The Memory Daughter-Card Version 1.5 User’s Manual

Marc A. Viredaz
The Western Research Laboratory (WRL) is a computer systems research group that was founded by Digital Equipment Corporation in 1982. Our focus is computer science research relevant to the design and application of high performance scientific computers. We test our ideas by designing, building, and using real systems. The systems we build are research prototypes; they are not intended to become products.

There are two other research laboratories located in Palo Alto, the Network Systems Lab (NSL) and the Systems Research Center (SRC). Another Digital research group is located in Cambridge, Massachusetts (CRL).

Our research is directed towards mainstream high-performance computer systems. Our prototypes are intended to foreshadow the future computing environments used by many Digital customers. The long-term goal of WRL is to aid and accelerate the development of high-performance uni- and multi-processors. The research projects within WRL will address various aspects of high-performance computing.

We believe that significant advances in computer systems do not come from any single technological advance. Technologies, both hardware and software, do not all advance at the same pace. System design is the art of composing systems which use each level of technology in an appropriate balance. A major advance in overall system performance will require reexamination of all aspects of the system.

We do work in the design, fabrication and packaging of hardware; language processing and scaling issues in system software design; and the exploration of new applications areas that are opening up with the advent of higher performance systems. Researchers at WRL cooperate closely and move freely among the various levels of system design. This allows us to explore a wide range of tradeoffs to meet system goals.

We publish the results of our work in a variety of journals, conferences, research reports, and technical notes. This document is a technical note. We use this form for rapid distribution of technical material. Usually this represents research in progress. Research reports are normally accounts of completed research and may include material from earlier technical notes.

Research reports and technical notes may be ordered from us. You may mail your order to:

Technical Report Distribution
DEC Western Research Laboratory, WRL-2
250 University Avenue
Palo Alto, California 94301 USA

Reports and technical notes may also be ordered by electronic mail. Use one of the following addresses:

Digital E-net: JOVE::WRL-TECHREPORTS
Internet: WRL-Techreports@decwrl.pa.dec.com
UUCP: decpa!wrl-techreports

To obtain more details on ordering by electronic mail, send a message to one of these addresses with the word “help” in the Subject line; you will receive detailed instructions.

Reports and technical notes may also be accessed via the World Wide Web: http://www.research.digital.com/wrl/home.html.
The Memory Daughter-Card Version 1.5
User’s Manual

Marc A. Viredaz

July 1998

Abstract

The *memory daughter-card* is memory extension board for the *Itsypocket computer* developed at Compaq Computer Corporation’s *Western Research Laboratory (WRL)*. It provides an additional bank of flash memory and up to three additional banks of DRAM. It is possible to boot from this daughter-card. This document describes the architecture and the low-level programming model of the memory daughter-card.

*Revision 1.0*
Contents

Disclaimer 1

1 Introduction 1
   1.1 Purpose ............................................................ 1
   1.2 History ............................................................ 1
   1.3 Notations ............................................................ 2

2 Architecture 2
   2.1 Memory system ......................................................... 2
   2.1.1 Flash memory ....................................................... 3
   2.1.2 Dynamic RAM ....................................................... 3

3 Programmer’s model 4
   3.1 Memory daughter-card general-purpose input/output signals .......... 4
   3.2 Non-volatile memory identification structure .......................... 5

References 5

List of Figures

1 Architecture of the memory daughter-card .............................. 2

List of Tables

1 Memory daughter-card general-purpose input/output signals .......... 4
1 Introduction

This document describes the architecture and the programmer’s model of the memory daughter-card for the Itsy pocket computer [Vir98]. It should be considered as a guide for low-level software developers.

A good understanding of the Itsy computer [Vir98] is assumed throughout this report.

1.1 Purpose

The design of the memory daughter-card was aimed at achieving several goals:

- Provide additional flash memory and DRAM (memory extension).
- Be usable as a “safety daughter-card,” i.e., it should be possible to boot from the daughter-card, in order to recover from a corrupted mother-board flash memory.
- Be usable as a system test and debug platform, i.e., it should be possible to boot from the daughter-card and use sockets for the flash memory, so that it can be programmed with an external programmer.
- Provide a prototyping platform to interface additional hardware to the Itsy computer.

1.2 History

The first printed-circuit board (PCB), referred to as memory daughter-card version 1.0, was completed in November 1997. This first prototype had a few minor flaws, all of which could be corrected. The logic design corresponding to a modified (i.e., patched) version 1.0 board is known as memory daughter-card version 1.1. This design corresponds only to a set of schematics, no physical PCB having been manufactured.

A second prototype, named memory daughter-card version 1.5 was complete in July 1998. It corresponds to a version 1.1 system with a few additional features. From the programmer’s point-of-view, there are almost no differences between the versions 1.1 and 1.5.

This document describes the memory daughter-card version 1.5. All relevant differences between the versions 1.1 and 1.5 are outlined in foot-notes.
1.3 Notations

In this report, electrical signals are represented as upper-case names in a sans-serif font (e.g., PWR_EN). Active-low signals are denoted by over-lines (e.g., RESET_OUT), while buses and element of buses are specified by subscripts (e.g., DCD_{31.0}, DCA_0). In the schematics, the same signals are represented using the syntax and conventions of the WindowSIL [Tha97] CAD tools. For example, the signal DCCS_2 appears as ~dccs [2].

2 Architecture

Figure 1 presents the architecture of the memory daughter-card (right part), shown in respect to the Itsy computer (left part). The memory daughter-card versions 1.1 and 1.5 are both compatible with both versions of the Itsy mother-board (i.e., versions 1.1 and 1.5).

2.1 Memory system

The memory daughter-card features a flash memory decoded as static-memory bank 2 and the three dynamic random-access memory (DRAM) banks 1, 2, and 3. Depending on the application’s
needs, only some of these banks (or even none) might be present. Since this daughter-card is bootable, the flash memory is also decoded as static-memory bank 0, from which the StrongARM SA-1100 processor [DEC98] boots.

### 2.1.1 Flash memory

A pair of 16-bit flash-memory circuits implement the 32-bit flash memory. These circuits can be either soldered on the board or put in zero insertion force (ZIF) sockets. Many different devices can be accommodated:

- AMD Am29LV160BOssPT [AMD97c]
- AMD Am29LV800O-ssPT/Am29LV800BOssPT [AMD97c, AMD97d]
- AMD Am29LV400O-ssPT [AMD97a]
- AMD Am29LV200O-ssPT [AMD97b]
- Hitachi HN29V080P-ss/HN29W080P-ss [Hit97c, Hit97d]
- Motorola M29F800A20Pss/M29F800A30Pss [Mot97]
- Sharp LH28F800SGP-Lss [Sha97]

where “O” specifies the internal sector organization, “ss” specifies the speed, “P” specifies the package, and “T” specifies the temperature range. Any other compatible parts can also be used.

Following the Itsy static-memory identification scheme [Vir98], a non-volatile memory identification structure describes the characteristics of the specific parts used on a given daughter-card and hence allows the software to configure the memory interface correctly.

The reset/power-down pin of the flash-memory circuits is asserted (0) during a reset (i.e., hardware, software, or watch-dog reset) and during sleep mode. When the flash memory does not need to be accessed, this signal can also be asserted (0) by setting the signal GPIO16 (DCFLFOFF) to 0 (see Section 3.1).\(^1\) When the Itsy computer must boot from the daughter-card, this signal should never be set to 0 during sleep mode, since the processor would be unable to read the boot memory upon wake up. This can easily be achieved by setting bit 16 of the power manager GPIO sleep state register PGSR of the StrongARM SA-1100 processor [DEC98] to 1.

The ready/not-busy pins of the flash-memory circuits can be monitored using the signals GPIO17 (DCFL0RY/BY), for the least significant 16 data bits D\(_{15...0}\), and GPIO18 (DCFL1RY/BY), for the most significant 16 data bits D\(_{31...16}\) (see Section 3.1).

The hardware write-protection mechanism, featured by some of the supported parts, is never used, and the corresponding pin is always de-asserted (1). However, it is still possible to protect the flash memory against write accesses, by using the jumper provided for this purpose. The flash memory is protected when this jumper is open and is writable when this jumper is short-circuited.

### 2.1.2 Dynamic RAM

Three pairs of 64 Mbit (i.e., \(2^{12} \times 2^{10} \times 16\)) self-refresh DRAM circuits implement the three 32-bit DRAM banks. Many different fast-page mode or enhanced data out (EDO) devices can be accommodated:

\(^1\)On the memory daughter-card version 1.1, the reset/power-down pins are only asserted (0) during sleep mode or when the signal GPIO16 (DCFLFOFF) is set to 0.

<table>
<thead>
<tr>
<th>Signal</th>
<th>Name</th>
<th>Function</th>
<th>Dir.</th>
<th>Def.</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPIO_{16}</td>
<td>DCFLOFF</td>
<td>Daughter-card flash memory force off</td>
<td>O</td>
<td>1</td>
</tr>
<tr>
<td>GPIO_{17}</td>
<td>DCF0RY/0BY</td>
<td>Daughter-card flash memory 0 ready/busy</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>GPIO_{18}</td>
<td>DCF1RY/0BY</td>
<td>Daughter-card flash memory 1 ready/busy</td>
<td>I</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Memory daughter-card general-purpose input/output signals.

Hitachi HM5165160ALTT-ss [Hit97a]
Hitachi HM5165165ALTT-ss [Hit97b]
Samsung KM416V4100AS-Lss/KM416V4100BS-Lss [Sam97a, Sam98a]
Samsung KM416V4104AS-Lss/KM416V4104BS-Lss [Sam97b, Sam98b]
Toshiba TC5165165AFTS-ss [Tos96]

where “ss” specifies the speed. Any other compatible parts can also be used.

Since all DRAM banks must be accessed at the speed of the slowest one, it is best to use the same devices as on the Itsy mother-board, that is, 50 ns EDO DRAM circuits (KM416V4104AS-L5, KM416V4104BS-L5, or TC5165165AFTS-50).

3  Programmer’s model

This section presents additional information on the model that the low-level software has of the memory daughter-card hardware.

3.1 Memory daughter-card general-purpose input/output signals

Table 1 shows the general-purpose input/output signals used on the memory daughter-card. Both input signals can be used as interrupts. After a hardware reset (i.e., power-up or push-button reset), all input/output signals are configured as input. Therefore, a pull-up resistor is used to provide a default value to the signal GPIO_{16} (DCFL0FF), as shown in the last column of Table 1. The function of all signals are:

**GPIO_{16}: DCFL0FF: Daughter-card flash memory force off**

This output signal is used to control the reset/power-down pin of the flash-memory circuits (see Section 2.1.1). When this signal is set to 0, the reset/power-down pins are asserted (0). When it is set to 1, the reset/power-down pins are only asserted (0) during a reset (i.e., hardware, software, or watch-dog reset) and during sleep mode. A pull-up resistor sets the default value of this signal to 1 when the GPIO_{16} pin is configured as input (e.g., after a hardware reset). When the Itsy computer must boot from the daughter-card, this signal should never be set to 0 during sleep mode, since the processor would be unable to read the boot memory upon wake up. This can be easily achieved by setting bit 16 of the power manager GPIO sleep state register PGSR of the StrongARM SA-1100 processor [DEC98] to 1.

---

On the memory daughter-card version 1.1, the reset/power-down pins are only asserted (0) when this signal is set to 0 and during sleep mode.
GPIO\textsubscript{17}: DCFL\textsubscript{0}RY/\overline{BY}: Daughter-card flash memory 0 ready/busy
This input signal is connected to the ready/not-busy pin of the flash-memory circuit used for the least significant 16 data bits \(D_{15..0}\) (see Section 2.1.1). It is set to 0 when the flash-memory circuit is executing an erase or program operation and to 1 when it is ready for use.

GPIO\textsubscript{18}: DCFL\textsubscript{1}RY/\overline{BY}: Daughter-card flash memory 1 ready/busy
This input signal is connected to the ready/not-busy pin of the flash-memory circuit used for the most significant 16 data bits \(D_{31..16}\) (see Section 2.1.1). It is set to 0 when the flash-memory circuit is executing an erase or program operation and to 1 when it is ready for use.

3.2 Non-volatile memory identification structure
Following the Itsy static-memory identification scheme, the flash memory must implement a non-volatile memory identification structure [Vir98]. The class identification value is CID = 0, the read-only bit is R = 0, the daughter-card bit is D = 1, and the width bit is W = 0. The values of the SIZE field and of the different MSC fields depend on the specific parts used on a given daughter-card (see Section 2.1.1). The fields EN\textsubscript{0}, RY/\overline{BY}\textsubscript{0}, EN\textsubscript{1}, and RY/\overline{BY}\textsubscript{1}, defining the general-purpose input/output signals, have the values EN\textsubscript{0} = 16 = 10_{16}, RY/\overline{BY}\textsubscript{0} = 17 = 11_{16}, EN\textsubscript{1} = 16 = 10_{16}, and RY/\overline{BY}\textsubscript{1} = 18 = 12_{16}.

References


WRL Research Reports


``Global Register Allocation at Link Time.’’ David W. Wall. WRL Research Report 86/3, October 1986.


“Network Locality at the Scale of Processes.”  

“Cache Write Policies and Performance.”  
**Norman P. Jouppi.** WRL Research Report 91/12, December 1991.

“Packaging a 150 W Bipolar ECL Microprocessor.”  

“Observing TCP Dynamics in Real Networks.”  

“Systems for Late Code Modification.”  

“Piecewise Linear Models for Switch-Level Simulation.”  

“A Practical System for Intermodule Code Optimization at Link-Time.”  

“A Smart Frame Buffer.”  

“Recovery in Spritely NFS.”  

“Tradeoffs in Two-Level On-Chip Caching.”  
**Norman P. Jouppi & Steven J.E. Wilton.** WRL Research Report 93/3, October 1993.

“Unreachable Procedures in Object-oriented Programming.”  

“An Enhanced Access and Cycle Time Model for On-Chip Caches.”  
**Steven J.E. Wilton and Norman P. Jouppi.** WRL Research Report 93/5, July 1994.

“Limits of Instruction-Level Parallelism.”  
**David W. Wall.** WRL Research Report 93/6, November 1993.

“Fluoroelastomer Pressure Pad Design for Microelectronic Applications.”  

“A 300MHz 115W 32b Bipolar ECL Microprocessor.”  

“Link-Time Optimization of Address Calculation on a 64-bit Architecture.”  

“ATOM: A System for Building Customized Program Analysis Tools.”  

“Complexity/Performance Tradeoffs with Non-Blocking Loads.”  

“A Better Update Policy.”  

“Boolean Matching for Full-Custom ECL Gates.”  

“Software Methods for System Address Tracing: Implementation and Validation.”  

“Performance Implications of Multiple Pointer Sizes.”  


