Chapter 16

FORENSIC ANALYSIS OF MOBILE PHONE INTERNAL MEMORY

Svein Willassen

Abstract Modern mobile phones store data in SIM cards, internal memory and external flash memory. With advanced functionality such as multimedia messaging becoming common, increasing amounts of information are now stored in internal memory. However, the forensic analysis of internal memory, including the recovery of deleted items, has been largely ignored. This paper presents two methods for imaging the internal memory of mobile phones. The methods are applied on several popular models to recover information, including deleted text messages.

Keywords: Digital evidence, mobile phones, internal memory

1. Introduction

Mobile phones have become the primary tool for personal communication and they frequently contain information that may have evidentiary value. It is, therefore, vital to develop forensically sound methods for extracting and analyzing digital evidence from mobile phones.

This paper presents two methods for imaging the internal memory of mobile phones. The methods are applied on several popular models to recover important evidence, including deleted text messages.

2. Mobile Phone Memory

Mobile phones are digital media; therefore, in principle, they have the same evidentiary possibilities as other digital media, e.g., hard drives. Deleted information can be extracted from a mobile phone in much the same way as it is obtained from a hard drive. Like other digital media, mobile phone memory is fragile and is easily deleted or overwritten.

Moreover, since a mobile phone is a complex, compact device, great care should be taken while attempting to extract evidence.

Considerable information is stored in mobile phones [19]. In addition to telecommunications-related information, modern phones contain images, sound files, multimedia messages, WAP/web browser history, email, calendar items and contact lists. SMS (short message service) text messages, which often contain useful information, are stored on the transmitting and receiving phones. Indeed, recovering deleted SMS messages was the main motivation of this research.

The following subsections describe the principal components of mobile phones that store information of evidentiary value.

2.1 Subscriber Identity Module

The advent of digital mobile telephony in the 1990s created a need for local storage in mobile phones. GSM, the most popular digital mobile phone system, mandates a SIM (Subscriber Identity Module) to be present inside each GSM device. A SIM card incorporates a processor and EEPROM memory. The SIM architecture is also used in at least one 3G system (USIM in UMTS).

SIM cards contain subscriber information and encryption keys for secure communications; they also store contact lists and text messages. Much of this information, including deleted text messages, can be recovered depending on the mobile phone model [19].

2.2 Internal/External Memory

Rigorous SIM specifications have prevented SIM cards from being used as general purpose memory storage devices. Therefore, as manufacturers implemented new functionality that required additional storage, mobile phones were equipped with internal memory, e.g., for storing missed and received calls, calendar events, text messages and contacts.

The first models used serial EEPROM chips as internal memory. However, the use of mobile phones as cameras and music players has led manufacturers to add external flash memory (e.g., SD, MMC, CF cards). External memory cards can be analyzed using commonly available tools [3, 4, 20].

3. Mobile Phone Architecture

Figure 1 presents the basic mobile phone architecture. The CPU performs all computational tasks, including controlling the communications circuits. The CPU uses the RAM for temporary storage during phone

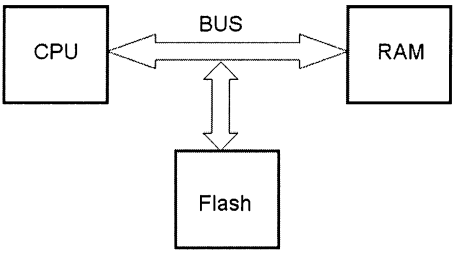

Figure 1. Mobile phone architecture.

operation. The RAM is a separate integrated circuit (IC) or it may be packaged with the CPU in a single IC.

Mobile phones also have secondary non-volatile storage for user and communications data that persist after they are powered down. Most commonly, secondary storage is implemented as separate flash memory integrated in the phone.

The CPU also communicates with the SIM card and additional external storage media, if present. Often, a special unit is present to control power usage, especially by the wireless transceiver.

4. Internal Memory Analysis

Standard methods do not exist for analyzing internal memory. Currently, information is extracted via AT commands using cable, infrared or bluetooth connections to a phone. GSM specifies a standard command set for extracting certain data [1]. Model- and manufacturer-specific data may be extracted using proprietary commands. Several software packages are available for this purpose, e.g., Oxygen Phone Manager, Paraben Cell Seizure and TULP2G [13, 14, 16].

AT commands use the phone's operating system to extract information. A major limitation is that deleted information may not be obtainable using these commands.

In some cases, potentially important evidence is deleted when a phone detects the presence of a new SIM card on start-up. Also, a phone might require a PIN code for it to be used with a specific SIM card. Therefore, if a phone is on when it is seized, it is suggested that it be kept on until the analysis is complete.

A jamming device or Faraday cage is recommended to ensure that a phone will not communicate with the network, and possibly have its memory modified. However, as discussed later, this may not prevent memory contamination. Moreover, if a phone is prevented from receiv-

ing signals, it will continuously probe the network for connectivity, consuming more energy and ultimately exhausting its battery.

Another method is to analyze a phone using a "cloned" SIM card. But this still involves turning the phone on and reading its memory using the operating system.

As long as a phone is on, it is possible for deleted information in its memory to be overwritten. Therefore, it is necessary to develop forensic methods for analyzing "dead" mobile phones. These methods would enable investigators to turn off mobile phones upon being seized, while ensuring that the internal states of the devices are maintained.

The following sections describe two methods for analyzing "dead" mobile phones: forensic desoldering that removes and images memory ICs, and imaging memory using a built-in test methodology. A third method is also possible – loading software into system RAM to read the flash memory through the system interface. But this method requires software specific to each mobile phone, and is not discussed in this paper.

5. Forensic Desoldering Method

The first method proposed for imaging the internal memory of mobile phones is to desolder the memory circuits and read the data off the chip.

5.1 Ball Grid Array Technology

Ball Grid Array Technology (BGA) is used to mount SMT (Surface Mount Technology) components on printed circuit boards (PCBs). Each chip has an array of pads on its lower side on which small balls of solder are placed. A chip is mounted by placing it on matching pads on the PCB and using reflow soldering [11] to bond the melted balls to the PCB pads. BGA technology uses less space on the PCB. Conventional ICs are limited not by chip area, but by the area for pins. BGA enables the entire area on the bottom of a chip to be used for signaling.

BGA presents difficulties to the forensic investigator. With conventional surface mounting, probes can be attached to the pins to read a chip. Thus, an investigator may access the memory contents without destroying the unit. However, BGA bonds the chip to the PCB. Individual probes cannot be attached to the chip, and the chip must be desoldered from the PCB before it can be read.

Desoldering must be performed with great care to avoid damaging the memory circuits. Melted solder residue must be removed so it does not short circuit the chip pads. Then, the chip is restored to its original state using a "reballing process." Alternatively, its contents are read by connecting directly to the pads.

5.2 Reballing

Many device readers require a chip package to be intact with its solder balls in place. Reballing restores the solder balls on the pads on the package's lower side. A special reballing stencil is used; this stencil has small holes that match the size and placement of the pads. The stencil is aligned so that each hole matches a pad. Solder paste is then applied on the stencil and evenly distributed until all the holes are properly filled. Finally, reflow is applied, which causes the solder paste to form balls that are fixed to the chip pads.

BGA reballing requires that the stencil match the ball size, inter-ball distance (pitch) and ball configuration of the chip. Since mobile phones have packages with different sizes and configurations, obtaining the right stencil can be quite difficult.

5.3 Reading Memory

A device programmer may be used to read memory circuits. A variety of adapters are available for device programmers, e.g., adapters for DIP or SOIC packages with up to 40 leads, or adapters for Intel 28F640 chips. In addition, the correct software is needed to read a chip as the pin configurations can vary. Fortunately, most manufacturers supply software with their device programmers that enables a variety of devices to be read. Many devices have built-in checksum capabilities to detect inconsistencies during the reading process; this capability must be supported by the software.

A device programmer may be used to read BGA circuits when conducting a forensic analysis of mobile phone memory. The adapter pins must match the ball size, pitch and layout. Adapters use either Y-shaped springs or spring-loaded pogo pins to establish chip connections. A Y-shaped pin must have a ball in place to establish a connection. A pogo pin can be used directly on the chip pads without balls. From the forensic point of view, spring-loaded pogo pins are better because reballing is avoided.

5.4 Desoldering

Equipment used for desoldering ranges from simple soldering irons to massive reflow ovens. The most important consideration from the forensic point of view is potential damage to the unit. An IC cannot endure more than a certain number of heatings to reflow temperature. Intel BGA circuits are guaranteed to endure reflow three times [7, 8]. Since a PCB with SMT components on both sides is assembled using two

reflows, only one heating to reflow temperature is available in a forensic investigation. This is not enough if both desoldering and reballing must be performed. However, experience indicates that components will survive more than the specified maximum number of reflows.

Proper temperature profiles must be maintained so as not to damage the unit. Since conventional mounting technology is not used, the entire chip must be heated. The maximum temperature should be nominally above the solder reflow temperature, and the gradient should not be too steep. Specifically, the temperature should have a "tent" profile with a peak of 210°C, which is reached in 3-4 minutes [10]. Pre-baking is recommended to remove moisture in the unit. Otherwise, heating may cause a "popcorn-effect," whereby water vapor causes the circuit substrate to crack. Moisture sensitivity and conditioning requirements are provided in [17].

These requirements rule out the use of hand soldering tools. Forensic desoldering should only be performed using an automatic soldering station with temperature gradient programming and automatic component removal capabilities. Such a soldering station typically employs a combination of convection heating using hot air and infrared (IR) radiation heating. It has a temperature-controlled hot air nozzle and can be programmed to follow specific heating curves.

5.5 Experimental Results

Several different phone models were dismantled, desoldered and imaged. The results are summarized in this subsection.

The mobile phone units were dismantled before desoldering. Manufacturers often provide dismantling instructions for models upon request. All dismantling should be done in an electrostatically safe environment as the exposed components are sensitive to electrical discharges.

One or more PCBs are exposed upon dismantling a unit. Due to space constraints, most phones have a single circuit board with surface mounted components on both sides. A circuit board contains several inner layers that are not exposed; therefore, it is not possible to trace the leads by examining the board.

The boards that were examined had 5-10 surface mounted BGA ICs. The ICs were identified by referring to databooks and/or manufacturer websites.

The flash memory circuits were desoldered by pre-baking them for 24 hours at 80°C to remove moisture. Next, the circuits were desoldered using a hot air soldering station with IR preheating. The temperature profile had a peak of 220°C, which was reached in 5 minutes. The

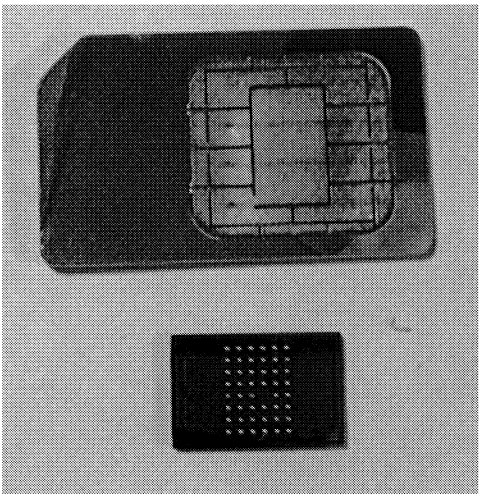

Figure 2. Reballed BGA package with 0.75mm pitch.

chips were removed from the PCB at the maximum temperature using a robotic arm with a vacuum sucker mounted on the soldering station.

Using the vacuum sucker, the chip may be removed with the solder balls intact at a temperature where the solder is marginally above the melting point of 183°C. However, this practice is risky as it is difficult to measure the exact temperature of the solder. Removing a circuit at too low a temperature can rip the pads off, permanently damaging it. Consequently, it was decided to use a temperature of 220°C, which was sufficiently above the melting point (210°C).

Removing the circuits from the board exposed the chip pads on their undersides; these were partially covered with solder residue. The residue was removed very efficiently using a temperature-controlled hot air soldering station and a soldering wick. Note that removing solder residue requires the unit to be heated, which should be done with great care. Finally, reballing was performed for the packages for which stencils were available. Figure 2 shows a reballed BGA package; a SIM card is shown above it for size comparison.

The chips from the mobile phones were mounted on a device programmer, and the contents read using the supplied software. An adapter was not available for some of the packages. For others, the built-in checksums indicated errors during the reading process; it could not be determined if the errors were due to damage caused by desoldering-reballing or for other reasons. A complete memory dump was obtained in most cases, and these files were be analyzed forensically.

6. Embedded Test Technology Method

Most electronic devices are built using embedded test technology. The reason is that manufacturers need automated systems for testing device elements and interconnections, otherwise the quality of the devices cannot be guaranteed.

The current trend in test technology is to use "boundary-scanning." This technique uses a shift register on an integrated component to probe and set the pins of the component. The shift register can be loaded serially from a test access port (TAP) accessible as test points on the device. If the device CPU supports boundary-scanning, the bus can be controlled from the TAP. It is then possible to read or program the memory attached to the bus. This technique is called "in-system programming." The JTAG standard (IEEE 1149.1) [5] governing boundary-scan implementations facilitates the interoperability of components from different manufacturers [6].

6.1 JTAG In-System Programming

JTAG supports in-system programming and reading. If the memory device supports JTAG itself, the TAP of the memory device can be addressed through the JTAG interface. For each memory address, the in-system programmer sets the address value on the device's address bus by shifting in the correct bits through the boundary-scan register. The data bus will then contain the content of the desired memory address and can be read by shifting the value of the boundary-scan register back through the JTAG chain. The entire memory contents can be read by probing memory addresses in this manner.

If the memory device does not support JTAG, in-system programming can be performed using the TAP of another device connected to the memory system bus. This device could be the system processor, which usually has a direct connection to the memory via the system bus. The memory is then read by manipulating the boundary-scan register of the CPU, setting an address on the bus, and reading data through the CPU TAP.

6.2 JTAG Memory Reading Challenges

Note that JTAG is not a bus standard but a standard for component TAPs that facilitates interconnections between components on a board. However, IC manufacturers determine the pin configuration and functionality, and the boundary-scan register configuration. Also, PCB designers can decide if JTAG ports should be interconnected or be acces-

sible from test points on the board. The designer can decide not to use JTAG, leaving the test pins unconnected. If a design uses BGA circuits, attaching a probe to the test pins is impossible and reading memory through JTAG is not an option. However, since manufacturers would otherwise have no way to test their designs, it is common for them to implement JTAG in conjunction with BGA circuits.

Before attempting to use JTAG to read memory, it is necessary to identify the system processor and memory circuits, and their bus connections. Otherwise it would be impossible to find the correct bits in the boundary-scan register. The JTAG test points on the printed circuit board must be located and their signals determined. Also, the memory read/write protocol must be known. Moreover, the correct voltage must be determined; using too high a voltage can damage the circuits.

The voltage can be determined by measurements on a live board. The memory protocol is generally available at the manufacturer's website. However, identifying the bus connections and determining the JTAG test point signals are very difficult without complete documentation. Such documentation is rarely available for mobile phones. Also, implementations for reading memory via JTAG differ from phone to phone. Even a small configuration change to a model may require a separate JTAG interface.

6.3 Experimental Results

Experiments were conducted on an Nokia 5110 mobile phone. The Nokia 5110 was chosen because it was inexpensive, and its service manual and schematics were available. The 5110 is an older model, and its internal memory only stores a few items. Nevertheless, the results can be extended to new mobile phone models.

The 5110 manual was examined for indications of JTAG implementation [12]. The CPU, listed as `MAD2`, had pinouts for `JTRst`, `JTClk`, `JTDI`, `JTMS` and `JTDO`. These pins were coupled in a line `JTAGEMU` onto a connector, listed in the schematics as "not assembled." A 5110 was disassembled, and the test points corresponding to this connector on the system board were found. The connections between the test points and the CPU were partly visible, enabling the test points to be identified.

Voltage measurements indicated that the test points were connected to the CPU's JTAG interface. Test wires were carefully soldered to the connectors using very thin wires from an 80-pin EIDE cable (see Figure 3). Since the test points were very small, the soldering was difficult, but sufficiently thin wires and soldering pate resulted in success. The risk of

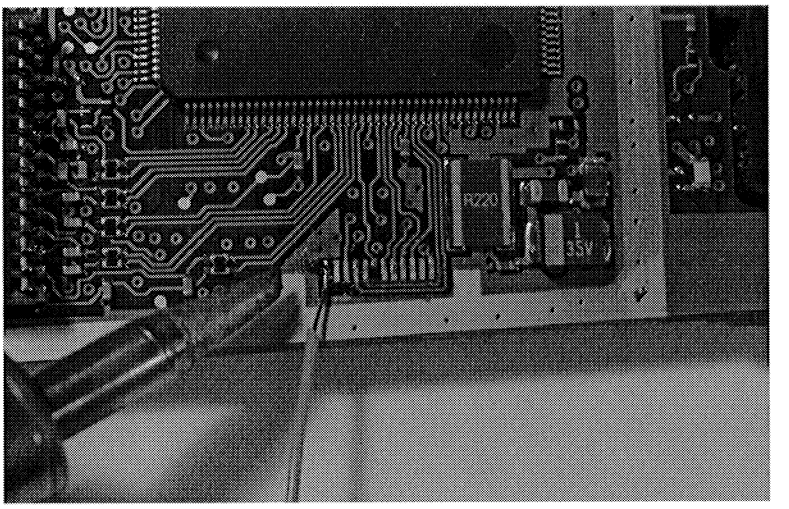

Figure 3. Thin wire attached to a JTAG test point on a Nokia 5110 board.

damaging circuits is lower than for BGA chip desoldering. However, it was tedious to obtain the proper connections without short circuits.

The 5110 was then connected to a computer through the JTAG interface using a breadboard. The Chameleon POD programmable JTAG interface was used in the experiment. When connecting the phone to the interface, the voltage level of the phone system board and the interface should be considered. The 5110 system board uses 2.7V technology, which is in the acceptable range for the Chameleon POD.

Next, the JTAG interface was connected to a Linux machine with the open-source JTAG Tools package [15]. This package allows for device connections via various adapters, including several that can be programmed on the Chameleon.

After some experimentation, a connection with the 5110 processor TAP was established. Although the service manual indicated that the processor design was based on ARM7 and documentation on its JTAG interface was available, the experiment treated the TAP as a "black box," as this would be the case in most situations. JTAG Tools allows for black box analysis via the `discovery` command. This function cycles a `1` through the JTAG chain and detects the number of available TAPs and the length of their instruction register (IR). For the 5110, the IR was determined to be 12 bits long with only one JTAG TAP connected (the processor). The software continued the analysis by probing all possible values of the IR by cycling a `1` through the JTAG chain and detecting the data register length for each instruction. Thus, the values for different instructions were determined. In all, 4096 different possibilities were

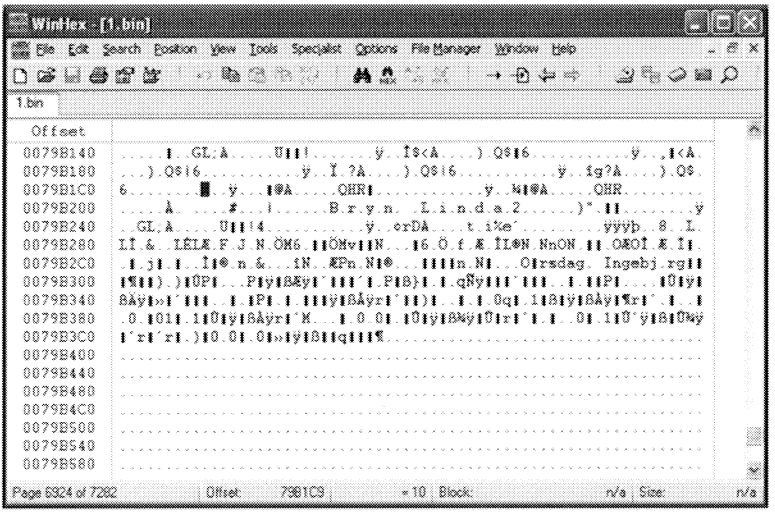

Figure 4. Sony Ericsson T68i memory dump with contact and calendar entries.

tested. Eventually this process resulted in the discovery of commands that could be used to set and probe the boundary-scan register. The command probing process does not change the boundary-scan register and can, therefore, be executed without risk of altering memory content.

Details for setting the boundary-scan register bits to probe the Sharp L28F800BE-TL85 flash memory were obtained from the schematics. These details were implemented in definition files for JTAG Tools, and the memory was then read using the `readmem` command.

7. Memory Analysis

After applying the forensic desoldering and JTAG techniques, the contents of flash memory were available as a binary file. This file was analyzed using a standard hex editor (WinHex [20]).

The mobile phone's Intel 28F640W18T chip yielded an 8 MB memory dump. At first glance, the dump appeared to contain useless information. However, further analysis revealed useful information, such as operating system error messages. This confirmed that the chip was read successfully and that the contents could be analyzed. Further examination showed that the first 4-5 MB of the dump contained binary data of high entropy, corresponding to system software. The remaining dump contained large areas filled with hex `FF` values and smaller areas filled with user data.

Analysis of the data areas revealed GIF images, JPEG images, phone numbers, calendar items and text messages in TPDU format [2]. The

images were extracted and viewed on a computer. The text included contact and calendar items (see Figure 4). Other information associated with these items, e.g., phone numbers and timestamps, had to be obtained by interpreting the binary data.

Memory dumps from other mobile phones were analyzed with similar results. However, much work remains to be done to identify evidentiary items in memory dumps and present them clearly.

7.1 Experimental Results

An important goal was to discover if deleted items could be recovered by analyzing memory dumps and, in particular, if deleted text messages could be recovered. Several experiments were performed that involved reading the internal memory after entering and deleting data via the phone's operating system. The limitations of current memory reading methods made it difficult to conduct more than a few tests. Another difficulty was the lack of a software tools for identification and interpretation of TPDU messages in memory dumps.

Nevertheless, the experiments indicated that text messages were still present in memory after they had been "deleted" using the operating system. Images, MMS messages, calendar items and contacts were also found after they had been deleted. In addition, data pertaining to SIM cards used with the mobile devices were found in some cases.

Interestingly, the search for deleted text messages revealed that memory managers were used to dynamically reallocate memory during phone use. The presence of memory managers has significant ramifications to the forensic analysis of mobile phones.

7.2 Implications

Currently, the forensic analysis of a mobile phone involves keeping the phone on after it has been seized. The memory contents are then analyzed by connecting the phone to a computer using a cable and reading data via the phone's operating system. Since the phone must be on during this activity, it is recommended that the phone be kept on after it is seized.

However, if deleted data can be recovered and if memory management software exists on a mobile phone, keeping the phone turned on for indefinite periods after seizure may not be appropriate. The memory manager might automatically reorganize the memory, potentially overwriting evidence at any time, especially the deleted items.

Techniques used for mobile phone analysis may ultimately mirror the evolution of forensic procedures for computers. Originally, computer

forensic analysis was conducted by booting the operating system and searching for evidence. Now it is recognized that this can destroy evidence; therefore, hard drives are imaged before being analyzed. Similar progress will likely occur in the case of mobile phones. Since mobile phones have much less memory than hard drives, there is a higher risk of losing evidence due overwriting if the phone is kept on.

8. Conclusions

Forensic desoldering and JTAG are promising methods for imaging the internal memory of mobile phones. The methods are technically demanding, and require specialized equipment and skills. However, the resulting memory dumps, which can be analyzed with standard hex editors, contain information of evidentiary value.

The presence of memory managers in modern cell phones raises an important issue. If a phone is kept on after being seized, there is a chance that the memory manager might move memory blocks and/or overwrite important evidence. This suggests that mobile phones must be turned off right after they are seized. Their SIM cards and external flash memories should be removed and analyzed separately. Then, their internal memories should be imaged and analyzed using the methods proposed in this paper.

The results of this work are applicable to other personal electronic devices (e.g., PDAs and GPSs) as well to vehicular navigation systems and other embedded devices [18]. In particular, using physical memory extraction instead of operating system commands facilitates the recovery of deleted information.

References

[1] 3G Partnership Project, *AT Command Set for GSM Mobile Equipment (ETSI ETS 300.642)*, October 1998.

[2] 3G Partnership Project, *Technical Realization of Short Message Service (ETSI ETS 123.040)*, September 2003.

[3] B. Carrier, Sleuthkit (www.sleuthkit.org).

[4] Guidance Software, EnCase (www.encase.com).

[5] IEEE, *IEEE Standard Test Access Port and Boundary-Scan Architecture (IEEE 1149.1)*, Piscataway, New Jersey, 2001.

[6] Intel Corporation, *Designing for On-Board Programming Using the IEEE 1149.1 (JTAG) Access Port*, Santa Clara, California, 1996.

[7] Intel Corporation, *Ball Grid Array Packaging Handbook*, Santa Clara, California, 2000.

[8] Intel Corporation, *Intel Wireless Communications and Computing Package User's Guide (Version 1.2)*, Santa Clara, California, May 2004.

[9] G. Le Bodic, *Mobile Messaging Technologies and Services: SMS, EMS and MMS*, John Wiley, New York, 2005.

[10] N. Lee, *Reflow Soldering Processes and Troubleshooting: SMT, BGA, CSP and Flip Chip Technologies*, Elsevier Science, Oxford, United Kingdom, 2001.

[11] H. Manko, *Solders and Soldering*, McGraw-Hill, New York, 2001.

[12] Nokia Corporation, *Nokia NSE-1 Series Cellular Phone Service Manual*, Salo, Finland, March 1998.

[13] Oxygen Software, Oxygen Phone Manager (www.oxygensoftware .com).

[14] Paraben Corporation, Cell Seizure (www.paraben.com).

[15] SourceForge.net, JTAG Tools (openwince.sourceforge.net/jtag).

[16] SourceForge.net, TULP2G: Forensic framework for extracting and decoding data (tulp2g.sourceforge.net).

[17] B. Vaccaro, R. Shook and D. Gerlach, The impact of lead-free reflow temperatures on the moisture sensitivity performance of plastic surface mount packages, *Proceedings of the International Conference of the Surface Mount Technology Association*, 2000.

[18] R. van der Knijff, Embedded systems analysis, in *Handbook of Computer Crime Investigation: Forensic Tools and Technology*, E. Casey (Ed.), Elsevier, London, United Kingdom, pp. 315-360, 2004.

[19] S. Willassen, Forensics and the GSM mobile telephone system, *International Journal of Digital Evidence*, vol. 2(1), 2003.

[20] X-Ways Software Technology, WinHex: Computer Forensics and Data Recovery Software (www.x-ways.net/winhex).