



XPLODER CART INFO

By Codeman

1/28/99 rev .1

email [damage4@yahoo.com](mailto:damage4@yahoo.com)

web <http://damage-inc.webjump.com>

#### Hardware Info :

256K Bytes EEprom SST 29EE020  
Addressed 0x1F000000 - 0x1F040000

The Psx will look for external rom in the pio port by looking for this string "Licensed by Sony Computer Entertainment Inc." add address 0x1f000004 and 0x1f000084

There are 2 entry points in the rom,  
The psx gets the 1st entry point address from 0x1f000080 which is called after power up before the white sony screen  
The psx gets the 2nd entry point address from 0x1f000000 which is called after the cd sony lic screen.

#### Standard Printer Port Interface PSX side port addresses

0x1f060000 On/Off switch address check bit 0  
0x1f060001 Read/Write Data port  
0x1f060002 handshake from pc check bit 0

PSX outputs data to PC 3 bits at a time on the folling lines  
slct, paper end, busy and uses /ack for handshake

YOU MUST INVERT BUSY WHEN READING FROM PC  
ALSO PRINTER SELECT FROM PC IS INVERTED

To send a byte to psx  
out data to psx  
set printer select line high  
wait for ack from psx to go high  
set printer select line low  
wait for ack from psx to go low

To get a byte from psx  
set printer select line high  
psx sets ack high and puts d6 on slct, d7 on pe  
read data  
set printer select line low  
psx sets ack low and puts d5 on busy, d4 on pe, d3 on slct  
read data  
set printer select line high  
psx sets ack high and puts d2 on busy, d1 on pe, d0 on slct  
read data  
set printer select line low  
psx sets ack low and puts low on busy, low on pe, high on slct  
read data  
set printer select line high

reassemble byte

# READ DATA FROM PSX

DATA FLOW	STEP 1	STEP 2	STEP 3	STEP4
PC PRINTER SEL	HIGH	LOW	HIGH	LOW
PSX SLCT	D6	D3	D0	HIGH
PSX PE	D7	D4	D1	LOW
PSX /BUSY	HIGH	D5	D2	LOW
PSX ACK	HIGH	LOW	HIGH	LOW

XPLoader COMMANDS :  
not all commands here yet.

**send bin file to psx**  
send 0x57  
send 0x53 upload bin file  
send address 4 bytes  
send file size 3 bytes  
send data  
send checksum 2 bytes

**Jump to address**  
send 0x57  
send 0x0f Jump to address  
send address 4 bytes

