Demo Abstract: *iAssist* – Rapid Deployment and Maintenance of Tiny Sensing Systems

Matthias Keller, Guido Hungerbuehler,
Oliver Knecht, Suhel Sheikh, and Jan Beutel
Computer Engineering and Networks Lab,
ETH Zurich, Switzerland

Stefanie Gubler, Joel Fiddes,
and Stephan Gruber
Department of Geography,
University of Zurich, Switzerland

matthias.keller@tik.ee.ethz.ch

Abstract

Commercial, coin-sized iButton temperature logger devices are well-suited for densely instrumenting large outdoor areas. An efficient workflow for deploying and maintaining those devices is necessary when striving to deploy and operate several hundreds of data logger devices. Additionally, a sophisticated data management is required for handling the emerging, large amounts of meta and measurement data. Therefore, we developed *iAssist*, a solution that integrates the handling of iButton data logger devices together with a GPS receiver and a digital camera for gathering accurate location information. *iAssist* efficiently supports the whole workflow consisting of deploying, relocating and reading tiny sensing systems. *iAssist* is especially tailored for outdoor operation asking for as little user interaction as possible.

1 Introduction

Dense instrumentations for environmental monitoring ask for cheap and easily installable sensing systems. For instance, a Maxim iButton® DS1923 [1] is a coin-sized, commercial device that integrates a micro-controller, 8 kB storage, a real-time clock, a temperature sensor, and a battery in a single package (see Figure 1). The lifetime of this low-power data logger is mainly determined by its configurable sampling interval, the device's data sheet lists more than 7 years of operation when hourly sampling data. An iButton can be programmed and read out by connecting the device to a PC by using a 1-wire interface. Without limiting the application area of these devices and also our presented management tool, this paper focuses on the deployment of iButton temperature loggers for densely instrumenting high-alpine locations. The overall goal of this project is to obtain data for geophysical research.

Within a first campaign, we have installed several hundreds of iButton devices in the Swiss Alps. At the beginning of a campaign, each device must be initially setup by uploading a set of mission parameters such as the sampling interval, the starting time of the measurement and the desired measurement resolution. As the data loggers are installed outdoors within rough terrains, it is crucial to have accurate information about the deployment location. Here, we use both assisted GPS as well as pictures of the location. This information is used to relocate the iButton devices during the campaign when data is sporadically read out.

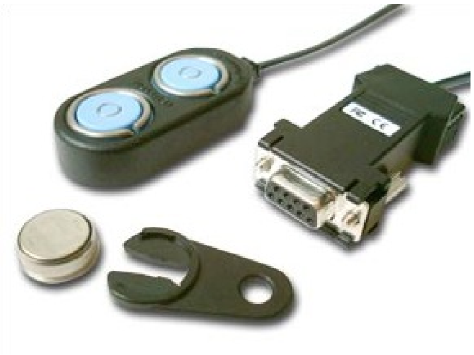


Figure 1. Coin-sized iButton device and 1-wire device programmer. The maximum diameter of an iButton device is 17.35 mm, the maximum height is 5.89 mm.

It becomes apparent, that running a campaign with hundreds of devices firstly asks for as much as automation as possible, and secondly also generates a huge amount of data that must be handled properly. In this paper, we present the *iAssist* management tool for the deployment, the localization and the maintenance of tiny iButton data loggers. Concretely, *iAssist* integrates an iButton programmer interface, a digital camera, and a GPS receiver into a single application. All data is stored in a relational DBMS, the current version of the software runs on an Intel Atom netbook running Linux.

2 Operation Concept

iAssist is tailored for the rapid deployment and maintenance of tiny iButton data loggers in the field. Each iButton device comes with a unique device identifier that is read when an iButton is inserted into the device programmer hardware.

The overall area covered by a measurement campaign can span many square kilometers. While it is not possible to fully cover such large areas, the measurement area is divided into several patches of dense instrumentation. Deploying sensors starts with registering a patch and configuring the mission parameters that are identical for a batch of devices. This

Copyright is held by the author/owner(s).
SenSys'10, November 3–5, 2010, Zurich, Switzerland.
ACM 978-1-4503-0344-6/10/11

allows to quickly program many devices without any further user interaction but only insertion and removal of devices from the device programmer hardware. GPS coordinates are automatically recorded by the software, optional pictures can be taken within the software by pressing a single button.

For relocating iButton devices, the connected GPS receiver is used for guiding the user to the location of deployed data loggers. Here, distance and direction to a particular device are given. This is sufficient since the user has to move within rough terrain without fixed routes. Additionally, a user can also access captured photos by employing the integrated picture viewer that is part of the software. When a deployed iButton is inserted into the device programmer hardware, all data is immediately read out. Reading out data loggers is also done in batches and can include the immediate reprogramming of a data logger after the stored data has been read out.

3 System Architecture

In this section, we give a brief overview of the most important aspects of our solution.

3.1 Time Synchronization

Several hundreds of iButton devices are deployed to gather timestamped temperature measurements. For a higher data quality, it is desirable that all data loggers periodically sample data at the same time instant. Firstly, the real-time clock of each data logger is synchronized with the PC when the iButton device is programmed. In turn, the local clock of the PC is synchronized by GPS or external NTP servers. Secondly, all data loggers are programmed for a delayed start of the measurement at a pre-defined time instant, i. e. the beginning of the next day. Thirdly, the timestamp of the last sampling point is compared with the time on the synchronized clock of the PC when data is read out. This information is used to (linearly) account for the drift of the real-time clock of the data logger.

3.2 Data Model

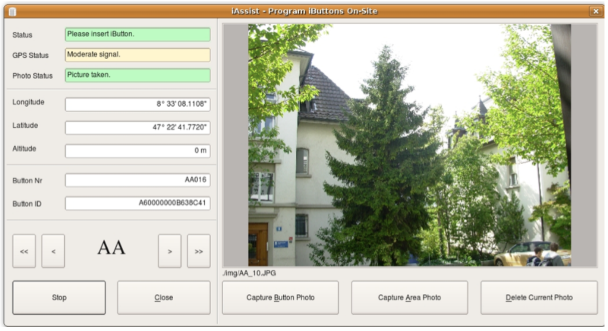
In general, all data is annotated with the unique device identifier of the corresponding iButton device. Stored meta-information includes the mission parameters of each deployed data logger, the time when a data logger was programmed, the time when a data logger was read out, information about available pictures and detailed location information. Additionally, measurement data consisting of a timestamp and a temperature measurement are stored for each sampling point. While pictures are stored in the file system, all other information are stored in a relational DBMS.

3.3 Software Components

iAssist is implemented in C++ using the Qt4 GUI toolkit [4]. The SQLite [5] database engine is used for storing data. Gphoto2 [2] is used for interfacing a digital camera that is connected to a USB port, location and time from a GPS receiver are accessed via gpsd [3].

4 Demo Setup

Our demo setup consists of the system that we actually use in the field. Concretely, this includes an Asus EeePC netbook, an external u-blox GPS receiver, a digital camera and the iButton device programmer. Additionally, our setup includes a bunch of Maxim DS1923 iButton devices that can be programmed and read out during the demonstration. We will demonstrate the whole workflow within the *iAssist* software plus visualizations of the measured data. While we sample with lower rates in the field, the demo becomes very interactive when setting the sampling interval of a logger to the smallest value of 1 second. For instance, we will encourage visitors of our demo to measure their (approximate) body temperature by holding an iButton for a while.

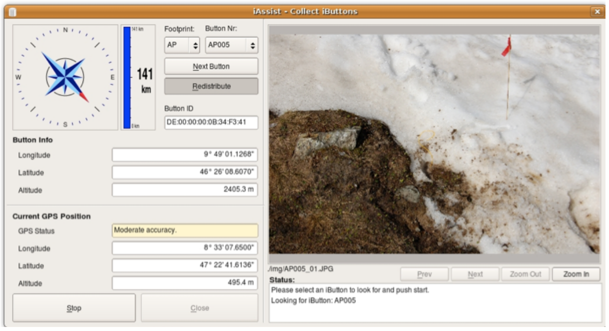


iAssist - Program iButtons On-Site

Status	Please insert iButton.
GPS Status	Moderate signal.
Photo Status	Picture taken.
Longitude	8° 33' 08.1108"
Latitude	47° 22' 41.7720"
Altitude	0 m
Button Nr	AA016
Button ID	A60000000B638C41

./img/AA_10.JPG

Figure 2. Screenshot of *iAssist* software showing the user interface for deploying a batch of tiny sensing systems. The interface contains status information (left) and a dialog for adding optional pictures about a location (right).



iAssist - Collect iButtons

Footprint: AP
Button Nr: AP005
Button ID: DE:00:00:00:0B:34:F3:41

Button Info

Longitude	9° 49' 01.1268"
Latitude	46° 26' 08.6070"
Altitude	2405.3 m

Current GPS Position

GPS Status	Moderate accuracy.
Longitude	8° 33' 07.6500"
Latitude	47° 22' 41.6136"
Altitude	495.4 m

./img/AP005_01.JPG

Status:
Please select an iButton to look for and push start.
Looking for iButton: AP005

Figure 3. Screenshot of *iAssist* software showing the user interface for locating a deployed data logger. A wind rose (left) and a picture (right) help the user to find the installed device.

5 Acknowledgments

The work presented was supported by NCCR-MICS, a center supported by the Swiss National Science Foundation under grant number 5005-67322 as well as by the project CRYOSUB supported by the Swiss National Science Foundation under grant number 200021-121868.

6 References

[1] DS1923 Product Page. http://www.maxim-ic.com/datasheet/index.mvp/id/4379.

[2] gPhoto2. http://gphoto.sourceforge.net/.

[3] gpsd. http://gpsd.berlios.de/.

[4] Qt UI framework. http://qt.nokia.com/.

[5] SQLite. http://www.sqlite.org/.