

# Pocket PC Battery Lifetime Planning with BatteryTimer

## Introduction

We developed [BatteryTimer](#) to help us answer questions from clients regarding the feasibility of using Pocket PC type devices in a variety of scenarios. One of the critical make-or-break questions was always 'will the battery last long enough for a typical day's usage?' - since most mobile users have limited opportunities to recharge their device during the working day.

Of course, typical usage varies from project to project, sometimes involving constant network access using a Wireless LAN card, and in others only involving 10 minutes use every hour or so throughout the day as a salesman makes his rounds. Add to this the complication of the various power saving options available (backlight settings, inactivity shutdown, network card shutdown etc.) and it quickly becomes quite difficult to say with any conviction whether a particular device is suitable, or a given scenario is even viable.

BatteryTimer allows approximation to a pattern of usage whilst logging battery charge in the main and supplemental batteries to a text file. The usage pattern can simulate constant use, in which case BatteryTimer logs data until the main battery charge reaches approximately 10%, or an 'On for X minutes off for Y minutes' pattern. By fixing the powersaving settings to their least economical settings (e.g. backlight always on, no automatic power off etc.) you can estimate a worst case scenario for a given pattern. By then repeating the tests with a variety of usage patterns you can build an idea of the possible working life between charges of the battery.

## Scenario

In this article we're going to examine a hypothetical scenario consisting of the proposed use of a Compaq iPaq 3630 running a data collection application for mobile use in a hospital ward. The device is required to be available for use immediately at all times during the working day (9am to 5pm).

The finished application will use SQL Server CE on the Pocket PC, connecting over HTTP to a backend SQL Server. Ideally the users would like updates to reach the back end database more or less immediately, and to this end a wireless LAN link is being considered, but of equal importance is the fact that the Pocket PC will spend most of the day in someone's pocket, hence recharging opportunities may be limited.

If the wireless LAN option is not viable, the users may be prepared to make do with a periodic database sync via returning the Pocket PC to it's cradle and using ActiveSync to provide the TCP/IP connection.

We will use BatteryTimer to obtain battery lifetime measurements for a range of scenarios as follows.

<u>Test</u>	<u>On</u>	<u>Off</u>	<u>Power Save Mode</u>	<u>Backlight</u>	<u>PCMCIA Jacket</u>	<u>WLAN Card</u>
1	Always	Never	Always On	Medium, Always On	No	No
2	Always	Never	Always On	Medium, Always On	Yes	No
3	10 mins	1 hour	Always On	Medium, Always On	Yes	No
4	5 mins	10 mins	Always On	Medium, Always On	Yes	No
5	Always	Never	Always On	Medium, Always On	Yes	Yes

## Results

BatteryTimer produces csv (comma separated variable) text files in a format suitable for immediate use in spreadsheet programs such as Microsoft Excel. The following charts are produced in Excel and chart the Main and Jacket battery percentage charge remaining on the Y axis, against elapsed time on the X axis.

Notes:

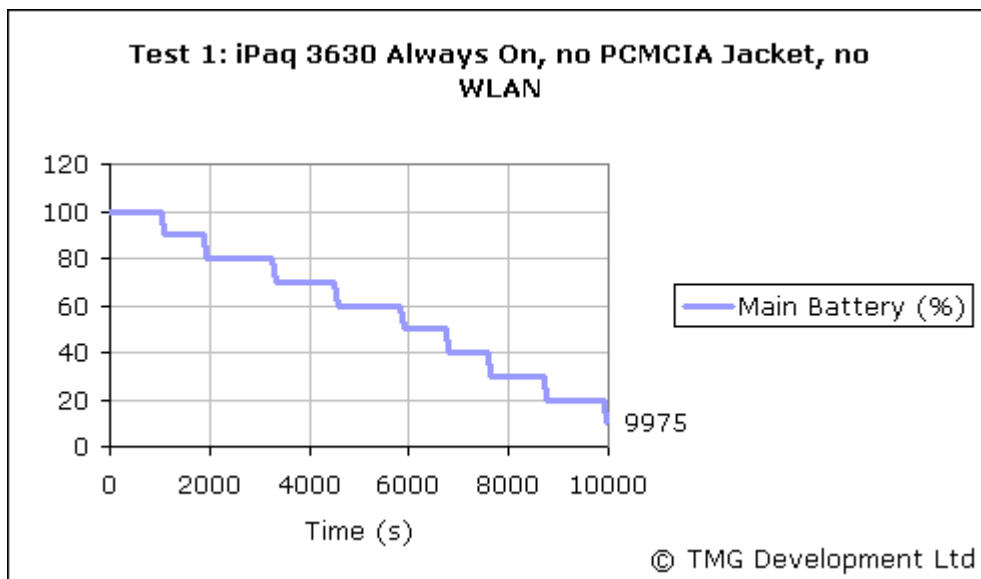
1. For each test the BatteryTimer settings are documented. All tests were performed using BatteryTimer version 1.1.0.7 on an iPaq Pocket PC 3630 running Pocket PC 2002 OS. The wireless LAN card used in these tests is an SMC2632W card using the PRISM 802.11 Wireless LAN Driver.

2. Each test was deemed to be complete when the Pocket PC was effectively no longer usable by a user using it in the manner of the scenario. This means that for 'Always On' scenarios the power could get as low as 10% since the device would be in constant use and the user could dismiss the low power warnings. For on-off scenarios the run might typically stop when the charge was at approximately 40% in the main battery because at this point Windows CE puts up a message along the lines of "To prevent possible data loss, replace or recharge your battery according to the owner's manual", and subsequently refuses to honour instructions to wake up at a future time. This disables the mechanism that BatteryTimer uses to simulate on-off usage, and the run ends.

### Test 1: Always on, no PCMCIA jacket, no WLAN card

BatteryTimer settings: on for 'Always', off for 'Never', whilst on log every 1 minute.

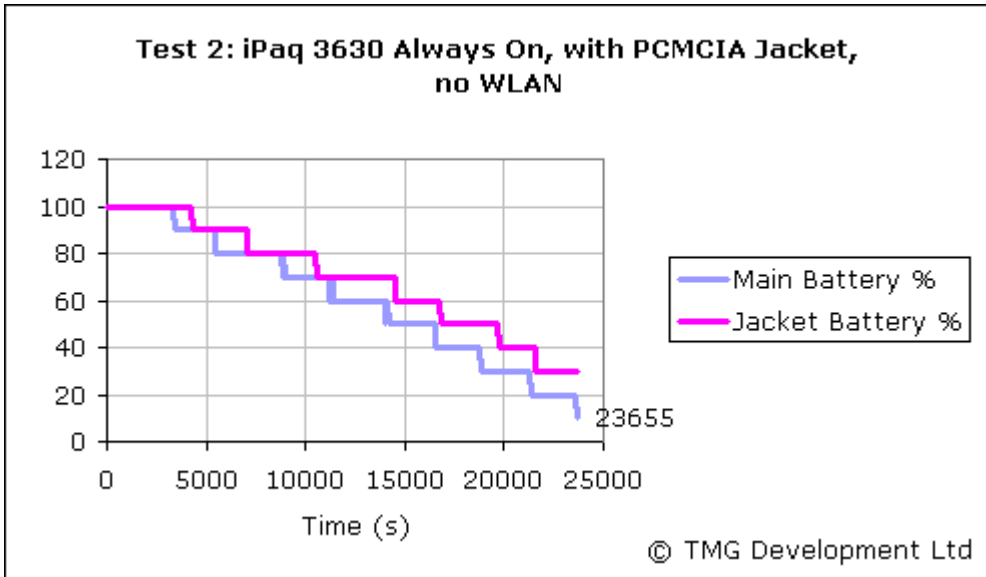
Total runtime: **2h 46m 15s**



### Test 2: Always on, PCMCIA jacket, no WLAN card

BatteryTimer settings: on for 'Always', off for 'Never', whilst on log every 1 minute.

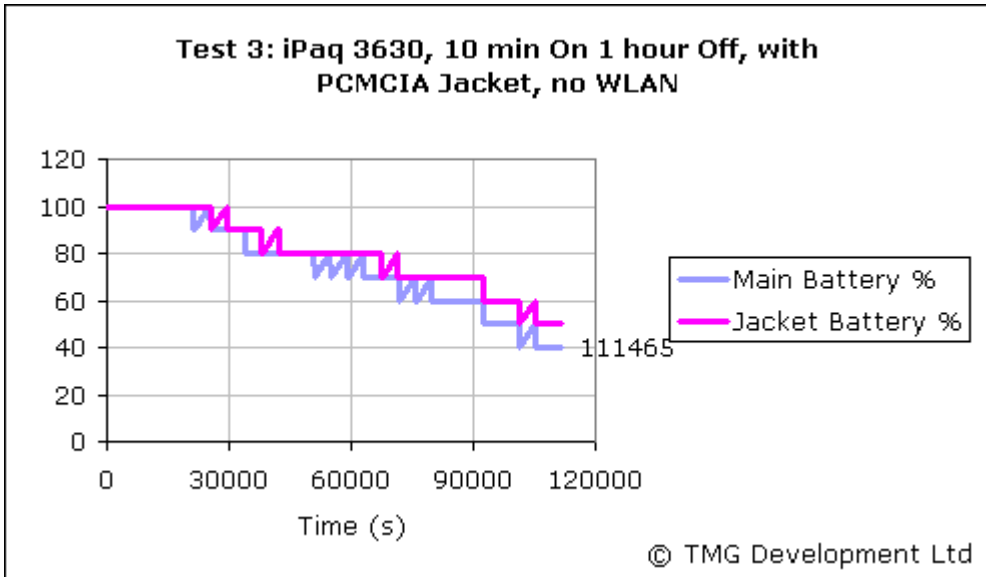
Total runtime: **6h 34m 15s**



**Test 3: 10 min On, 1 hour off, PCMCIA jacket, no WLAN card**

BatteryTimer settings: on for '10 min', off for '1 hour', whilst on log every 1 minute.

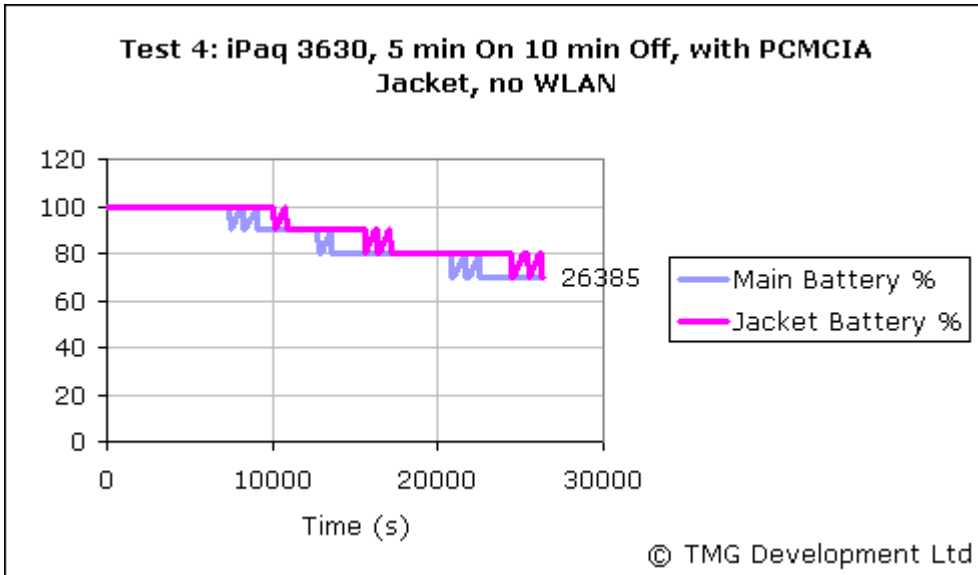
Total runtime: **1d 6h 57m 45s**



**Test 4: 5 min On, 10 min Off, PCMCIA jacket, no WLAN card**

BatteryTimer settings: on for '5 min', off for '10 min', whilst on log every 1 minute.

Total runtime: **7h 19m 45s**



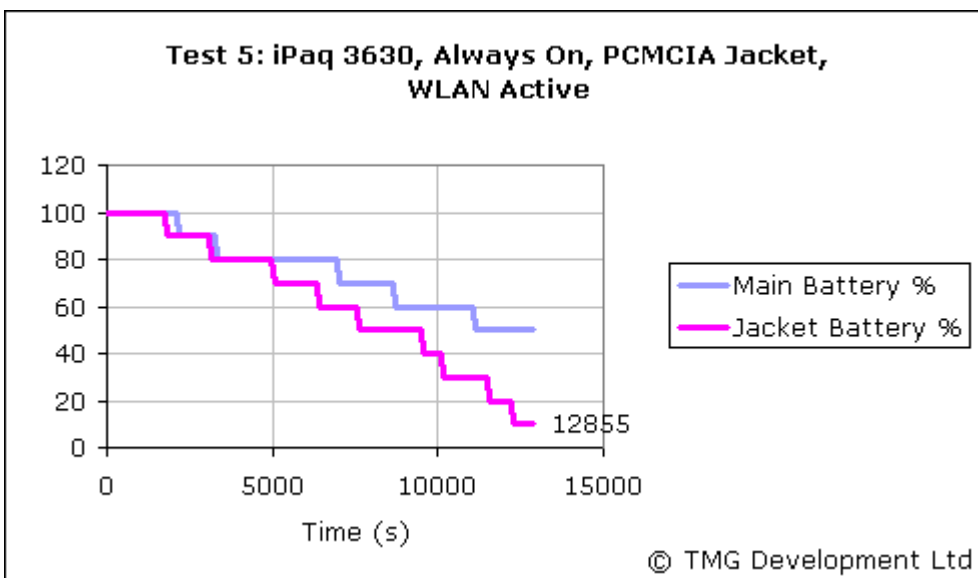
**Test 5: Always on, PCMCIA jacket, WLAN card Active**

BatteryTimer settings: on for 'Always', off for 'Never', whilst on log every 1 minute.

Additionally, to ensure that the Wireless LAN card did not go into powersave mode during the test run, pocket IE was opened on a page on a remote webserver containing an HTTP-EQUIV="Refresh" tag pointing at itself. By this mechanism pocket IE forces the network card to remain active by requesting a refresh of the page every five seconds. The web page was called reload.htm and contained the following:

```
<html>
<head>
<meta HTTP-EQUIV="Refresh" CONTENT="5; URL=http://192.168.1.20/reload.htm">
</head>
</html>
```

Total runtime: **3h 34m 15s**



**Conclusion**

These tests represent the worst case in each scenario - a typical user would enable the backlight and shutdown power saving options, and in the wireless LAN test the WLAN card would not normally be transmitting data in quite the constant way that we engineered, which would result in further power savings.

Even with power saving modes enabled, the wireless LAN requirement of our users looks like it would require two devices in alternate use. Two devices each giving over 3.5 hours of continuous use (Test 5) would more or less cover the entire working day. With powersaving modes enabled the batteries could last considerably longer, though further tests would be required to fully quantify this.

Many users complain that having a PCMCIA jacket attached makes the device too bulky, but Test 1 shows that the option of running standalone (ie no PCMCIA with backup battery and no WLAN) is not really practicable, as the built-in battery gives well under 3 hours life. Further tests would be required to determine possible extensions in battery life that could be made by running with no backlight and having automatic powersave enabled.

The hybrid scenario of returning the device to its cradle for data transfer (and a small amount of charging) would definitely be feasible with a PCMCIA jacket attached - even with no powersaving options enabled Test 2 shows that we get over 6.5 hours continuous use. Test 3 approximates the scenario of 10 minutes use every hour or so, and in this case we get over a day between recharges.

Once we have decided which arrangement to implement we could perform an in-situ set of tests where the device is trialed in the working environment while we have BatteryTimer running in the background (in Always on mode). As the device is switched on and off, returned to the cradle and then removed again, we would capture a log of the status of the batteries throughout the day. This data would enable us to develop guidelines about how long, and how frequently, the device should be returned to the cradle for recharging in order to ensure continuous availability.

---

BatteryTimer is produced by [TMG Development Ltd](http://www.tmgdevelopment.co.uk). For more information or to download a trial version please visit the BatteryTimer website at <http://www.tmgdevelopment.co.uk/batterytimer>.