(power), np.std(power),
        dbl_voltage, dbl_current)
    )

with open(saveFileName, "a") as out:
    tm = time.strftime("%Y-%m-%d_%H-%M-%S", time.gmtime())
    out.write("{}, {:.6f}, {:.2f}, {:.0f}, {:.0f}".format(
        tm, rf_freq[i], rf_power[i], mpa_idx, dbl_idx))
    out.write("{:.2f}, {:.3e}".format(
        keithley.getVoltage(), keithley.getCurrent()))
    out.write("{:.3e}, {:.4e}\n".format(
        np.mean(power), np.std(power)))

keithley.setVoltage(Vmin)
logger.info("DBL voltage is set to {:.2f}".format(Vmin))
```

```
gen.set_power(-50)
gen.set_rf(0)
logger.info("power is off")
time.sleep(.2)
mpa_gpio.setBias(0)
logger.info("MPA TTL is 0")
time.sleep(.2)
doubl_gpio.setBias(0)
logger.info("doubler TTL is 0")
time.sleep(.5)

keithley.close()
raspi.cleanup()
gen.close()
```