



## Application Notes

### MG216 AoA 应用 Extended Adv

#### Revision History:

Rev. No.	History	Issue Date	Remark
0.1	First version	Dec 30, 2021	First Release

#### Important Notice:

MACROGIGA reserves the right to make changes to its products or to discontinue any integrated circuit product or service without notice. MACROGIGA integrated circuit products are not designed, intended, authorized, or warranted to be suitable for use in life-support applications, devices or systems or other critical applications. Use in such applications is done at the sole discretion of the customer. MACROGIGA will not warrant the use of its devices in such applications.



## 目录

1. 概述.....	3
2. 控制接口.....	3
3. 广播包类型.....	3
4. 控制方法.....	4
4.1 传统广播包控制参数生成.....	4
4.2 扩展广播包控制参数生成.....	4
4.3 周期性广播包控制参数生成.....	5
5. 扩展广播包广播数据的更新方法.....	6
6. 芯片时钟校准.....	6
7. FAQ.....	6
8. 附录.....	8
8.1 参数生成.....	8
8.2 手机兼容性.....	8



## 1. 概述

本文介绍 MG216 芯片作为广播设备进行符合 BLE 核心规范 V5.1 工作的一些基本原理，以及在各对应工作模式下的控制原理及方法。

## 2. 控制接口

芯片包含有三种类型的控制接口，分别为：

- SPI
- SPI - HCI
- I2C Slave

建议使用 SPI 接口。

## 3. 广播包类型

芯片遵从的蓝牙核心规范 V5.1，主要包含如下三类广播类型，分别为

- 普通广播包
- 扩展广播包
- 周期性的扩展广播包（支持 AoA）

普通广播包一次最多广播的数据长度为 31 字节，广播数据仅在 1Mbps 的物理通道上广播，可以广播的广播包类型包括：

- 不可连接广播包（ADV\_NONCONN\_IND）
- 定向广播包（ADV\_DIRECT\_IND）

扩展广播包可发送长度大于 31 字节的广播包，广播数据会占用主广播通道（Primary Advertising Channel）及次广播通道（Secondary Advertising Channel），广播可在 1Mbps、2Mbps、500Kbps 及 125Kbps 的物理通道上广播，广播包类型主要有如下的一些组合

- 包含一个长包的扩展广播（ADV\_EXT\_IND+AUX\_ADV\_IND）
- 包含两个长包的扩展广播（ADV\_EXT\_IND+AUX\_ADV\_IND+AUX\_CHAIN\_IND）

包类型同样可以是 NONCONN 或 DIRECT，可按需配置。

周期性广播包可按需动态更新广播包内容，也可以按需发送 AoA 定位信息，具体包含的广播包情况如下



周期性广播包 (ADV\_EXT\_IND+AUX\_ADV\_IND+AUX\_SYNC\_IND)

AoA (ADV\_EXT\_IND+AUX\_ADV\_IND+AUX\_SYNC\_IND, AoA enable)

## 4. 控制方法

为简化代码的控制，用户可根据参数生成工具将相关参数按需配置并生成参数头文件 (mg\_adv\_par\_cfg.h)，导入到工程中直接编译即可完成基本的功能。参数配置工具如下图所示。

图 1 广播参数生成工具

### 4.1 传统广播包控制参数生成

不使用。传统广播包使用 MG223 的例程。

### 4.2 扩展广播包控制参数生成

按需根据期望的广播间隔值，选择广播期望速率所在的物理通道类型，根据需要广播数据量的多少选择是否使能 AUX\_CHAIN\_IND 包，点击对应【Gen\_Par】即可生成参数配置文件，生成的配置文中关键数据如下所示：



```
u8 g_adv_type = MG_TYPE_ADV_EXT_IND;
u8 g_adv_ext1_type = MG_TYPE_ADV_EXT_IND;
u8 g_adv_ext2_type = MG_TYPE_AUX_ADV_IND;
u8 g_adv_ext3_type = MG_TYPE_AUX_CHAIN_IND;

u16 g_adv_interval = 150; //unit ms, for sync case, it is syn interval

u8 gc_adv_data_n[] = {0};
u8 gc_adv_data_n_len = 0; //0 means not used

u8 g_adv_data_pkt2[] = { //Extended(AUX_ADV_IND) AdvData
    0x02, 0x01, 0x04,
};
u8 g_adv_data_pkt2_len = 3;

u8 g_adv_data_pkt3[] = { //ADV_SYNC_IND or AUX_CHAIN_IND AdvData
    0x03, 0x09, 0x6D, 0x67,
};
u8 g_adv_data_pkt3_len = 4;

u8 g_ext_phy_sel = MG_ADV_PHY_125K;

u8 g_aux_ch_sel = MG_ADV_CH_AUTO;
u8 g_aux_phy_sel = MG_ADV_PHY_125K;
u8 g_interval_2_aux_step_num = 4;
u16 g_interval_2_aux_step[] = {48, 201, 348, 501, };

u8 g_interval_2_sync_step_num = 0; // 0 means not used
u16 g_sync_offset1_2_aux = 0; // 0 means not used
u16 g_sync_offset2_2_sync = 0; // 0 means not used
u8 g_aoa_enable_flag = 0; // 0 means disabled or not used

u8 g_chain_ch_sel = MG_ADV_CH_AUTO;
u16 g_chain_offset = 48; //unit ms
```

### 4.3 周期性广播包控制参数生成

按需根据期望的广播间隔值，选择周期性广播包的周期，点击对应【Gen\_Par】即可生成参数配置文件，生成的配置文中关键数据如下所示：

```
u8 g_adv_type = MG_TYPE_ADV_EXT_IND;
u8 g_adv_ext1_type = MG_TYPE_ADV_EXT_IND;
u8 g_adv_ext2_type = MG_TYPE_AUX_ADV_IND;
u8 g_adv_ext3_type = MG_TYPE_AUX_SYNC_IND;

u16 g_adv_interval = 200; //unit ms, for sync case, it is syn interval

u8 gc_adv_data_n[] = {0};
u8 gc_adv_data_n_len = 0; //0 means not used

u8 g_adv_data_pkt2[] = { //Extended(AUX_ADV_IND) AdvData
    0x02, 0x01, 0x04,
};
u8 g_adv_data_pkt2_len = 3;

u8 g_adv_data_pkt3[] = { //ADV_SYNC_IND or AUX_CHAIN_IND AdvData
    0x03, 0x09, 0x6D, 0x67,
};
u8 g_adv_data_pkt3_len = 4;

u8 g_ext_phy_sel = MG_ADV_PHY_1M;

u8 g_aux_ch_sel = MG_ADV_CH_AUTO;
u8 g_aux_phy_sel = MG_ADV_PHY_2M;
u8 g_interval_2_aux_step_num = 0; // 0 means not used
u16 g_interval_2_aux_step[] = {0}; //offset, 0 means not used

u8 g_interval_2_sync_step_num = 4;
u16 g_sync_offset1_2_aux = 66; // unit ms
u16 g_sync_offset2_2_sync = 66; // unit ms
u8 g_aoa_enable_flag = 1; // 0 means disabled or not used

u8 g_chain_ch_sel = 0xFF; //0xFF means NOT used
u16 g_chain_offset = 0; // 0 means not used
```



## 5. 扩展广播包广播数据的更新方法

扩展广播包的内容更新可以有两种更新方式：

- 静态更新
- 动态盲更新
- 动态完整更新

静态更新模式是指设备处于非广播状态下对广播包数据进行更新。

动态盲更新模式为广播数据随时进行更新，不论该数据是否正在传输。这种更新方式会产生一次空的广播数据，但更新结束后的广播内容即恢复正常。

动态完整更新模式为在更新的时候硬件会做保护，如果该广播数据正在被传输，那么会更新失败，可通过查询对应寄存器的状态获取是否更新成功。这种更新方式能确保发出的广播包的数据是完整的。

接口函数 `update_adv_ext_ind_data()` 用于更新第一个长广播包（`AUX_ADV_IND`）的数据

接口函数 `update_aux_chain_ind_data()` 用于更新第二个长广播包（`AUX_CHAIN_IND`）的数据

接口函数 `update_aux_sync_ind_data` 用于更新周期性广播包（`AUX_SYNC_IND`）的数据

上述接口函数如果使用【`DISABLE_LOCK_DATA_BIT`】表示使用动态盲更新模式，否则为动态完整更新模式。

## 6. 芯片时钟校准

由于芯片的基准低频时钟会随着温度、电压等因素产生一定幅度的漂移抖动，因此上层应用程序可按需周期性调用对应校准更新程序接口，对应的接口函数如下（具体可参考 SDK 示例代码）

校准第一个长包广播时钟 `update_adv_ext_ind_par()`

校准第二个长包广播时钟 `update_aux_chain_ind_par()`

校准周期性广播包时钟 `update_aux_sync_ind_par()`

上述时钟校准更新接口函数仅在**扩展广播包**的应用场景下才有调用的要求。

## 7. FAQ

1. 扩展广播包允许的广播数据长度是多少？

扩展广播包有三种组合，分别为：

- 1) 一个扩展广播包（`AUX_ADV_IND`）
- 2) 一个扩展广播包以及 一个扩展链广播包（`AUX_ADV_IND + AUX_CHAIN_IND`）
- 3) 周期性广播包（`AUX_ADV_IND + AUX_SYNC_IND`）





由于扩展广播包除了包含广播数据内容（Adv Data）外还可以同时包含一些额外必需的数据，因此不同情况下每种广播包内实际可包含广播数据的最大长度是不同的。

### 1) 一个扩展广播包（AUX\_ADV\_IND）

$\text{Len\_Adv\_Data\_Max} = 255 - 2 - \text{Len\_AdvA\_O6} - \text{Len\_ADI\_2} - \text{Len\_TxPower\_O1} - \text{Len\_ACAD\_Ox} = 251 \text{ (max)}$

其中可选内容为：**Len\_AdvA\_O6 (6Bytes)**，**Len\_TxPower\_O1 (1Byte)**，**Len\_ACAD\_Ox**

		Common Extended Advertising Payload Format fields								
Adv Mode	Event Type	AdvA	TargetA	CTE Info	ADI	Aux Ptr	Sync Info	Tx Power	ACAD	Adv Data
0b00	Non-Connectable and Non-Scannable Undirected	C4	X	X	M	O	O	O	O	O
0b00	Non-Connectable and Non-Scannable Directed	C4	C2	X	M	O	O	O	O	O
0b01	Connectable Undirected	M	X	X	M	X	X	O	O	O
0b01	Connectable Directed	M	M	X	M	X	X	O	O	O
0b10	Scannable Undirected	M	X	X	M	X	X	O	O	X

Table 2.5: Common Extended Advertising Payload Format fields permitted in the AUX\_ADV\_IND PDU

### 2) 一个扩展广播包以及 一个扩展链广播包（AUX\_ADV\_IND + AUX\_CHAIN\_IND）

$\text{Len\_Adv\_Data\_Max\_AUX\_ADV} = 255 - 2 - \text{Len\_AdvA\_O6} - \text{Len\_AuxPtr\_3} - \text{Len\_ADI\_2} - \text{Len\_TxPower\_O1} - \text{Len\_ACAD\_Ox} = 248 \text{ (max)}$

		Common Extended Advertising Payload Format fields								
Adv Mode	Event Type	AdvA	TargetA	CTE Info	ADI	Aux Ptr	Sync Info	Tx Power	ACAD	Adv Data
0b00	Chained data	X	X	C5	C3	O	X	O	X	O
0b01 to 0b11	RFU									

Table 2.7: Common Extended Advertising Payload Format fields permitted in the AUX\_CHAIN\_IND PDU

$\text{Len\_Adv\_Data\_Max\_AUX\_CHAIN} = 255 - 2(1) - \text{Len\_TxPower\_O1} - \text{Len\_ACAD\_Ox} = 254 \text{ (max)}$

### 3) Sync 周期性广播包

扩展广播包包含的广播包数据情况为

$\text{Len\_Adv\_Data\_Ex\_Max} = 255 - 2 - \text{Len\_AdvA\_O6} - \text{Len\_ADI\_2} - \text{Len\_Sync\_Info\_18} - \text{Len\_TxPower\_O1} - \text{Len\_ACAD\_Ox} = 233 \text{ (max)}$

		Common Extended Advertising Payload Format fields								
Event Type	Adv Mode	AdvA	TargetA	CTE Info	ADI	Aux Ptr	Sync Info	Tx Power	ACAD	Adv Data
Non-Connectable and Non-Scannable Undirected without auxiliary packet	0b00	M	X	X	X	X	X	O	X	X
Non-Connectable and Non-Scannable Undirected with auxiliary packet	0b00	C1	X	X	M	M	X	C1	X	X

Table 2.4: Common Extended Advertising Payload Format fields permitted in the ADV\_EXT\_IND PDU

$\text{Len\_Adv\_Data\_Sync\_Max} = 255 - 2(1) - \text{Len\_CteInfo\_O1} - \text{Len\_TxPower\_O1} - \text{Len\_ACAD\_Ox} = 254 \text{ (max)}$

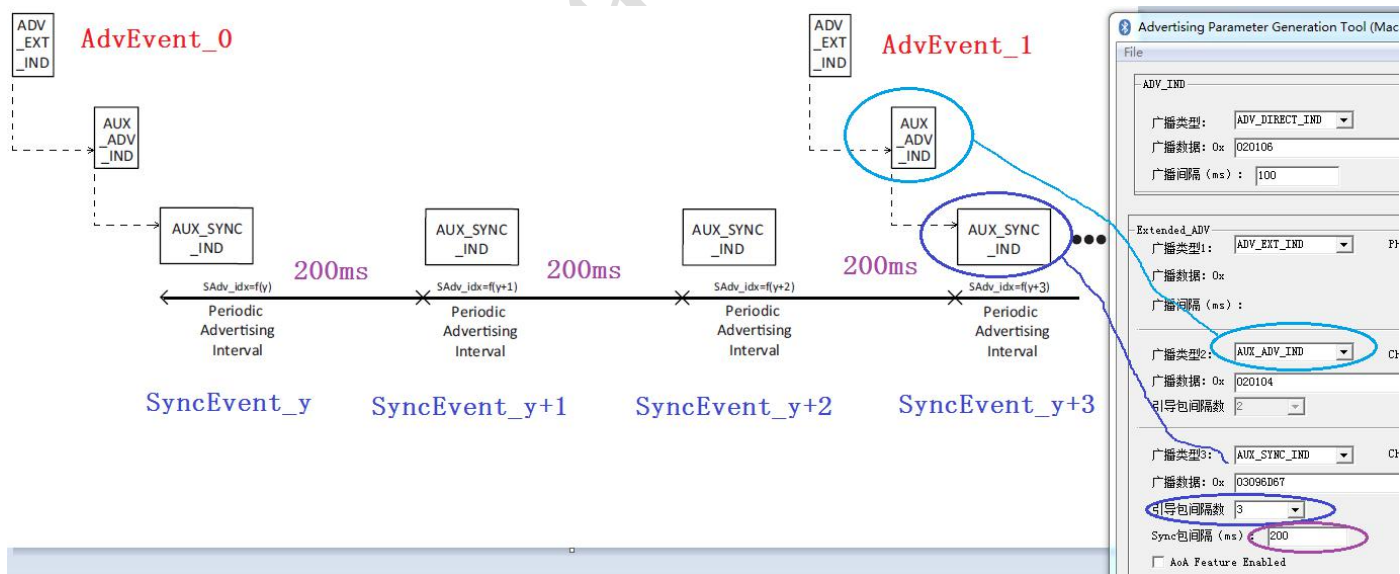
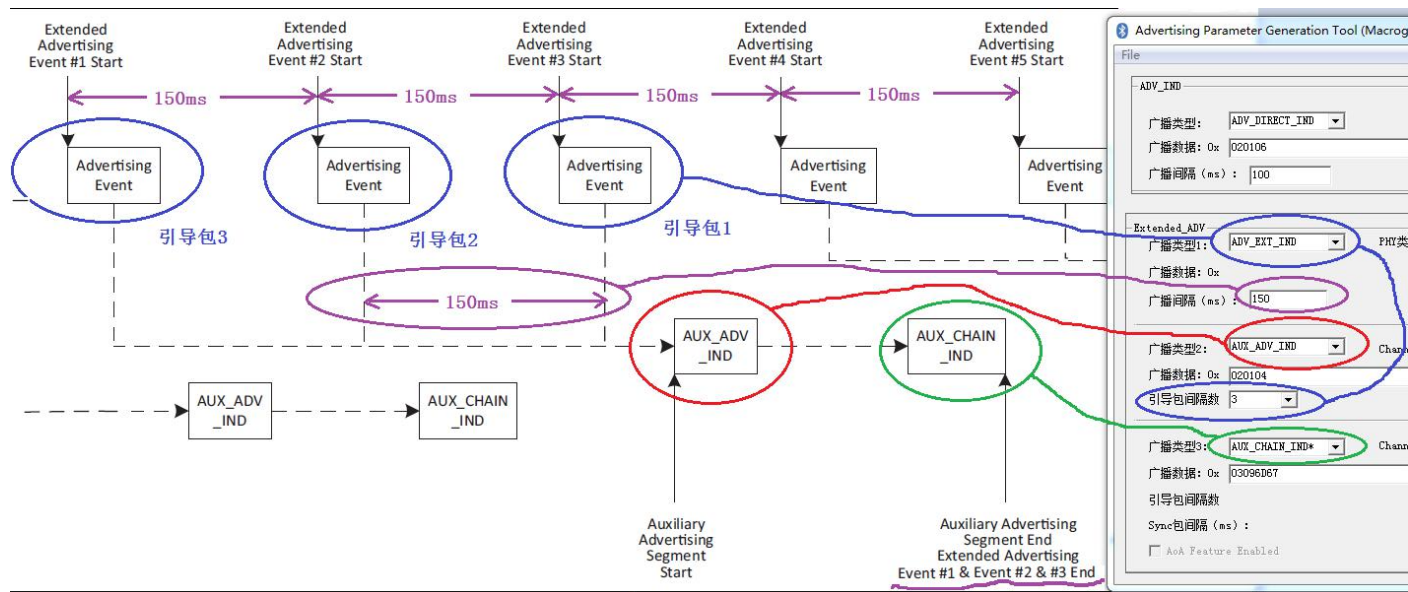
		Common Extended Advertising Payload Format fields								
Adv Mode	Event Type	AdvA	TargetA	CTE Info	ADI	Aux Ptr	Sync Info	Tx Power	ACAD	Adv Data
0b00	Non-Connectable and Non-Scannable Undirected or Directed	X	X	O	X	O	X	O	O	O
0b01 to 0b11	RFU									

Table 2.6: Common Extended Advertising Payload Format fields permitted in the AUX\_SYNC\_IND PDU



## 8. 附录

### 8.1 参数生成



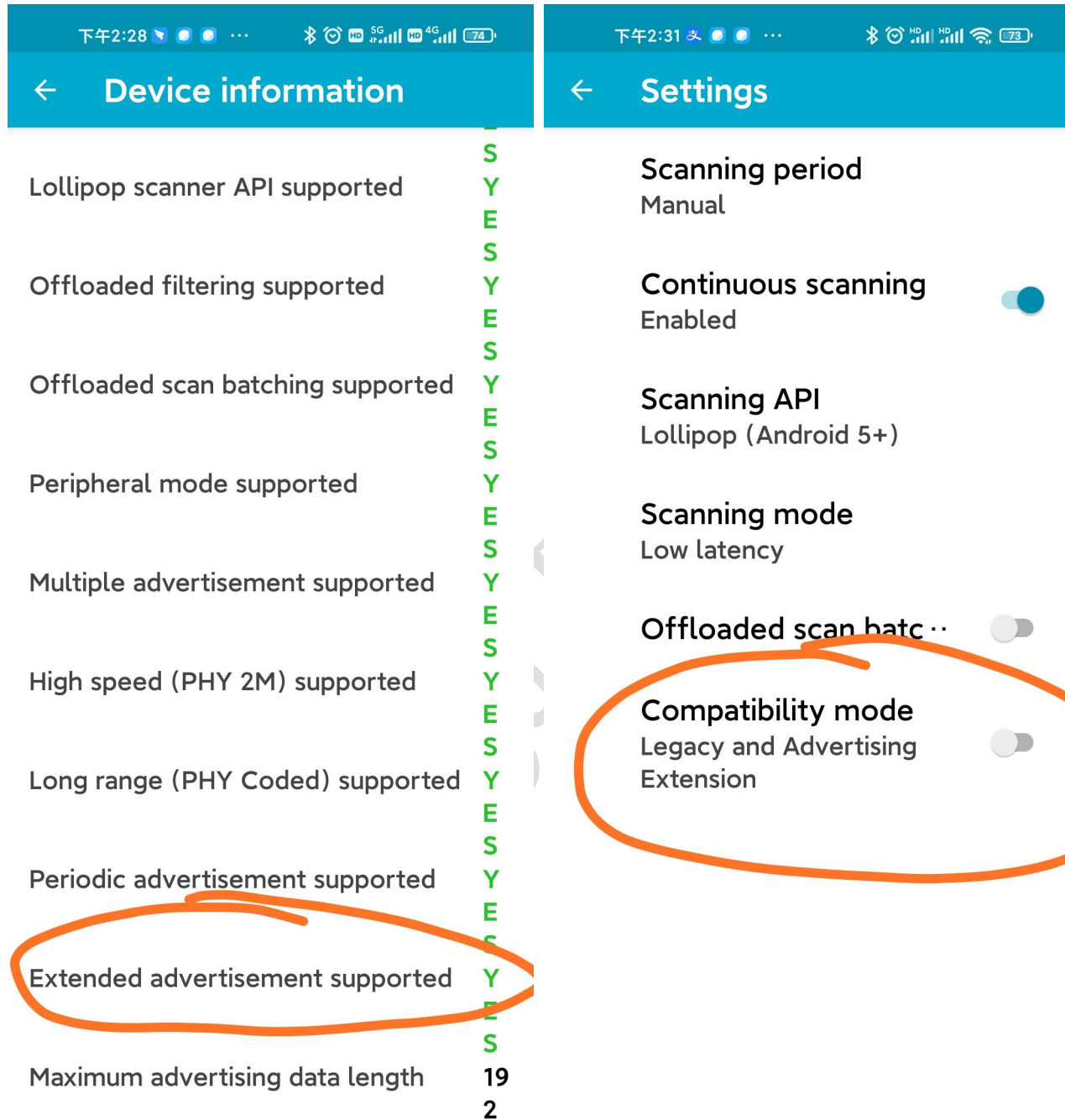
### 8.2 手机兼容性

扩展广播包需要手机支持，可以使用 nRF Connect App 来检查手机是否支持。





nrf connect app 的 device info 支持扩展广播包显示一定要 YES。



另外，setting 设置中 compatibility mode 不要选中。这样的手机可以收到扩展广播包，如下图所示。

