# EM6819_Beacon Parameter Dictionary

## Parameters in Comma Separated Value (CSV) Format

Parameter values are defined in a Comma Separated Value or CSV file. The first record lists the names of the parameters that appear in 1:1 correspondence to the values in all subsequent records. Following this, each line in the file defines one record of values for a single beacon. Here is a simple example with three fields and data records for 4 different beacons.

```
beaconPeriod,outputPowerLevel,beaconMode
100,6,0x13
200,4,0x15
500,2,0x16
750,0,0x17
```

Any value in any data record may be omitted but its trailing comma must be present to preserve the 1:1 relationship between names and values. The result of omitting a value or set of values is described by a few simple rules given in the USAGE document. Spaces are not needed for any parameter names or values and their use is discouraged. Quotes are allowed but not required.

## Beacon Dictionary of Parameter Names and Values

Although the parameters themselves have changed very little since the 2.3.0 release, subtle and sometime not-so-subtle changes to the semantics of the values have occurred over time. The following descriptions are for the current firmware release. Earlier versions may generate errors for values that comply with the following specifications. Note that the lastest version of the SDK which includes the latest documentation on the beacon programming and update utilities is available on the EM Developer Forum.

### deviceAddress

: This field can be used to provide each beacon with the unique 48-bit Bluetooth radio MAC address described in the Bluetooth 4.0 Core specification, Volume 6, Part B, Section 1.3. The default parameter value is 0x0EF3EE010205. This makes it a *locally administered public address*.

: Later EM6819 devices have a globally unique *universally administered public address* permanently programmed in flash memory Sector 5, Row 62, bytes 24:29, outside the parameter block. To use this pre-programmed MAC address, set the value of the deviceAddress parameter to zero. This will signal the firmware to use the globally unique value. If the deviceAddress parameter is set to zero in a device that does not have a pre-programmed globally unique public address, the beacon will enter a failure state with both LEDs remaining on as soon as power is applied. This can be corrected by providing a non-zero value for the deviceAddress parameter.

: *Format and Validation:* The Beacon Tool scripts will accept any value for this parameter provided it does not exceed 6 bytes (48 bits) in length. In the CSV file, the deviceAddress must appear most significant byte first.

### idDataUUID

: The Universally Unique Identifier (UUID) is used by both IdData and AltBeacon packets. It is a 128-bit field used by iBeacon and AltBeacon technologies to reference data that can be retrieved by mobile applications that receive the packet. The default value provided by the firmware is 0x699EBC80E1F311E39A0F0CF3EE3BC012. Also see the idDataMajorID and idDataMinorId fields, below.

: *Format and Validation:* The Beacon Tools will accept any value that does not exceed 16 bytes. When encoded with hexadecimal notation, it must be represented MSB first.

### idDataMajorId

: This is a beacon owner-specified value that, together with the idDataMinorId and the idDataUUID make the beacon unique. The default value is 0x005A, (EM Microelectronic's assigned Bluetooth company code). This value is also transmitted in AltBeacon packets.

: *Format and Validation:* The Beacon Tools will accept any value for this field that does not exceed 2 bytes.

### idDataMinorId

: This is a beacon owner-specified value that, together with the idDataMajorId and the idDataUUID make the beacon unique. The default value is 0x324D, an arbitrary value. This value is also transmitted in AltBeacon packets.

: *Format and Validation:* The Beacon Tools will accept any value for this field that does not exceed 2 bytes.

### EddyStoneUID

A 10-byte Namespace ID combined with a 6-byte Instance ID as defined in the Eddystone Protocol Specification.

The default provided by the firmware is

```
namespace   0x699EBC80E19A0F0CF3EE
instance    0x050201EEF30E
```

### EddyStoneURL

A properly formed URL for use in Eddystone-URL (E-URL) packets.

The URL prefix must be one of four allowable formats:

```
http://www.
https://www.
http://
https://
```

Up to 17 bytes can follow the prefix. The default URL is https://www.emdeveloper.com.

### beaconPeriod

: The beaconPeriod is the amount of time, in milliseconds, from the start of one advertising cycle to the start of the next. The default value is 1000 ms (1 second). Per the bluetooth 4.0 specification The amount of time specified by the beacon Period is randomly extended by 0 to 10 milliseconds to prevent accidentally synchronized beacons from colliding continuously. Note that beaconPeriod only applies to the CUSTOM state machine (defined below). In the PRODUCT state machines, beaconPeriod is predefined for each state.

: *Format and Validation:* The Beacon Tools will accept values in units of milli-seconds between 30 mSec and 26 minutes (1,560,000 mSec), inclusive, in increments of 10 mSec, i.e., 30, 40, ..., 100, 110, ..., 1560000.

### outputPowerLevel

: Defines the power level (0..7) at which to transmit. The default value is 0x05 (-3 dBM). Note outputPowerLevel only applies to the CUSTOM state machine (defined below). In the PRODUCT state machines, outputPowerLevel is predefined for each state.

: *Format and Validation:* The Beacon Programming and Update Tools will accept any value from 0 to 7. Nominal power levels for EM9301 revs 07 and 10 are given in the following table. Production EMBC0x beacons have EM9301 revs 10 or higher.

| Value | Typical (dBm) | |
|---|---|---|
| | (07) | (10+) |
| 0 | -18 | -18 |
| 1 | -15 | -15 |
| 2 | -12 | -12 |
| 3 | -9 | -9 |
| 4 | -6 | -3 |
| 5 | -3 | 0 |
| 6 | 0 | +2 |
| 7 | +3 | +4 |

### beaconMode

: Specifies which packet types to enable. Any combination is valid. If more than one type is enabled, the beacon will advertise each packet type in round-robin fashion, one advertisement per beaconPeriod. If no packet types are enabled, the beacon will signal an error (both LEDs remain on) on startup.

: *Format:* Bit flags.

```
Value       Meaning
0x01        1=enable Sensor Data advertisements
0x02        1=enable enable ID Data (iBeacon compatible)
0x04        1=enable AltBeacon
0x08        1=enable Eddystone URL (E-URL)
```

```
     0x10       1=enable Eddystone Telemetry (E-TLM)
     0x20       1=enable Eddystone UID (E-UID)
```

**beaconMachines**

```
     Value    Meaning
     0x01     CUSTOM
     0x02     PRODUCT
     All other values are reserved and will result in a startup error (both LEDs remain on).
```

: *State Machine Definitions*

```
     CUSTOM
     1. Off (Red LED flashes once)
     2. On  (Green LED flashes once)

     In the CUSTOM state machine, pushing the button toggles advertisements on/off.
     The time between advertisements is specified by beaconPeriod, power is
     specified by outputPowerLevel, and packet types are specified by beaconMode.


     PRODUCT state machines (the factory default) are based on the beacon type as follows:

     EMBC01
     1. Off
     2. Proximity, outputPowerLevel 1, beaconPeriod  100 ms.
     3. Proximity, outputPowerLevel 3, beaconPeriod  500 ms.
     4. Proximity, outputPowerLevel 5, beaconPeriod 1000 ms.

     Where 'Proximity' = IDData + AltBeacon + E-UID

     EMBC02 (EMBC01 + Accelerometer)
     1. Off
     2. Sensor Data,            outputPowerLevel 4, beaconPeriod 1000 ms.
     3. IDData + E-UID + E-TLM, outputPowerLevel 3, beaconPeriod  100 ms.
     4. IDData + E-URL + E-TLM, outputPowerLevel 7, beaconPeriod  100 ms.

     EMBC03 (EMBC01 + Temperature Sensor)
     1. Off
     2. Sensor Data,            outputPowerLevel 5, beaconPeriod  500 ms.
     3. IDData + E-UID + E-TLM, outputPowerLevel 5, beaconPeriod  100 ms.
     4. IDData + E-URL + E-TLM, outputPowerLevel 5, beaconPeriod  100 ms.
```

: *Validation:*

The beacon programming and update tools allow any value.

**openSensor**

: Determines which values are displayed for the OpenSensor and OpenEventCounter fields in the EMBeacon packet. Up to 16 sources can be specified for each one. The upper 4-bits specify the content of the OpenEventCounter and the lower 4-bits the OpenSense field. The default value for EMBC01 is 0x01 (Button/Firmware Rev), for EMBC02 its 0xCB (Taps/Acceleration), and for EMBC03 its 0x34 (Low Temperature/Temperature).

```
     Upper Nibble    OpenEventCounter     Lower Nibble    OpenSense

         0           Button Press             0           Light (Lux)
         1           Low Battery              1           Firmware Rev
         2           One-Shot Cal Cycles      2           Autocal Constants
         3           Low  Temperature         3           Unspecified
         4           High Temperature         4           Temperature
         5           Low  Pressure            5           Reserved
         6           High Pressure            6           Reserved
         7           Low  Humidity            7           Reserved
         8           High Humidity            8           Reserved
         9           Close Magnet             9           Reserved
```

| A | Far Magnet | A | Reserved |
|---|---|---|---|
| B | Any Movement | B | Acceleration |
| C | Taps | C | Reserved |
| D | Falls | D | Reserved |
| E | Alarm | E | Reserved |
| F | Buzzer | F | Reserved |

: *Format and Validation:* The Beacon Tools will accept any value that can be represented as a single byte.

**temperatureOffset**

: Temperature offset provides for manual adjustment of the reported temperature for each beacon. The value, which defaults to zero, is added to each temperature reading generated by the beacon.

: *Format and Validation:* Internally, the temperatureOffset is maintained as a two-byte, signed, 2's complement fixed-point number with the binary point placed between the two bytes, giving it a range from -128.0 to +127.996. The Beacon Programming and Update tools will accept any two-byte hexadecimal value and any decimal value in the given range. When decimal values are given, truncation may occur internally to values that can be represented in binary form. NOTE that the format of the offset is the same as the format used for the temperature itself so large magnitudes could cause overflow.

## Parameters and Logging

The `program_beacon` and `update_beacon` scripts create log files in CSV format each time they're run. They list the final values of *all* parameters that are present in the updated device. In addition they include three values that are *not* part of the parameter block, *dateTimeProgrammed* (ISO 8601 format), *FirmwareVersion* and *EM6819SerialNumber* (a 7-byte Manufacturer tracking code stored in flash memory).