Serial EEPROMs in Sega Genesis / Mega Drive cartridges

Version 3 (07/01/09)

Serial EEPROM devices are present in a few Sega Genesis cartridges and used as external Backup RAM. As you will later see, they can NOT be emulated as usual parallel SRAM and also use various modes depending on their memory size. Most Genesis/Megadrive emulators out there does not emulate those custom devices, resulting in various glitches (beside the fact that saving game data is impossible) in games that rely on this.

Here are some links to datasheets for each kind of EEPROM types for those who couldn't understand my bad english ;)

24C01(mode1) : http://www.icmic.com/datasheets/X24C01.pdf

24C01-24C16(mode2) : http://www.atmel.com/dyn/resources/prod_documents/doc0180.pdf

24C64 (mode3):http://www.phys.hawaii.edu/~bryce/component_data/24LC65-ISM.pdf

Basically, the protocol used so that the Main CPU (68000), considered as MASTER device, can communicate with the serial EEPROM (considered as SLAVE device), could be summarized this way:

- SLAVE is controlled by the MASTER using two serial lines:
    - /SDA is the **DATA line**, used to send/receive data (READ/WRITE)
    - /SCL is the **CLOCK line** used to manage the clock cycles (WRITE ONLY)

  Each line is mapped in the 68000 address space at a specific addres, at one specific bit position (see various used mappers in game database) and can have LOW (bit=0) or HIGH (bit=1) state.

- A clock cycle starts by a transition of /SCL from LOW to HIGH and ends by a transition of /SCL from HIGH to LOW. This is used by the MASTER to indicate when /SDA state refers to a **new valid DATA bit** (received or sent).

- MASTER and SLAVE devices communicate through 8-bits DATA word, which require **8 clock cycles**. The 9th cycle is called the **ACK cycle** where the receiver (SLAVE or MASTER, depending on the current operation) must send an acknowledge to the sender by setting /SDA line LOW during this cycle, when it is ready to receive more data.

- Each READ or WRITE operation must be initiated by the MASTER with a **START** condition and terminated with a **STOP** condition:
    - START condition is set when /SDA is changed from HIGH to LOW while /SCL remains HIGH
    - STOP condition is set when /SDA is changed from LOW to HIGH while /SCL remains HIGH

- After the START condition, depending on the EEPROM mode, the MASTER immediately send one or more **8-bit words** to specify the address that should be later read or written. There are 3 known EEPROM modes:

o Mode 1 (24C01 only) have a maximal size of 128bytes (7 bits address) and need only one address word. The 7 first bits set the memory address to be read/write and the 8th bit set the operation type (READ or WRITE)
o Mode 2 (24C01 to 24C16) can have up to 8 devices linked together, with a maximal size of 2 Kbytes (11 bits address max.), which means one 24C16, two 24C08, four 24C04 or eight 24C02/24C01 :

- the 1st word has 4 fixed bits, then 3bits setting the device address and/or eventually the upper bits of the memory address to read/write (24C04-24C16). The 8th bit set the type of the operation (READ or WRITE).

- the 2nd word set the lower bits of the memory address to read/write (only 7bits for 24C01 in mode 2, because it's only 128 bytes wide)

o Mode 3 (24C32 and more) work like Mode 2 excepted that the maximal address size is 64 Kbytes (max. 16 bits address) per device (a maximum of eight devices can be linked with no size restriction).

- the 1st word is like in the previous mode, excepted bits d3-d1 always set the device address (0-7). Last bit always set the operation type.

- the 2nd word set the upper bits of the memory address to read/write (the number of bits depends on the memory size)

- the 3rd word set the eight lower bits of the memory address to read/write.

NB: In modes 2 and 3, before a READ operation, only the 1st word is actually sent, the EEPROM contains an address counter that saves the address of the next word to be accessed (incremented from a previous READ or WRITE operation, as stated below)

- After a WRITE operation has been initiated, the MASTER will send one or more 8bits DATA words through the /SDA line. After each written word, the EEPROM must send a ACK and increment the address for the next data word to be written. The address will roll up to base memory address when maximal writepage size has been reached. If the MASTER send a STOP condition, the WRITE operation ends, otherwise the WRITE operation continues to next address.

- After a READ operation has been initiated, the MASTER will read one or more 8bits DATA words through the /SDA line. After each received word, the MASTER will send a ACK and the EEPROM will increment the address of the next data word to be read. The address will roll up to 0 when maximal memory size has been reached. If the MASTER does not send a ACK during the 9th cycle, the READ operation ends and the EEPROM waits for STOP condition, otherwise, the READ operation continues to next address.

This is pretty how eeprom's access are emulated in Genesis Plus (see eeprom.c for more details). Thanks a lot to 8bitwizard from spriteminds.net forums who initially give me indications about the various EEPROM types and mappers used in Sega Genesis games.

Following is a game database of Sega Genesis games that I found using serial EEPROM, listed by companies (as they usually use specific company mappers).Their specific caracteristics (mappers, mode, size,...) are also mentionned, deducted from various testings.

Feel free to use these informations and my sourcecode in your own project.

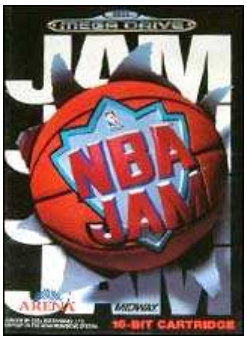
ACCLAIM (81)

TYPE #1

MODE: 8BITS WORD ADDRESS (MODE2)
SIZE_MASK: 0xFF (24C02)
PAGE_MASK: 0x03
SDA_IN : 0x200001 (bit 0)
SDA_OUT: 0x200001 (bit 1)
SCL : 0x200001 (bit 1)

NBA Jam (UE)(J)

T-081326, T-81033
header OK ($200001-$200001)



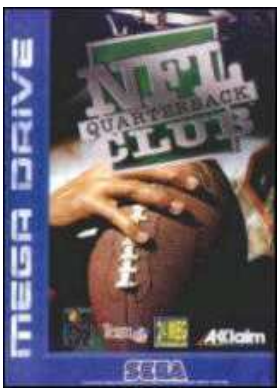
***NB**: this game uses a different EEPROM mapper than any other Acclaim games. Also it uses 16-bits write & read to access EEPROM.*

TYPE #2

MODE: 8BITS WORD ADDRESS (MODE2)
SIZE_MASK: 0xFF (24C02)
PAGE_MASK: 0x03
SDA_IN : 0x200001 (bit 0)
SDA_OUT: 0x200001 (bit 0)
SCL : 0x200000 (bit 0)

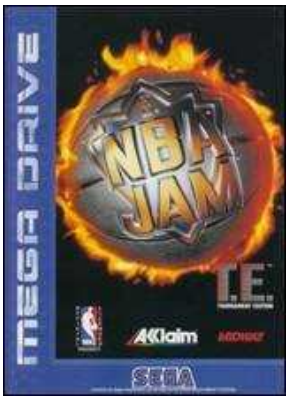
NFL Quarterback Club (JUE)

T-081276
header OK ($200000-$200001)



Blockbuster World Video Game Championship II (U)
NBA Jam Tournament Edition (UE)(J)

T-81406, T-81143
header KO



***NB**: Rev 00 of the game has buggy eeprom support, only Rev01 version will correctly save game data.*

TYPE #3

MODE: 8BITS WORD ADDRESS (MODE2)
SIZE_MASK: 0x7FF (24C16)
PAGE_MASK: 0x07
SDA_IN : 0x200001 (bit 0)
SDA_OUT: 0x200001 (bit 0)
SCL : 0x200000 (bit 0)

NFL Quarterback Club 96 (UE)

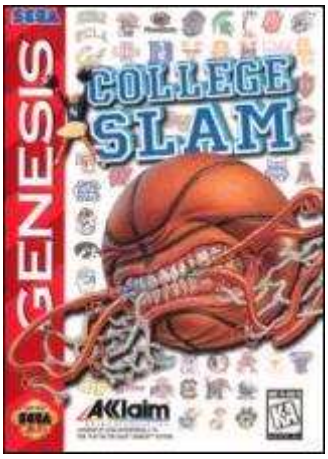
T-081586
header OK ($200000-$200001)

TYPE #4

MODE: 16BITS WORD ADDRESS (MODE3)
SIZE_MASK: 0x1FFF (24C64)
PAGE_MASK: 0x07
SDA_IN : 0x200001 (bit 0)
SDA_OUT: 0x200001 (bit 0)
SCL : 0x200000 (bit 0)

College Slam (U)

T-81576
header OK ($200001-$200001)



Frank Thomas Big Hurt Baseball (UE)

T-81476
header OK ($200000-$200001)

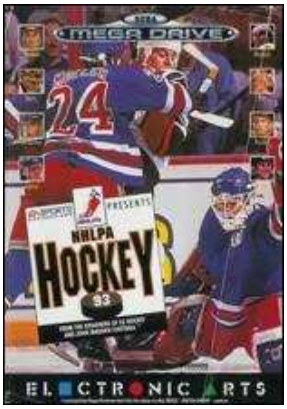
ELECTRONIC ARTS (50)

TYPE

MODE: 7BITS WORD ADDRESS (MODE1)
SIZE_MASK: 0x7F (24C01)
PAGE_MASK: 0x03
SDA_IN : 0x200001 (bit 7)
SDA_OUT: 0x200001 (bit 7)
SCL : 0x200001 (bit 6)

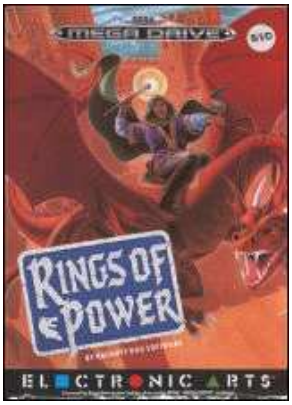
NHLPA Hockey 93 (UE)

T-50396
header KO



Rings of Power (UE)

T-50176
header KO



These two games use 16-bits write & read to access EEPROM.

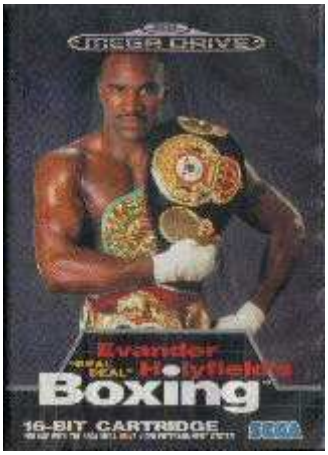
SEGA

TYPE

MODE: 7BITS WORD ADDRESS (MODE1)
SIZE_MASK: 0x7F (24C01)
PAGE_MASK: 0x03
SDA_IN : 0x200001 (bit 0)
SDA_OUT: 0x200001 (bit 0)
SCL : 0x200001 (bit 1)

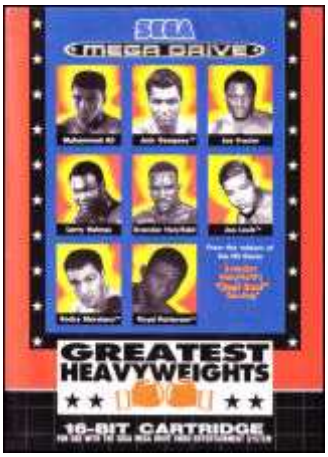
Evander 'Real Deal' Holyfield's Boxing (UE)(J)

MK-1215, G-4084
header OK ($200001-$200001)



Greatest Heavyweights of the Ring (J)(U)(E)

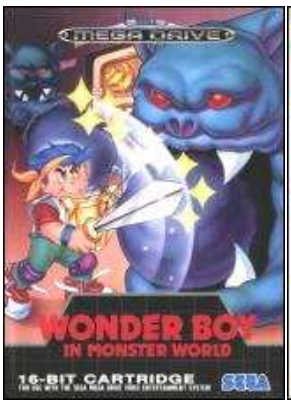
G-5538, MK-1228, PR-1993
header OK ($200001-$200001)



Wonder Boy in Monster World (UE)

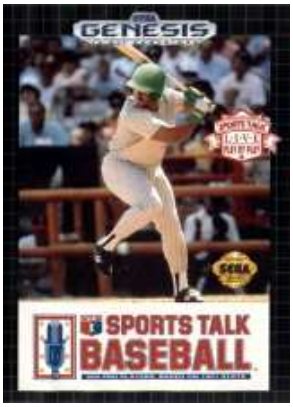
Wonder Boy V - Monster World III (J)

G-4060
header OK ($200001-$200001)



Sports Talk Baseball

00001211-00
header KO



Honoo no Toukyuuji Dodge Danpei

00004076-00
header OK ($200001-$200001)

CAPCOM (12)

TYPE

MODE: 7BITS WORD ADDRESS (MODE1)
SIZE_MASK: 0x7F (24C01)
PAGE_MASK: 0x03
SDA_IN : 0x200001 (bit 0)
SDA_OUT: 0x200001 (bit 0)
SCL : 0x200001 (bit 1)

(same as SEGA mapper)

Megaman - The Wily Wars (E) / Rockman Mega World (J) [alt]

T-12046, T-12053 (checksum = 0xEA80)
header OK ($200001-$200001)

***NB**: the original version of Rockman Mega World (J) uses traditional SRAM*

CODEMASTERS

TYPE #1

MODE: 8BITS WORD ADDRESS (MODE2)
SIZE_MASK: 0x3FF (24C08)
PAGE_MASK: 0x0F
SDA_IN : 0x300000 (bit 0)
SDA_OUT: 0x380001 (bit 7)
SCL : 0x300000 (bit 1)

Micro Machines 2 - Turbo Tournament (E) (J-Cart)

T-120096-50
header KO

***NB**: this game needs the external RAM to be initialized with 0xFF*

Micro Machines Military (E) (J-Cart)

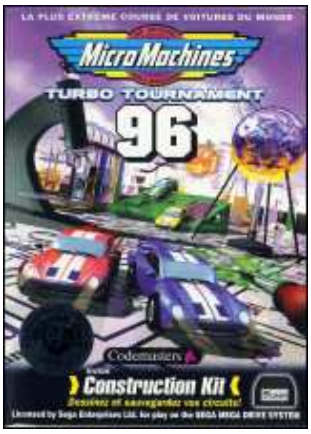
00000000-00 (checksum = 0x168B or 0xCEE0)
header KO

TYPE #2

MODE: 8BITS WORD ADDRESS (MODE2)
SIZE_MASK: 0x7FF (24C16)
PAGE_MASK: 0xF
SDA_IN : 0x300000 (bit 0)
SDA_OUT: 0x380001 (bit 7)
SCL : 0x300000 (bit 1)

Micro Machines Turbo Tournament 96 (E) (J-Cart)

00000000-00 (checksum = 0x165E or 0x2C41)
header KO



TYPE #3

MODE: 16BITS WORD ADDRESS (MODE3)
SIZE_MASK: 0x1FFF (24C64)
PAGE_MASK: ?
SDA_IN : 0x300000 (bit 0)
SDA_OUT: 0x380001 (bit 7)
SCL : 0x300000 (bit 1)

Brian Lara Cricket 96 / Shane Warne Cricket

T-120146-50
header KO