Bluetooth SMART Advertise-Only Beacon Reference Design

Features

- **COMPLETE BLE BEACON SYSTEM**
  - EM9301 – Bluetooth SMART Controller
  - EM6819 – 0.9V Flash Microcontroller
  - RF Front End – Chip or Folded Dipole Antenna
  - DC-DC Converter if Vbatt <1.8V
  - Sensors –
    - Ambient Light
    - Temperature
    - Battery Voltage
    - Button Press
- **Ultra-Low Power Consumption**
  - VDD from 0.9V to 3.6V
  - 155uJ per BLE Advertise Packet
  - 228uJ for VDD <1.8V
- **Small Form Factor**
- **emBeacon iPhone® App**

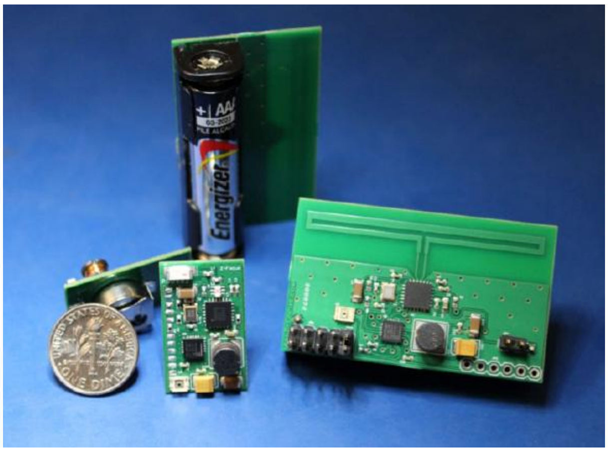

Figure 1: TINY and LOWCOST emBeacons

General Description

The **emBeacon** Reference Design is a complete Bluetooth Low Energy (BLE)/Bluetooth SMART reference design optimized for ultra-low power BLE Advertisement. **emBeacon** can last 3 years on an AAA alkaline battery, or operate continuously on harvested energy from typical indoor lighting. Silver-Oxide, Zinc-Air, and Lithium Ion coin cells can also be used. Sensor data, including ambient light level, temperature, and battery voltage is broadcast in the BLE Advertise Packet to all smartphones within the vicinity (up to 100 meters). With slight modifications, the emBeacon can collect and transmit other data such as humidity, motion or impact, etc…

The free **emBeacon** iPhone App receives emBeacon data from all advertisers in the area and presents it on the phone screen.

GPIO on the EM6819 power the sensors and the EM9301 radio, maximizing efficiency. Firmware on the EM6819 controls and measures the sensors, monitors power supply level, controls the DC-DC converter, and programs EM9301 radio parameters such as transmit power, beacon interval and advertise packet information via Host Control Interface (HCI) commands over an SPI interface.

A TINY **emBeacon** design (schematic, layout, bill-of-materials, and firmware) is provided for applications limited size constraints, and a LOWCOST emBeacon design is provided for lowest cost applications not limited by size.

Applications

- Remote Environmental Sensors
- Geolocation & RTLS (indoor mapping)
- Attraction Guides
  - Museums
  - Theme Parks
  - Shopping Malls
- Proximity Detectors ("find me")
- Facility Monitoring – Doors, Lights

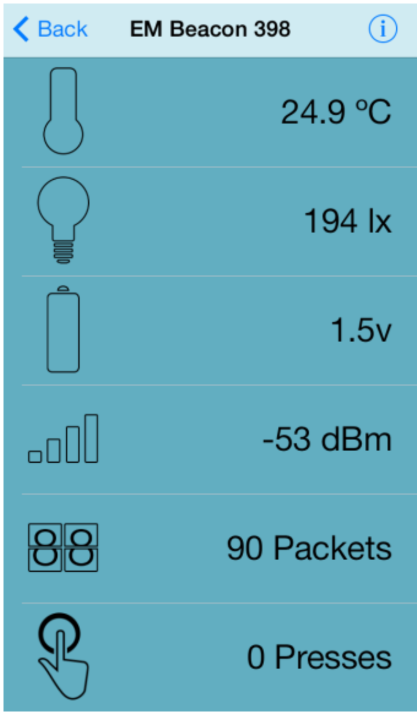

Figure 2: emBeacon iPhone App Screen

A photograph of the TINY and LOWCOST emBeacons are shown in Figure 1. The LOWCOST beacon has a folded dipole antenna, considered almost "free" since it is printed on the PCB and matches directly to the EM9301 antenna pins. This has been replaced by a balun and chip antenna on the TINY beacon for a slightly higher BOM, but much smaller board implementation. The firmware for the EM6819 is programmed through the 10-pin GASP connector as seen on the LOWCOST beacon board. The GASP connector is provided as a break-off for the TINY beacon and discarded after programming. A block diagram of both the TINY and LOWCOST emBeacons are shown in Figure 3. Both antenna options are shown connected to the EM9301. Additionally, a low cost 26MHz +/-50ppm crystal is required for BLE timing and frequency channelization. The EM9301, light sensor, and push button are powered through the EM6819 GPIO. Additionally, the HCI commands are issued over SPI also on EM6819 GPIO. A coil is required by the DC-DC converter on the EM6819 if the beacon is operating below 1.8V

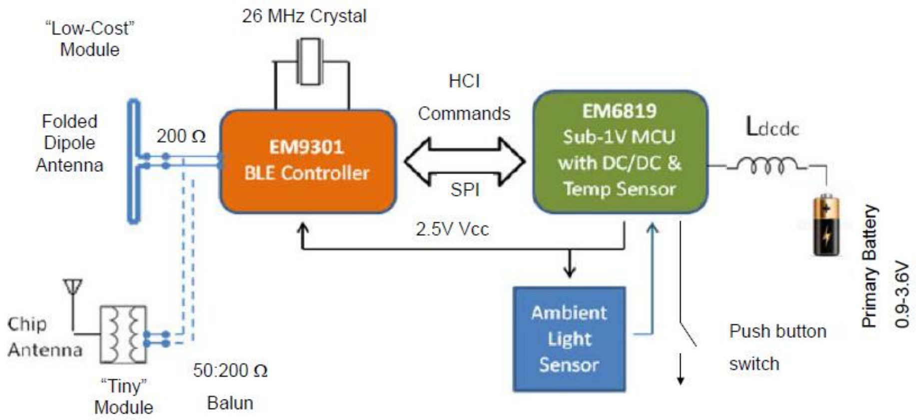

Figure 3: emBeacon Block Diagram

The emBeacon firmware has several adjustable parameters including transmit power, beacon interval, emBeacon number. Additionally, a second advertisement has been added to support the iBeacon™ format (UUID/Major/Minor ID). These values are flashed to the EM6819 when the device is programmed in production. By default the transmit power is set to 0 dBm which results in an indoor range of typically 50m and an outdoor range of over 100m. The beacon interval is typically set to 1 second for the LOWCOST emBeacon and 5 seconds for the TINY emBeacon. A longer interval is used for longer battery life, but this becomes more difficult for the scanning device to detect (iPhone, for example) and some packets could be missed. For simplicity, the three least significant digits of the serial number of the EM6819 are used for the emBeacon number and the five least significant digits are used for the Minor ID by default. This results in a large number of beacons without duplication.

Firmware Flowchart

A flowchart of the firmware is shown in Figure 4. On power up, the beacon first checks to see if the push button switch is pressed. If it is, only the iBeacon format data will be sent, and otherwise both the iBeacon and emBeacon data are sent. Next, the battery voltage is sampled by the EM6819 10-bit ADC through an internal GPIO. If the battery voltage is less than 0.9V, the EM6819 will go to sleep and periodically wake up to see if the voltage has increased to over 0.9V. This is useful for energy harvesting applications where a PV cell is charging a large hold capacitor, which may take a long time.

Once 0.9V on the battery pin has been achieved, the DC-DC converter is enabled and set to generate an output voltage of 2.5V. If the battery voltage is already over 2.2V (for example, a 3.0V battery is used), the DC-DC converter is not required. The battery voltage is also recorded for transmission in the packet.

If the emBeacon packet format is defined, on alternate packets the temperature sensor is sampled and then the light sensor is sampled. The temperature sensor is on the EM6819, uses the on-chip ADC, and is calibrated during chip production. The temperature sensor can be used to monitor the temperature inside a house or even used as an outdoor weather station. The light sensor is a Taos (now ams AG) TSL12T and has an analog output. The light sensor is first powered, sampled by the EM6819 ADC, and then powered off. The maximum output of the TSL12T is limited by the supply voltage and accurate reading of large amounts of light typically requires a high supply voltage (up to 5.5V). However, lower levels of light can be discerned. The light sensor can easily be used to tell whether a bedroom light was left on or not, for example.

The EM9301 is then powered on. The crystal oscillator is turned on to meet precise BLE timing requirements and for the frequency reference of the transceiver. The chip is then calibrated, and is ready for transmission. If the emBeacon packet format is defined, the advertise packet manufacturers address and data (battery voltage, light measurement, etc.) are downloaded from the EM6819 to the EM9301 using standard BLE HCI commands, and the advertising data is transmitted in compliance with Bluetooth Smart standard on the three BLE advertising channels using the EM9301 internal link layer. Then the iBeacon manufacturer's address and data (UUID/Major/Minor ID) are downloaded from the EM6819 to the EM9301 again using standard BLE HCI commands. After the advertisement is complete, the EM9301 is powered off.

The EM6819 is finally put in sleep mode for the remaining interval period for best-in-class system power consumption (1-2uA).

An additional interrupt service routine has been implemented to process the button press. A count indicates the number of times the button has been pressed since the battery has been inserted. This could be used to monitor a door, for example. Furthermore, the number of beacons since the battery has been inserted is also available and can be used for estimating the battery lifetime. (After a battery has discharged below 0.9V, the data is still in memory. A power supply can be used to power the device and resume beaconing in order to take a final reading before replacing the battery.)

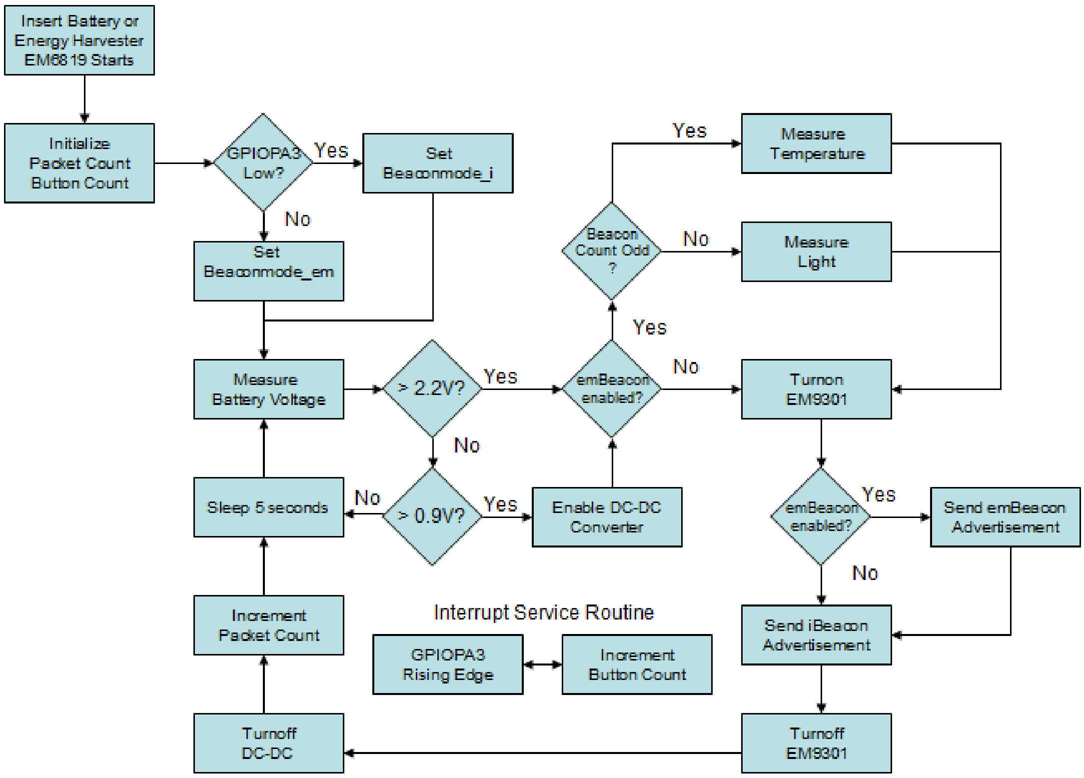

Figure 4: emBeacon Firmware Flowchart

Power Consumption Profile

The emBeacon/iBeacon power consumption measurement is shown in Figure 5 for a 2.5V battery voltage. The battery trace C2 is shown at the top of the oscilloscope display, with a 1V/div voltage resolution and 10ms/division time resolution. The current trace F1 is shown in yellow, with a 2mA/div resolution. The integrated current consumption per second is shown in trace F2 with 20uAs/div resolution. This trace multiplied by 2.5V results in uJ.

The first event at 21ms from the left of the display is the EM6819 waking up and measuring light, which consumes 7.5uJ. A sharp spike in the current results when the EM9301 is turned on at 30ms, but this is negligible energy corresponding to the charging of the decoupling caps. The next significant event is at 40ms, which is the EM9301 calibration procedure. This requires 92.5uJ. The next significant event at 58us is the emBeacon advertisement, which takes 62.5uJ. This advertisement includes turning on the RF, sending packets on the 3 BLE advertising channels, and returning to standby mode in-between each transmission. The three peaks corresponding to these transmissions are identifiable.

Total energy consumption up to this point is 162.5uJ. This would be the total consumption if only one packet were advertised. However, the second iBeacon packet is advertised, which consumes another 62.5uJ, bringing the total to 225uJ. (Note if only an iBeacon packet is desired, the total required energy is 155uJ.) Then the EM9301 is powered down and the EM6819 is put into sleep mode where the current is 2.2uA.

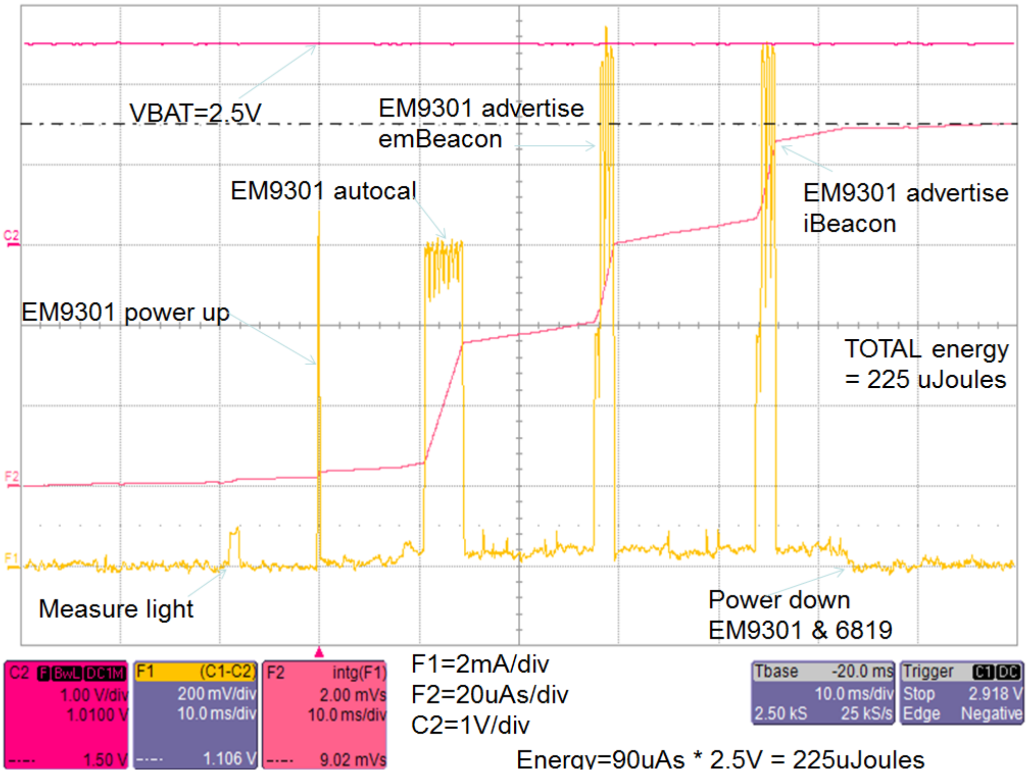

Figure 5: emBeacon Current/Energy Consumption at 2.5V

Similarly, the emBeacon powered at 1.2V, which uses the EM6819 DC-DC converter, is shown in Figure 6 with slightly different vertical scales to accommodate the higher consumption due to the DC-DC efficiency. In this case, the first event at 20ms from the left of the display is the enabling of the DC-DC converter and consumes 24uJ. The next event that consumes significant energy is the EM9301 calibration, which consumes 120uJ when using the DC-DC converter. The two transmissions then consume 84uJ each, for a total of 228uJ for one packet, and 312uJ for two packets. Then the EM9301 is powered down and the EM6819 is put into sleep mode where the current is 1.3uA.

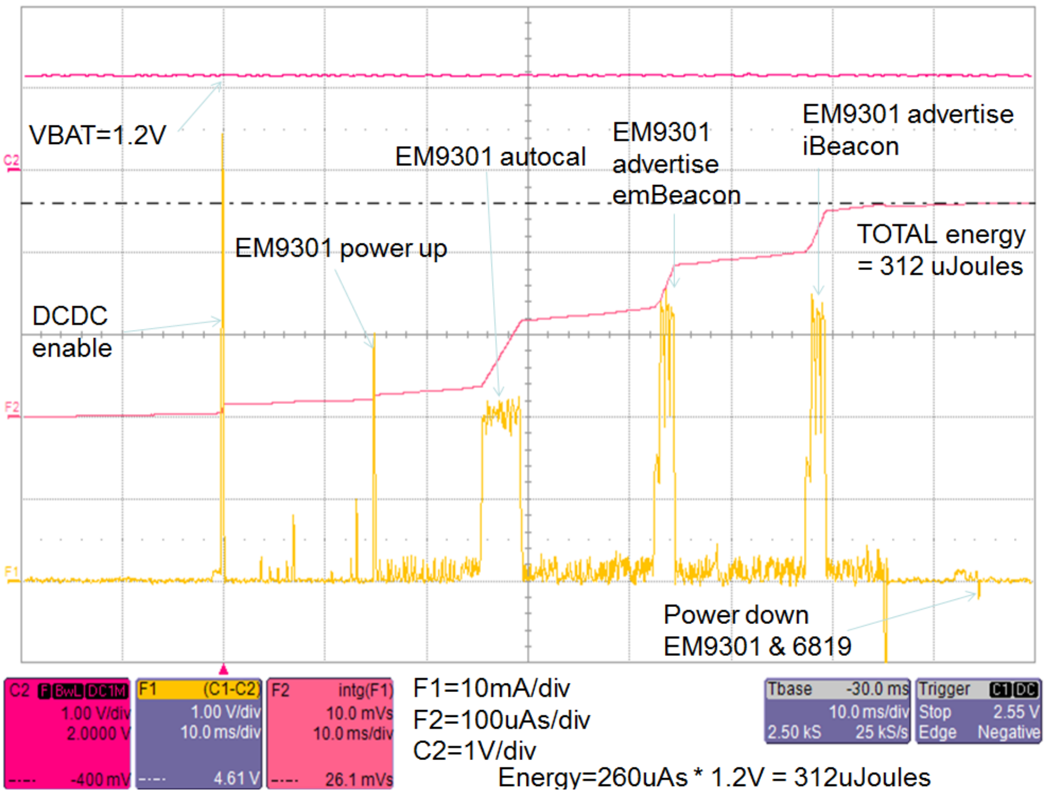

Figure 6: emBeacon Current/Energy Consumption at 1.2V

Packet Format

The emBeacon PDU format is shown in the Figure 7 below and is 37 bytes long. The Advertise Address is in the first few columns and consists of the 3 byte public address of EM Microelectronic (LSB first). The second 3 bytes are uniquely determined from the lower 3 bytes of the EM6819 serial number. The rest of the columns comprise the Advertise Data. The Advertise Data consists of the Local Name, Manufacturers ID and Data, each preceded by the length in bytes. For the Local Name, we put the ASCII text "EM Beacon XYZ", where XYZ is the last 3 digits of the EM 6819 serial number. In the example below XYZ = "035". The Manufacturers ID is filled with the EM Microelectronics ID from the Bluetooth SIG (90 = 0x005A). This is followed by the light sensor reading in Lux from 0-32000, temperature in Celsius (integer and decimal parts), and battery voltage where the 4MSBs are integer and the 4LSBs are the decimal part. For example, 2.8V = 0x28). This is followed by a 32-bit integer for the number of packets transmitted (MSB first), and then a 16-bit integer for the number of switch events or buttons pressed.

AdvAddress						AdvData																														
EM specific adv address (public), from IEEE			beacon specific address from 6819 serial number lower 3 bytes			number of bytes to/including end of complete local name	complete local name (18.3)	E	M		B	e	a	c	o	n		unique beacon number derived from 6819 serial number (035)			end of complete local name	number of user data bytes to follow	manufactirer specific data, company identifier code	Maufacturer ID (low)	Maufacturer ID (high)	ALS (high)	ALS (low)	Temp (int)	Temp (dec)	Battery (V)	number of packets transmitted	number of packets transmitted	number of packets transmitted	number of packets transmitted	number of switch events (MSB)	number of switch events (LSB)
0C	F3	EE	S2	S1	S0	0F	09	45	4D	20	42	65	61	63	6F	6E	20	30	33	35	00	0E	FF	5A	00	LL	LL	TT	TT	BB	PP	PP	PP	PP	SW	SW

Figure 7: emBeacon Packet Format

The PDU for the iBeacon format is shown in the Figure 8 and is 36 bytes long. It starts with the same EM public advertising address. Then the advertising data is comprised of the predefined iBeacon prefix, the UUID, the Major ID, and the Minor ID. The last byte represents the calibrated TX power used for range estimation. The default UUID used by the emBeacon is one commonly used for development purposes. The major ID is set to 90, the EM BT SIG number. And the Minor ID is set to the LSBs of the EM6819 serial number.

AdvAddress						AdvData																													
EM specific adv address (public), from IEEE			beacon specific address from 6819 serial number lower 3 bytes			Apple's fixed iBeacon advertising prefix									iBeacon profile UUID																Major ID		Minor ID		2's compliment of the calibrated TX power
0C	F3	EE	S2	S1	S0	02	01	1A	1A	FF	4C	00	02	15	E2	C5	6D	B5	DF	FB	48	D2	B0	60	D0	F5	A7	10	96	E0	MJ	MJ	MN	MN	C5

Figure 8: iBeacon Packet Format

Battery Lifetime Estimates

The battery lifetime has been estimated for the emBeacon for several different battery options and beacon configurations. The beacon intervals of 1 or 5 seconds have been used in combination with the energy dissipation measurements for the 1 or 2 beacon configurations (iBeacon, emBeacon, or both). Average power is then calculated, and this is used with the average voltage of the battery and rated mAh for the battery to estimate the expected lifetime of the battery. Two charts are shown below. The chart in Figure 9 is for several 1.5V batteries, and the chart in Figure 10 is for 3.0V battery configurations.

In Figure 9 lifetimes for several Alkaline batteries are estimated including the AAA battery used in the low-cost beacon, and the LR44 battery used in the tiny beacon. The low-cost beacon will last approximately 1/2 year with a 1 second interval sending out both packets, but it is increased to 3 years when beaconing the iBeacon packet a 5 second interval. The tiny module with an LR44 will typically last about 4 months with a 5 second beacon interval. However, the tiny module with a ZA675 battery (same size) lasts almost two years in the same configuration. Some smaller batteries were considered that could beacon for several months. The ultra-thin Blue Spark 101-UT1 printed battery will beacon for up to 1 month with 5 second intervals.

1.5V Battery Lifetime Estimates		Interval (s)	1	1	1	5	5	5
		Beacons	iBeacon	emBeacon	Both	iBeacon	emBeacon	Both
		E (uJ)	228	228	312	228	228	312
		P (uWavg)	228	228	312	46	46	62
Battery	Vavg	mAH	Years	Years	Years	Years	Years	Years
AA	1.25	2000	1.3	1.3	0.9	6.3	6.3	4.6
AAA (Low-Cost)	1.25	1000	0.6	0.6	**0.5**	3.1	3.1	2.3
AAAA	1.25	500	0.3	0.3	0.2	1.6	1.6	1.1
LR44 (Tiny)	1.3	115	0.1	0.1	0.1	0.4	0.4	**0.3**
ZA675	1.4	650	0.5	0.5	0.3	2.3	2.3	1.7
			Weeks	Weeks	Weeks	Weeks	Weeks	Weeks
SO-365	1.55	40	1.6	1.6	1.2	8.1	8.1	5.9
ZA5	1.4	33	1.2	1.2	0.9	6.0	6.0	4.4
Blue Spark 101-UT1	1.4	20	0.7	0.7	0.5	3.7	3.7	2.7

Figure 9: emBeacon Lifetime Estimates for 1.5V Batteries

The emBeacon is even more efficient in a 3V configuration, as shown in Figure 10. Two AAA batteries last almost 2 years as an iBeacon with a 1 second interval, or over 9 years with a 5 second interval. The popular CR2032 coin-cell battery lasts for almost 2 years beaconing both packets with a 5 second interval. Several button cell batteries are also estimated with lifetimes of 1-8 months.

3V Battery Lifetime Estimates		Interval (s)	1	1	1	5	5	5
		Beacons	iBeacon	emBeacon	Both	iBeacon	emBeacon	Both
		E (uJ)	155	163	225	155	163	225
		P (uWavg)	155	163	225	31	33	45
Battery	Vavg	mAH	Years	Years	Years	Years	Years	Years
2xAA	2.5	2000	3.7	3.5	2.5	18.4	17.6	12.7
2xAAA	2.5	1000	1.8	1.8	1.3	9.2	8.8	6.3
2xAAAA	2.5	500	0.9	0.9	0.6	4.6	4.4	3.2
CR2450	2.9	540	1.2	1.1	0.8	5.8	5.5	4.0
CR2032	2.9	225	0.5	0.5	0.3	2.4	2.3	1.7
			Weeks	Weeks	Weeks	Weeks	Weeks	Weeks
CR2016	2.9	90	10.0	9.6	6.9	50.1	47.8	34.5
CR1220	2.9	40	4.5	4.2	3.1	22.3	21.2	15.3

Figure 10: emBeacon Lifetime Estimates for 3V Batteries

iOS App

The emBeacon uses a free iOS app available on the App Store for iPhones® (4S and greater) and iPads®. This app was created by KS Technologies for EM. The screenshot of the app in Figure 11 is made from the iPad app. Five beacons are shown in range in a column on the left side of the screen, with beacon numbers and RSSI. Any beacon on the list can be selected to reveal more real-time data. In this case, “EM Beacon 398” is selected, and the temperature, light, battery, and RSSI data are shown, as well as the number of packets that the beacon has sent out and number of button presses. A screen shot of the iPhone app is shown in Figure 2.

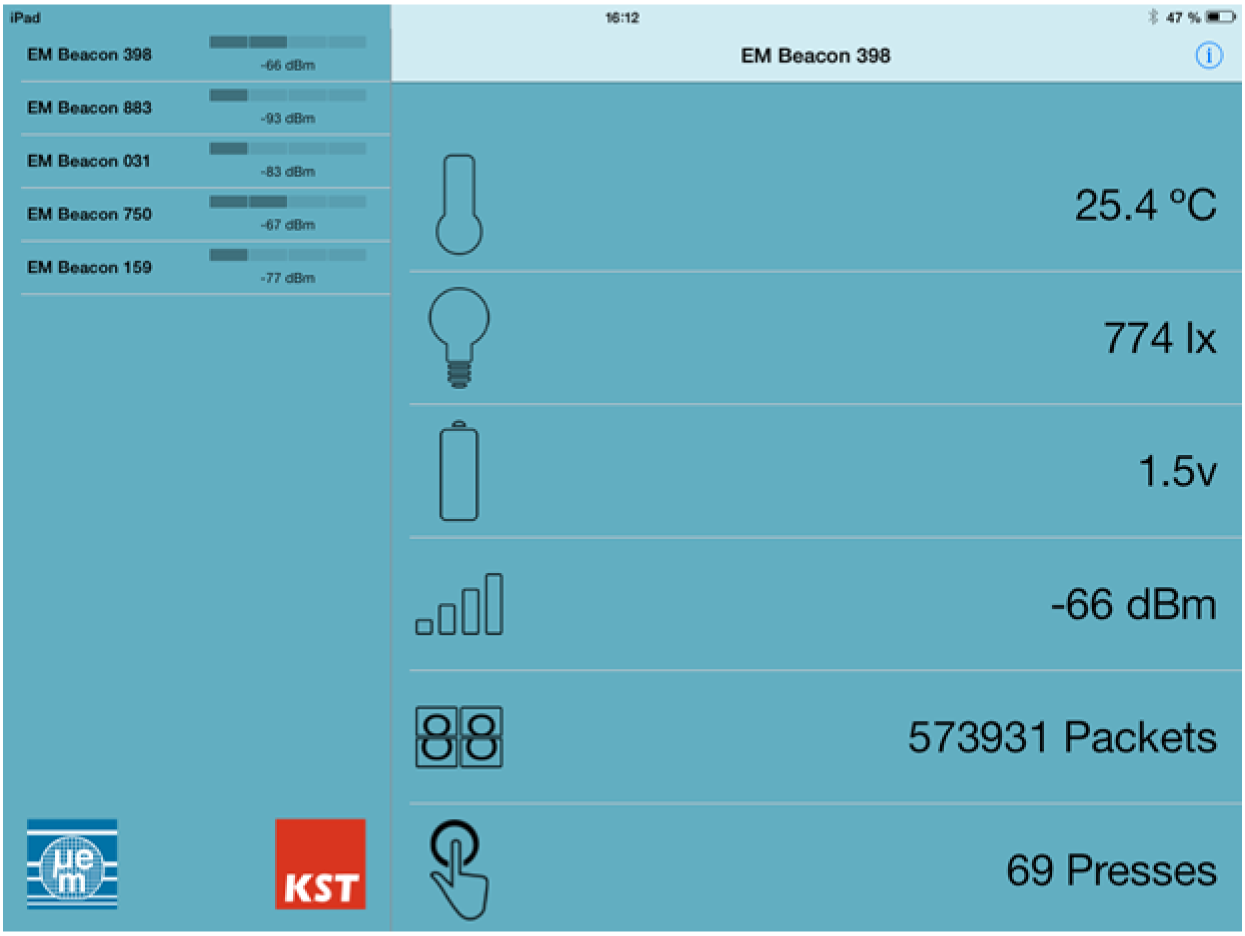

Figure 11: emBeacon iPad App

The EM and KST icons have hot links to the respective company websites. The "i" circle at the upper right corner of the app is a hot-link to any pre-programmed website. For example, at the EH&C conference the beacon with powered by the energy harvested solution provider provided a hot-link to that companies web-site. The link can easily be changed by EM by modifying a text file on the Internet.

The iBeacon formatted packet is detectable by using the KS Technology "Particle Detector" also available for free from the App Store. The UUID of the beacons defaults to E2C56DB5-DFFB-48D2-B060-D0F5A71096E0 which is typically used for development purposes. This has to be selected at the bottom of the app. Then all visible beacons will be displayed with Major and Minor ID's, and an estimated range based on RSSI of the iPhone or iPad.