



# IET7820 Design Specification

Atari 7800 RAM Cart development system.

Rev: PA2

Date: 25/01/06

Author: Pontus Oldberg

Author:  
Pontus Oldberg

## Table Of Content

1.Revision History:.....	2
2.Introduction.....	3
3.Terminology.....	3
4.System Description.....	3
5.RAMCart system requirements.....	4
6.Hardware requirements.....	5
7.Hardware Implementation suggestion.....	5
7.1.Device Control Register (DCR) - \$0440.....	6
7.2.Device Status Register (DSR) - \$0440.....	8
7.3.Bank Select Register (BS0, BS1).....	8
7.4.SIFLIP - \$0444.....	8
7.5.DRER - \$0445.....	9
8.Hardware Considerations.....	9



# IET7820 Design Specification

Atari 7800 RAM Cart development system.

Rev: PA2

Date: 25/01/06

Author: Pontus Oldberg

---

## 1. Revision History:

<i>Date:</i>	<i>Rev:</i>	<i>Comments</i>
2005-01-25	PA2	Added this table. Updated requirements table with new requirements. Added the SIFLIP and DRER registers.
2005-01-21	PA1	First release.

## IET7820 Design Specification

### 2. Introduction

This specification describes the design of the development cartridge IET7820 (RAMCart) targeted the Atari 7800 game system. This specification contains the overall design for this cartridge with in depth details where it is necessary to closer describe wanted functionality.

### 3. Terminology

This specification uses a number of terms to refer to the different subsystems of the RAMCart.

**SRAM:**

Static Read And Modify memory. The SRAM is a volatile memory type.

**FLASH:**

FLASH refers to a special memory technology where the memory cell can be erased and reprogrammed many times. A FLASH memory is a non volatile memory type.

**ROM:**

Read Only Memory, a memory type that was more common when the Atari 7800 was manufactured. This is a non volatile memory type.

**CPLD:**

Complex Programmable Logic Device. This is a integrated circuit that can be programmed to hold any combination of complex and non complex logic equations.

**PCB:**

Printed Circuit Board. The circuit board that holds and connects the different integrated circuits in the design.

**USB:**

Universal Synchronous Bus. A serial communication bus used in by many PC's to communication with peripherals. The USB2.0 can transfer data up to 12MBit/s.

### 4. System Description

The RAMCart is a system for developing applications (games) on the Atari 7800 Super Game System.

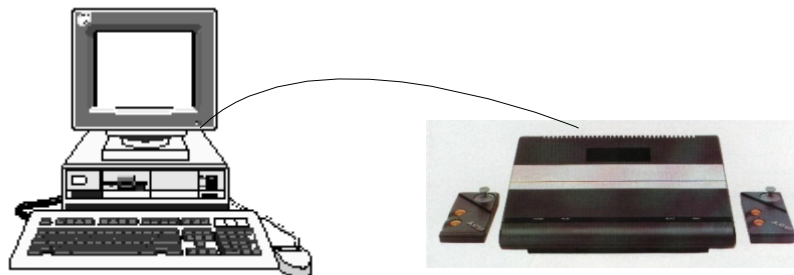
In a typical system setup you would have the Atari 7800 serving as the basic development system, and a PC on which you would develop your software on. The PC needs to have software for either compiling or assembling the code that the developer is writing.

As of up to now most developers have been using the Cuttle Cart 2 (CC2) system to download the software to the A7800 and testing it. This option is certainly more appealing than the crude home built systems that was available before the CC2 was available. However the CC2 was not intended to be a full fledged development system, it simply has the option to download code via a serial interface and executing it.

Instead the RAMCart development system was devised. The RAMCart is a SRAM based cartridge which also contains a FLASH memory that holds a monitor software that can download code, insert breakpoints, start and stop the program, read and modify the content of the entire system. With all this power combined with a PC software that can give the monitor a high level user interface we now have a professional development system that will ease the task of debugging code written for the A7800.

The RAMCart is connected to the PC via a USB cable allowing code, data and commands to be transferred to and from the RAMCart at very high speeds. The fact that the transfer speed is so high and the information that needs to go to and from the A7800 is so small should make it extremely interesting to write the PC software. One could for instance have on the fly assembly/compilation and downloading to the RAMCart making the debug process extremely efficient.

Below is a simple system overview.



Picture 1

## 5. RAMCart system requirements.

It has been decided that the RAMCart should have following characteristics:

1. 256-512 Kbyte of SRAM (ROM emulation memory)
2. Sufficient amount of FLASH memory. 32KByte should do but as larger sizes are easier to come by it is probably wise to use a modern 4Mbit device which is still in production.
3. A USB2.0 compatible interface. USB1.1 would do but with the larger feature set of USB2.0 it has been decided that a USB2.0 device should be used.
4. The PCB manufactured must fit in a casing as supplied by VideoWiz.

## 6. Hardware requirements

The special needs of the RAMCart makes it necessary that it complies with a number of requirements.

1. The device **must** have at least 32KByte of FLASH memory to hold the monitor software.
2. The device **must** have at least 256KByte RAM which servers as emulated cartridge ROM, it is recommended that 512Kbyte is implemented. The actual implemented size can be calculated by the processor.
3. The device **must** support switching operating modes between the monitor FLASH and the emulation RAM.
4. The device **should** support "inline" operation mode switching, either by delayed switching (synching the switch with a Jump instruction) or by reserving a small area of the monitor FLASH that is common for both modes of operation.
5. The device **must** have at least one control register that controls the behaviour of the unit.
6. The device **must** have at least one status register that can report back the internal status of the unit.
7. The device **must** have support for bank select logic according to the Atari Super Cartridge specification. See [this](#) external document for more detailed information.
8. A fine grain memory location select mechanism **must** be implemented. For instance, control and status registers of the device **must** only occupy one position in the memory space.
9. In order to support fast downloading the implementation **should** support background SRAM write accesses. It is however recommended that the implementation support this, if not supported the downloaded content needs to be buffered in local A7800 RAM, and on an unexpanded unit this is only 4KByte in total. The support must be reported in a device status register.
10. The device **must** have a mechanism to completely disable all emulative read functions (control registers, USB and RAM) at startup, and only have the FLASH enabled. This is due to how the A7800 disables the cartridge by simply gating A15 with its internal enable signal.

## 7. Hardware Implementation suggestion

As a guide the author of this document suggests the following register model for the unit.

<b>Mnemonic</b>	<b>Description</b>	<b>Address</b>
DCR	Device Control Register	\$0440 (W)
DSR	Device Status Register	\$0440 (R)
USBDAT	USB Data Port	\$0441 (R/W)
BS0	Bank Select Register 0	\$8000 (W) Emulation mode \$0442 (W) Monitor mode
BS1	Bank Select Register 1	\$4000 (W) Emulation mode \$0443 (W) Monitor mode
SIFLIP	SI Toggle register	\$0444
DRER	Device Read Enable Register.	\$0445

## 7.1. Device Control Register (DCR) - \$0440

This register Controls the behaviour of the RAMCart Unit. The following bits are identified and defined for this register.

<b>Bit 7</b>	<b>Bit 6</b>	<b>Bit 5</b>	<b>Bit 4</b>	<b>Bit 3</b>	<b>Bit 2</b>	<b>Bit 1</b>	<b>Bit 0</b>
RAMEN	FLASHWE	RAMWE	USBEN	USBINT	EM2	EM1	EM0

**RAMEN** – When set (=1) this bit disables the FLASH BIOS memory and enables the emulation SRAM. The bank selection registers (BS0 and BS1) are now mapped to \$8000 and \$4000 respectively.

**FLASHWE** – When set (=1) this bit enables write cycles to the FLASH memory. If the **RAMEN** bit is set (=1) this bit has no effect on the system.

**RAMWE** – When set (=1) this bit enables write cycles to the SRAM. Setting this bit to 1 at the same time as **FLASHWE** will override the **FLASHWE** function. If this bit is set at the same time as the SRAM is disabled (**RAMEN** = 0) you can read from the FLASH and write to the SRAM at the same time (A mode called background write), thus reducing download time. Setting this bit to 1 when the SRAM is enabled (**RAMEN** = 1) has no effect since the SRAM is fully protected during ROM emulation.

**USBEN** – When set (=1) this bit enables the on board USB transceiver. When the USB transceiver is disabled all its related registers are also disabled and are invisible.

**USBINT** – When set (=1) this bit enables USB transfer interrupts. Interrupts happen on two different USB events, data needed (USB TX FIFO is empty) and data available (USB RX FIFO has data). The user should check its RXF and TXE in the status register.

**EM2 - EM0** – These bits define the emulation mode of the SRAM. The unit supports 4 different emulation modes described as follows:

- Mode 0 (000), Linear memory mapped mode 32K. This memory mode provides 32KByte of linearly mapped emulation SRAM from \$8000 - \$FFFF. No bank switching is available in this mode thus writing to BS0 or BS1 has no effect. This mode probably benefits implementation wise from only accessing one memory, thus needing to be able to linearly access one of the SRAMs on board.
- Mode 1 (001), Linear memory mapped mode 48K. This memory mode provides 48KByte of linearly mapped emulation SRAM from \$4000 - \$FFFF. No bank switching is available in this mode thus writing to BS0 or BS1 has no effect. This mode probably benefits implementation wise from only accessing one memory, thus needing to be able to linearly access one of the SRAMs on board.
- Mode 2 (010), Standard Atari Super Cartridge mode. This mode maps the topmost 16KByte (Bank 32) memory sector into the area between \$C000 - \$FFFF. The area between \$8000 - \$BFFF can then be mapped into any of the 16KByte sectors in the SRAM bank. Mapping is performed with the BS0 register. Only the 5 least significant bits are used in the register.
- Mode 3 (011), Extended Atari Super Cartridge mode. This mode maps the topmost 16KByte (Bank 32) memory sector into the area between \$C000 - \$FFFF. The area between \$8000 - \$BFFF can then be mapped into any of the 16KByte sectors in the SRAM bank. Mapping is performed with the BS0 register. Only the 5 least significant bits are used in the register. This mode is also extended with having Bank 0 mapped into \$4000 - \$7FFF.
- Mode 4 (100), X-Game cartridge mode. This mode is an X-Game specific memory mode which requires the use of an extra bank select register. This mode maps the topmost 16KByte (Bank 32) memory sector into the area between \$C000 - \$FFFF. The area between \$8000 - \$BFFF can then be mapped into any of the 16KByte sectors in the SRAM bank. Mapping is performed with the BS0 register. Only the 5 least significant bits are used in the register. This mode is also extended with the ability to map any memory bank into the area \$4000 - \$7FFF. To accomplish this another bank select register has been implemented, BS1. Only the 5 least significant bits are used in this register. With this mode, care has to be taken if the Xboard expansion board is installed so that there is no collision. This mode is only useful for the X-Game FLASH cartridges for the Atari 7800.
- All other mode settings are illegal and may cause undesirable effects.

## 7.2. Device Status Register (DSR) - \$0440

This register holds all status bits of the RAMCart Unit. The following bits are identified and defined for this register.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Not Used	RESET	PWREN	TXE	RXF	BACKRAM	VER1	VER0

**RESET** – Indicates the status of the USB controller. If this bit is 0 the device is in reset mode or disconnected from the USB cable. The unit needs to be powered by a USB cable in order to work.

**PWREN** – This is just a feed trough of the PWREN bit from the USB controller. It can be used to determine the status the USB controller is at the moment.

**TXE** - When high (=1), do not write data into the USB FIFO. When low, data can be written into the USB FIFO.

**RXF** - When high, do not read data from the USB FIFO. When low, there is data available in the USB FIFO which can be read.

**BACKRAM** – If this bit is set this unit supports SRAM background writes while the FLASH monitor is enabled.

**VER1, VER0** – Version information. Two bits indicating the version of this board.

## 7.3. Bank Select Register (BS0, BS1)

The Bank Select registers (BS0 and BS1) controls what bank is selected in each corresponding bankable area. The bankable areas are, for BS0 \$8000 - \$BFFF and for BS1 \$4000 - \$7FFF. Both registers are 5 bits deep enabling 32 banks of 16Kbyte each to be selected in each area.

Both registers are write only and have dual memory locations. In order to be able to write to these register when the unit is in monitor mode (FLASH enabled) the registers are mapped into the alternative locations \$441 (BS0) and \$442 (BS1)

## 7.4. SIFLIP - \$0444

A write access (with any data) will toggle the SI pin on the FTDI USB chip on the board. This will cause the USB controller to immediately send out any data held in its internal buffers. This way short packets of data can be transferred across the USB connection at very high packet speeds. If this function was not available you would have to pad USB packets and keep track of this in your driver.



## 7.5. DRER - \$0445

This register enables or disables any read registers available in the device.

The A7800 comes up from power on with the inserted cartridge disabled. This is done internally in the A7800 simply by gating some address lines with an enable signal. This causes a problem for any cartridge with peripherals mapped to an address below \$8000, as read accesses to an address which equals the address that the cartridge sees when it is disabled would cause a bus collision. For instance, the status register on this device resides at address \$0440. This means that while the RAMCart is disabled by the A7800, it will also recognize all addresses with the following address bit pattern, XX0X010001000000 as valid read accesses. Writing 0x01 to this register enables reads from the status register and writing 0x00 disables reads again. Write accesses are always enabled.

## 8. Hardware Considerations

The PCB **must** be designed so that it can fit inside a standard plastic enclosure as manufactured by the company VideoWiz or old salvaged A7800 cartridges. The USB connector should be mounted so that it is easy to access and relieves the cable from any strain.

The PCB connector on the unit **must** be gold plated with chemical gold in order to give good lasting contact to the base unit.