



Provided by the author(s) and University of Galway in accordance with publisher policies. Please cite the published version when available.

Title	Connectivity solutions to link a bluetooth camera to the internet
Author(s)	Ionas, Adrian; Corcoran, Peter M.
Publication Date	2001
Publication Information	P. Bigioi, A. Ionas, G. Susanu, P. Corcoran (2001) "Connectivity solutions to link a bluetooth camera to the internet", IEEE Transactions on Consumer Electronics, Vol. 47, No. 3, pp. 294-299.
Publisher	IEEE
Item record	http://hdl.handle.net/10379/281

Downloaded 2024-03-20T11:16:08Z

Some rights reserved. For more information, please see the item record link above.



CONNECTIVITY SOLUTION TO LINK A BLUETOOTH CAMERA TO THE INTERNET

Petronel Bigioi¹, Adrian Ionas², George Susanu¹ and Peter Corcoran²

¹FotoNation Ireland Ltd. Galway, Ireland

²Dept. of Electronic Engineering, National University of Ireland, Galway

Abstract

Digital photography is gaining day by day more and more credibility over conventional photography due to a number of factors easy to guess. Bluetooth connectivity in the digital camera provides cordless communication with PC, printer and phone resulting in better usage of digital cameras. More over, with the adequate software, the digital photographs could go directly from camera to desired target: disk, printer, web sites, e-mail or web prints.

1. Introduction

Bluetooth wireless technology is a low-cost, low-power, short-range radio link for mobile devices and for WAN/LAN access points. It offers fast and reliable digital transmissions of both voice and data over the globally available 2.4 GHz ISM (Industrial, Scientific and Medical) band. The Bluetooth wireless technology comprises hardware, software and interoperability requirements. It has been adopted not only by all major players in the telecom, computer and home entertainment industry, but also in such diverse areas as the automotive industry and health care, automation and toys, etc. - almost all sectors of the economy. Bluetooth chipset vendors have done a lot of work but still integrating bluetooth with a target application is a very complex task, involving sometime a full redesign of the application, making the adoption of bluetooth difficult.

In this paper a complete bluetooth connectivity system, software, hardware and firmware is presented. The system is aimed for easy integration with the target application, digital camera. The whole system, Figure 1, comprises of a bluetooth enabled prototype digital camera, bluetooth enabled PC, firmware and software. Moreover, the PC side software is able to detect a bluetooth enabled digital camera, download the pictures, store, print locally or upload for print or sharing on photographic sites.

Conventional downloading mechanism incurs the existence of a cable, RS232, USB or 1394. This system provides an alternative wireless means of interconnecting

electronic equipment with a minimal requirement for additional user interaction, keeping the conventional cable connection methods. It provides easy picture transfer between the digital camera and Internet through a standard PC.

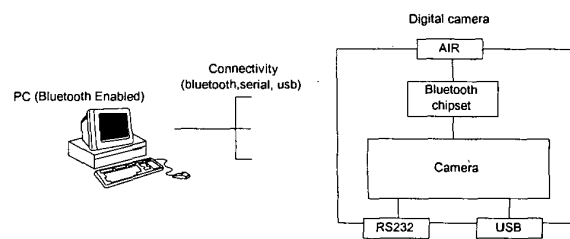


Figure 1: Connectivity system

Security and data encryption for the wireless connection are inherent component of the bluetooth architecture upon which this system relies. A development system comprising of a standard PC, two bluetooth modules and a prototype digital camera has been used. Additional bluetooth modules have been used to prove the interpretability and no radio interference between the test modules and any other bluetooth hardware that might operate in the same coverage area.

2. Achievements

The major goal behind this product lies in the implementation of a complete bluetooth solution that enables wireless picture transfer between a digital camera and a standard PC. The ability of bluetooth protocol to replace the physical serial connection with a virtual one, made possible to design a camera communication protocol that works in the same manner with direct cable or a bluetooth link. The difference between the two is made at the PC end by specifying the transport type. Moreover, the camera communication protocol is transport independent, easily usable for other types of transport such as USB or 1394.

The device discovery and name discovery, built in the bluetooth architecture, enabled the PC to auto discover the digital camera. The bluetooth address and name discovery provided a mean to uniquely identify the digital camera and the protocol it runs and load the appropriate device drivers and application software.

The image processing module, included in the application software at the PC end, together with home/internet printing and uploading/sharing capabilities make the system a very powerful tool for end customers. Traditionally, before a digital picture reached its final destination (i.e. printing, sharing, processing and storing) had to go through at least three of four different applications. Unless special demands are rising, with the proposed system an 'all in one' meaning is provided.

3. System overview

The presented system provides an easy mean of inter-devices connectivity over bluetooth links, taking advantage of existing hardware and software infrastructure. In particular, a complete connectivity kit targeted for digital cameras is provided.

The idea of providing such a kit originates from the lack of digital image manipulation from both digital cameras and scanners. Today, a digital image, from source to destination takes a very long a complicated path. In example, if professional prints are wanted out of a digital image located in a digital camera, several actions need to be taken: to make the cable connection with PC, to start the acquisition software, to start image processing software, to start the uploading software and finally, to start an internet browser to place the print order). All those operations make the usability of a digital camera a nightmare for the end user.

The overall workflow and digital imaging devices usability would be allot improved if the user wouldn't need to connect the device to PC and moreover, automatically, a software application that would include all the above mentioned functionality, would automatically start.

In this section, we describe the system architecture, hardware, software and firmware and its integration with the PCs operating system.

The system comprises of following components:

- PC software comprise of several components: low-level transport device drivers, transport abstraction layer, digital camera protocol implementation, image processing and filtering layer, image

management, including printing local storing and internet sharing, storing or printing.

- Bluetooth modules, at both PC end and digital camera end.
- Digital camera firmware, including the communication protocol, OS adaptation layer and transport adaptation layer.

4. System description

4.1 PC software

In this section, we describe the software required to manage the system, divided in two parts: bluetooth network management service and user application. A functional diagram of the PC software is given in Figure 2.

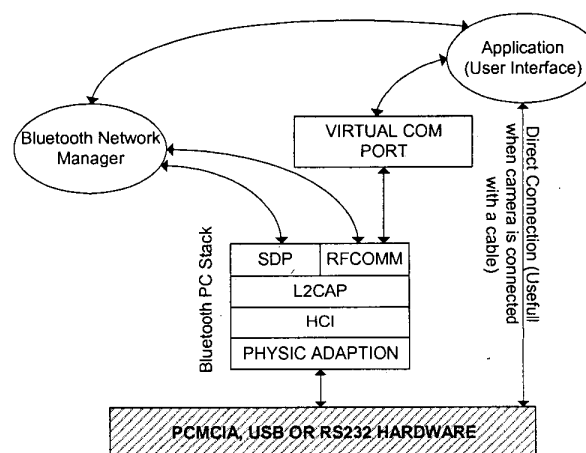


Figure 2 PC software

The interaction between the application and the Bluetooth stack is done through a virtual serial port, that exports towards the application an exact software interface as a physical serial port.

More intimate interaction between the bluetooth stack and our software is done at the bluetooth network manager level, where we have interaction with SDP (Service Discovery Protocol) layer and HCI (Host Controller Interface) layer.

To keep the compatibility with cameras that aren't bluetooth enabled, the PC software is able to communicate directly with those cameras. This is possible by exporting towards the application a generic camera access API.

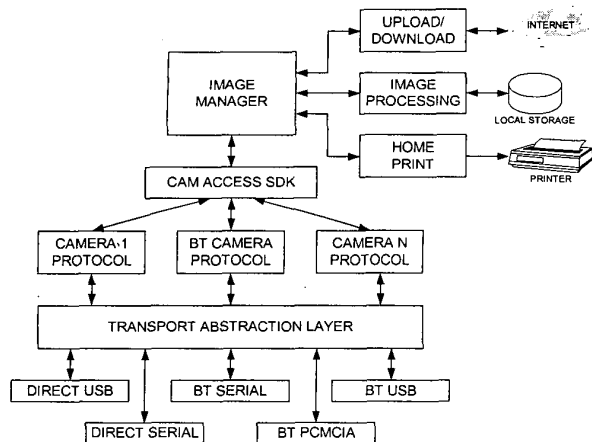


Figure 3 User application

4.1.1 Bluetooth Network Manager

This component runs as a background service on the PC, performing repeated device inquiry, at programmed intervals. Once a new bluetooth device is found, a further name discovery is performed. Once the device is identified and it appears to be imaging device, then the user is signaled and the application component could be started.

The Figure 4, Figure 5, Figure 6 and Figure 7 show the bluetooth network manager operation: scan without success, scanning for new cameras, new camera found and offering options for action.

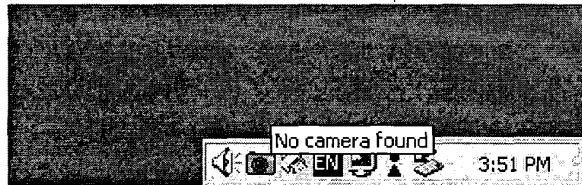


Figure 4: BT Manager (1)

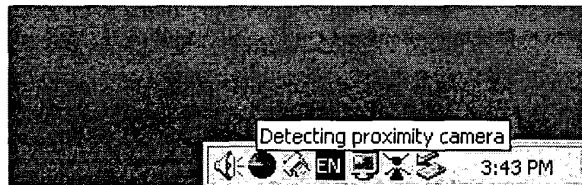


Figure 5: BT Manager (2)

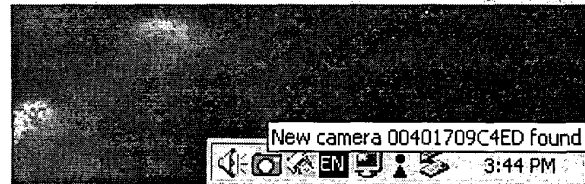


Figure 6: BT Manager (3)

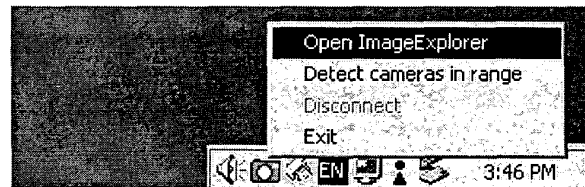


Figure 7: BT Manager (4)

4.1.2 User Application

This software component is the main piece of the PC end software. It is designed as a collection of sub software components, incorporating the user interface, image processing, support for multiple camera protocols, home printing and Internet connectivity.

A detailed view of the sub-components and of the interaction between them is given in Figure 3. Some features supported by the application are:

- Multiple transport support (bluetooth, direct serial cable, direct USB cable);
- Multiple camera support (virtually, any camera that has a known protocol can be supported by writing a very thin layer that implements the camera protocol);
- Generic access to camera protocol, through a dedicated camera access API (this layer exports towards the application a set of generic camera access functions);
- Image processing features (unique algorithm for red eye reduction, unique algorithm for auto picture fix, crop, loose-less rotation, etc.);
- Direct home printing, including pre-printing image processing;
- Direct Internet connectivity, both upload and download to a standard sharing photo server using a customisable API (this module can be used in standard mode or secure mode);
- Easy to use user interface and picture management (this interface exports towards the user all the above functionality, in a very simple and concise manner);

Upon request, other functionality could be added into the user application, such as direct web printing, more image processing functions, skin support, etc.

The software element that glues the other together is the 'IMAGE MANAGER', which is basically a GUI element, making use of the all software sub-components. Most of the components are designed as COM objects, with independent functionality and a very well defined interface with outside world.

Lets try to summarize and highlight the functionality of every component of the user application:

- Image manager is the graphical user interface, which pulls together all the other software components of the system, exporting to the end user all the features of the system;
- Cam access SDK is mainly a set of easy to use access camera functions, exporting the same behavior for each individual digital camera;
- Camera protocol layer comprise of a set of protocols, one per supported digital appliance. This is the only place were, if a new camera is released, new code to support it occurs;
- The transport abstraction layer exports to the camera protocol a set of functions, always the same, independent of the used transport. The link between the type of the used transport and the application is made at this level;
- Upload/download module is a software component that provides direct web connectivity for client side applications;
- Home print module provides the flexibility to print digital images to the local printer, providing functions such as print preview, layout organization, pre-print image processing and region printing.
- Image processing module comprise of in-house developed image processing techniques, such as looseless rotation, auto picture fix, a unique red eye reduction technique and cropping. Additional features could be added upon request.

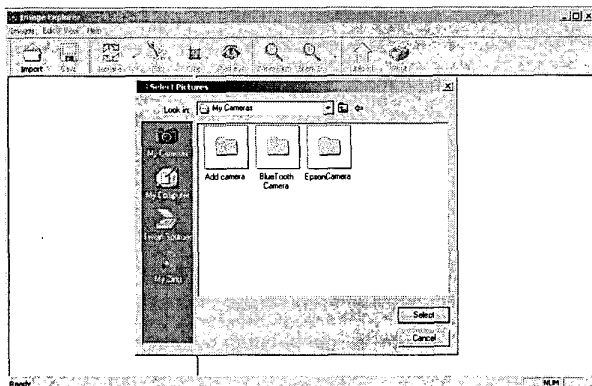


Figure 8: Application - Select Picture Source

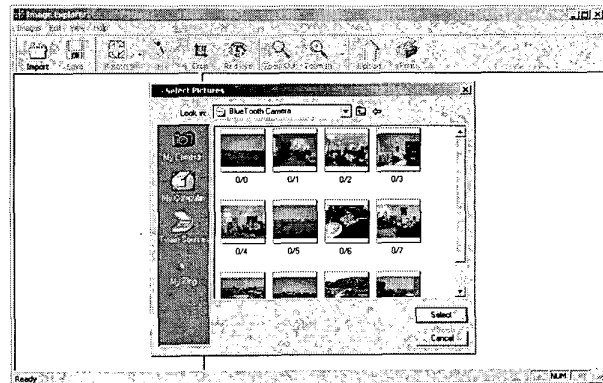


Figure 9: Application - Select Pictures from Bluetooth Camera

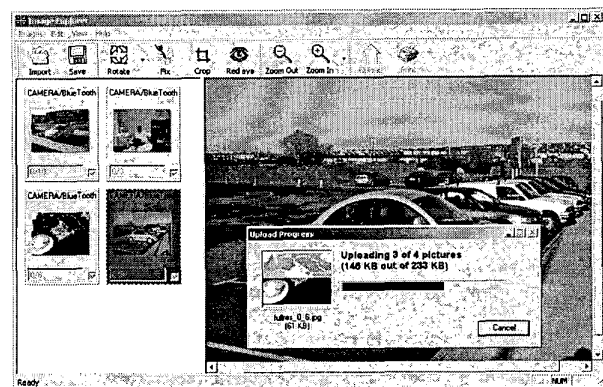


Figure 10: Application - Upload Pictures

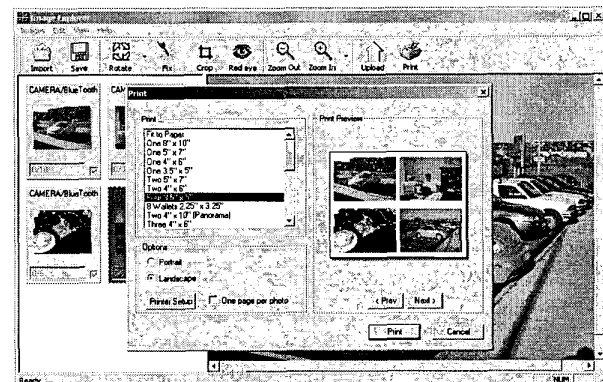


Figure 11: Application - Home Printing

4.2 Bluetooth modules

The camera end bluetooth module runs both the low-level bluetooth stack and the high-level bluetooth protocol stack. The Figure 12 shows exactly the organization and interaction of the bluetooth protocols.

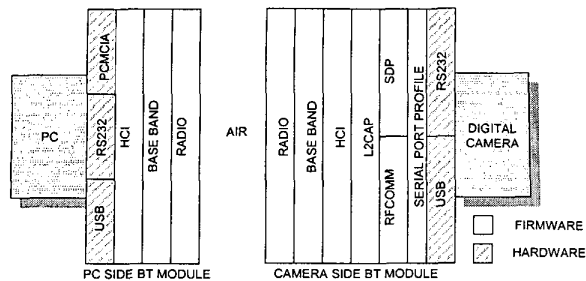


Figure 12 Bluetooth firmware

The bluetooth stack protocols at the PC end are distributed between the embedded bluetooth module and the PC. This organization is required for access to specific features of the high-level bluetooth protocol stack, such as device discovery, name discovery, etc.

5. Digital camera firmware

Of course, only the firmware concerned with the transfer protocol will be described. Otherwise, the digital camera incorporates an impressive amount of firmware, which is out of the context of this paper.

The interaction of the transfer protocol with the digital camera resources, such as storage and operation system is shown in Figure 13.

The components of the digital camera firmware are:

- Communication protocol is the core protocol, packet based, transport independent;
- Storage access module is exporting towards the communication module standard file access/search functions used by the core module to find, access and delete images on the camera's file system;
- OS access module exports generic operating system functions, such as memory allocation functions, basic timers, display, etc. The adaptation between the camera's OS and core module is done here;
- Transport abstraction module exports transport generic access functions towards the communication protocol module and make the link with the transport specific functions available in the serial driver module or USB driver module;

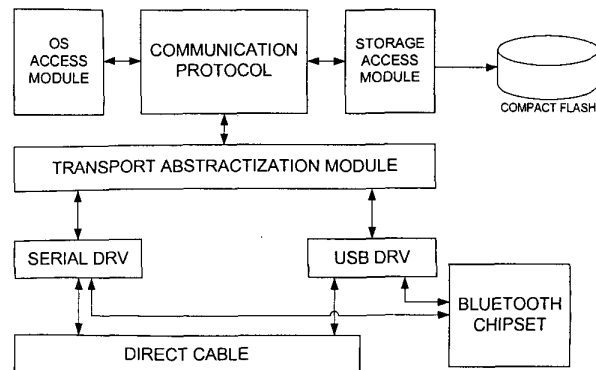


Figure 13: Digital camera firmware

6. Conclusions

The current system is targeted as a communication SDK for camera manufacturers. The most expensive components are the bluetooth chipsets, but the evolution of bluetooth market lately indicates a clear tendency for usability in consumer industry, so critical factor, such as price, will be overcome.

Better approaches could have been used, such as building a dedicated profile for imaging devices, but we have preferred to use the most known bluetooth profile (serial port profile) as a working base for our work. The simplicity of the serial port profile assures the compatibility of our solution within different bluetooth hardware manufacturers.

7. References

- [1] Bluetooth official site
<http://www.bluetooth.com>
- [2] Digianswer A/S: "Bluetooth application note"
<http://www.digianswer.com>
- [3] Troy XCD
<http://troyxcd.com>
- [4] Brent A. Miller, Chatschik Bisdikian "Bluetooth Revealed", Prentice Hall, 2001
- [5] Jennifer Bray, Charles F. Sturman "Bluetooth Connect Without Cables", Prentice Hall, 2001

8. Bibliographies



Petronel Bigioi received his B.S. degree in Electronic Engineering from "Transilvania" University from Brasov, Romania, in 1997. At the same university he received in 1998 M.S. degree in Electronic Design Automation. He received a M.S.

degree in electronic engineering at National University of Ireland, Galway in 2000. Currently he is working as R&D Engineer with FotoNation Ireland, a Zing Networks subsidiary. His research interests include VLSI design, communication network protocols and embedded systems.



Peter Corcoran received the BAI (Electronic Engineering) and BA (Math's) degrees from Trinity College Dublin in 1984. He continued his studies at TCD and was awarded a Ph.D. in Electronic Engineering for research work in the field of Dielectric Liquids. In 1986 he was appointed to a

lectureship in Electronic Engineering at UCG. His research interests include microprocessor applications, environmental monitoring technologies. He is a member of I.E.E.E.



Adrian Ionas received his BS in computer science from "Lucian Blaga University". Sibiu, Romania, in 1997. His interests include Object Oriented Programming, Object Oriented Design and Object Oriented Analyze. With a strong

background in windows architecture and technologies (ActiveX, COM, DCOM) he is currently working as programming team leader with FotoNation Ireland.



George Susanu received his BS degree in microelectronics from "Kishinev Polytechnic Institute", Kishinev, Republic of Moldova. With an experience of eight years in RTOS and embedded systems, having a wide experience in C/C++ programming he currently is working as R&D engineer with

FotoNation Ireland Ltd. His research areas include real time operating systems and device connectivity.