insideGadgets

GBxCart RW v1.1

A device for backing up GB, GBC and GBA cart’s game ROMs, save games and restoring saves from your PC.

Main Features

- Backup games ROM to your PC
- Backup save games to your PC
- Restore save games from your PC
- Supports Gameboy, Gameboy Colour, Gameboy Advance cartridges
- Supports SRAM/Flash/EEPROM saves for Gameboy Advance cartridges
- GUI or Console interfacing programs

Other Features

- Backup GB Camera (active images) to BMP with a one click program called GB Camera Saver
- GB Sachen ROM mapper reading support
- GBA “24-in-1” Flash cart ROM mapper reading support (may not work for all flash carts)

Gameboy Flash Cart supported/tested

- Catskull – Gameboy 32K Flash Cart

Specifications

Size: 54mm (L) x 57mm (W) x 9mm (H)
Voltage: switchable between 3.3V (for GBA) and 5V (for GB/C)
Weight: 18 grams

Requirements

USB Mini cable

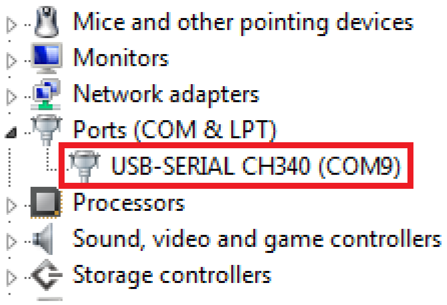
Contents

How to Use

1. Before inserting the cartridge, select the correct voltage for your cartridge using the switch (GB/GBC use 5V while GBA uses 3.3V) and connect the USB cable to ensure the correct voltage LED lights up and then unplug the device. If the device isn't automatically detected, you will need to open up Device manager, find the "USB 2.0 Serial" device and update the driver using the files in the \Drivers folder.
2. Insert your cartridge to the device and connect the USB cable.
3. For the GUI interface, follow "using the GUI Program" steps or for the console program follow "using the Console Program" steps.
4. When changing cartridges, always unplug the USB cable from the device and follow step 1 again.

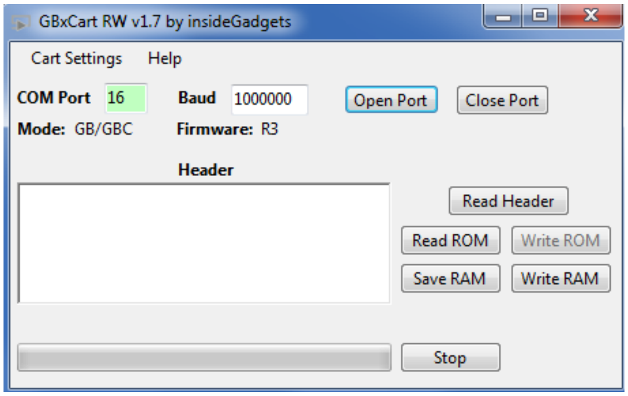
Using the GUI Program

1. Download and Install Microsoft Visual C++ Redistributable Packages (x86 and x64) for Visual Studio 2015 - https://www.microsoft.com/en-us/download/details.aspx?id=53840



Mice and other pointing devices
Monitors
Network adapters
Ports (COM & LPT)
USB-SERIAL CH340 (COM9)
Processors
Sound, video and game controllers
Storage controllers

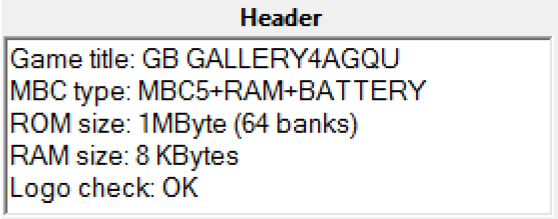
2. Open \Interface_Programs\GBxCart_RW_vx.x_GUI_Interface\GBxCart_RW_vx.x.exe and change the COM port to match the COM port your operating system has assigned the device, this can be found in device manager.


GBxCart RW v1.7 by insideGadgets

COM Port 16 Baud 1000000

Mode: GB/GBC Firmware: R3

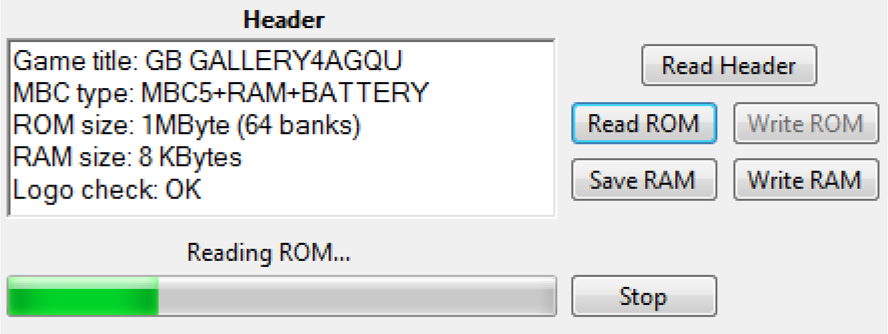
3. Click the Open Port button; if the COM port text box turns green then it has successfully connected. If it turns red, please double check the COM port and try again. Once the port is successfully opened, the port number and baud rate will be remembered for next time.


Header

Game title: GB GALLERY4AGQU
MBC type: MBC5+RAM+BATTERY
ROM size: 1MByte (64 banks)
RAM size: 8 KBytes
Logo check: OK

4. Press the Read Header button and the cartridge information should show up. (GBA carts take 4 seconds as tests are being performed to determine the ROM and SRAM/Flash/EEPROM sizes). Make sure that the "Logo check" comes back as OK otherwise the cartridge may not be read correctly and you will have to disconnect the device, remove and re-insert the cartridge.

   If the cartridge information doesn't match what you know to be correct, you can press the "Specify Cart Info" button to change the ROM size, RAM size/type.


Header

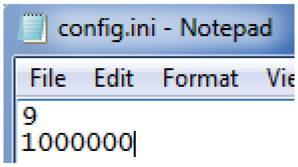
Game title: GB GALLERY4AGQU
MBC type: MBC5+RAM+BATTERY
ROM size: 1MByte (64 banks)
RAM size: 8 KBytes
Logo check: OK

Reading ROM...

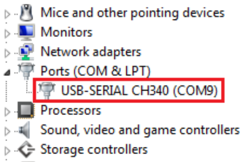
5. You can now press the corresponding buttons to read the ROM, save the RAM to the PC (save game) or write the RAM (save game) from the PC.

We recommended verifying your Gameboy ROMs using BGB (a Gameboy emulator) and for Gameboy Advance games we recommend using VisualBoyAdvance. It's a good idea to verify your save files too by running the ROM and verifying the save file loads ok.

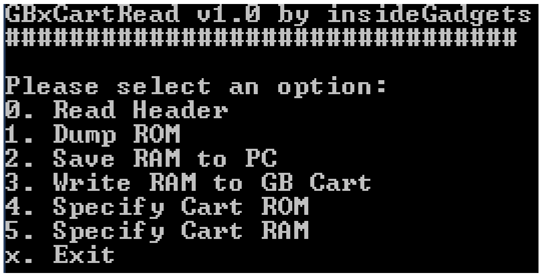
Using the Console Program



```
9
1000000
```

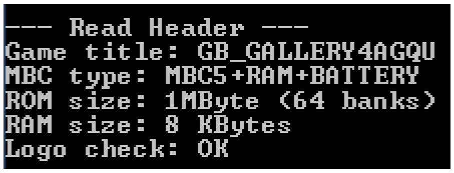


1. Open the \Interface_Programs\GBxCart_RW_vx.x_Console_Interface\config.ini file and change the first number to your COM port, e.g. 9 = COM9 and change the second number to the baud rate, default is 1Mbit (1000000) which should work fine. Your COM port can be found in device manager.



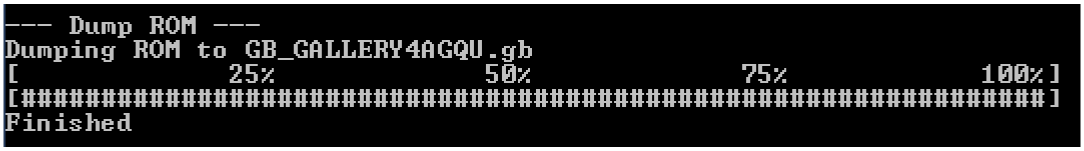
```
GBxCartRead v1.0 by insideGadgets
################################

Please select an option:
0. Read Header
1. Dump ROM
2. Save RAM to PC
3. Write RAM to GB Cart
4. Specify Cart ROM
5. Specify Cart RAM
x. Exit
```

2. Open GBxCart_RW_Console_v1.0.exe and the option menu should appear.



```
--- Read Header ---
Game title: GB_GALLERY4AGQU
MBC type: MBC5+RAM+BATTERY
ROM size: 1MByte (64 banks)
RAM size: 8 KBytes
Logo check: OK
```

3. Press “0” and hit enter to read the cartridge information. Make sure that the “Logo check” comes back as OK otherwise the cartridge may not be read correctly and you will have to disconnect the device, remove and re-insert the cartridge.



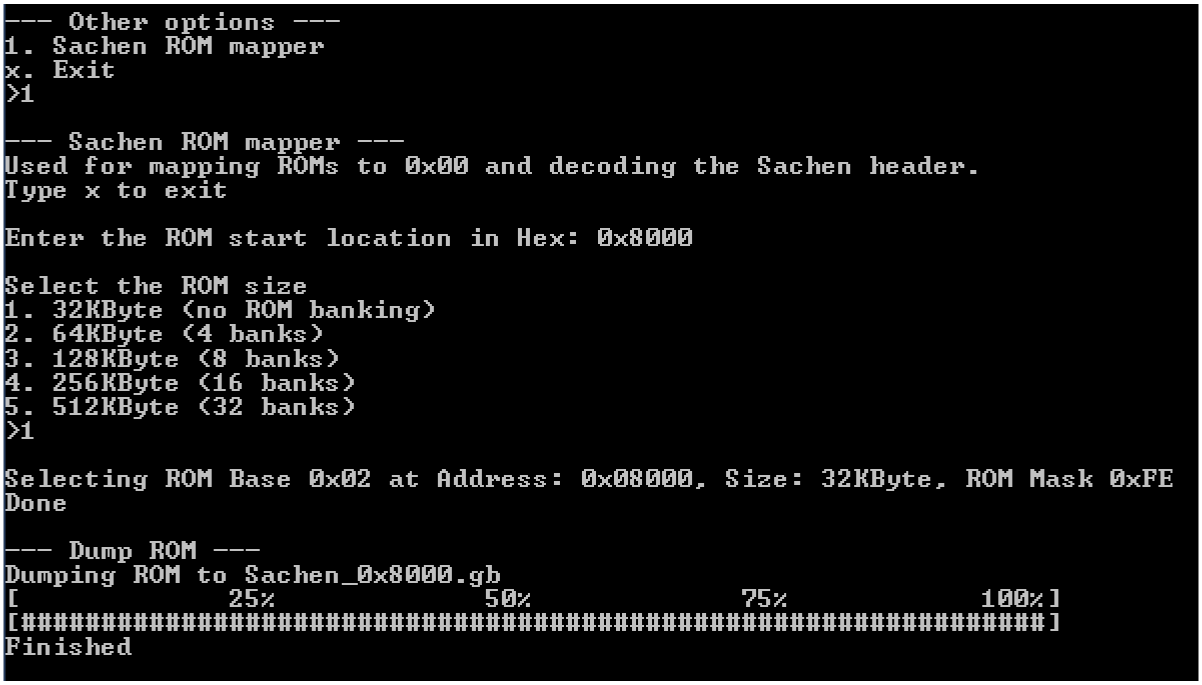
```
--- Dump ROM ---
Dumping ROM to GB_GALLERY4AGQU.gb
[             25%              50%              75%             100%]
[#################################################################]
Finished
```

4. You can now press the corresponding buttons to read the ROM, save the RAM to the PC (save game) or write the RAM (save game) from the PC.

We recommended verifying your Gameboy ROMs using BGB (a Gameboy emulator) and for Gameboy Advance games we recommend using VisualBoyAdvance. It’s a good idea to verify your save files too by running the ROM and verifying the save file loads ok.

Sachen ROM Mapper support

You can use the “Other options” in console program to select ROMs to dump within a Sachen cart; the ROMs will be dumped into their own file. You will be required to input the ROM starting address in hex and the number of banks.

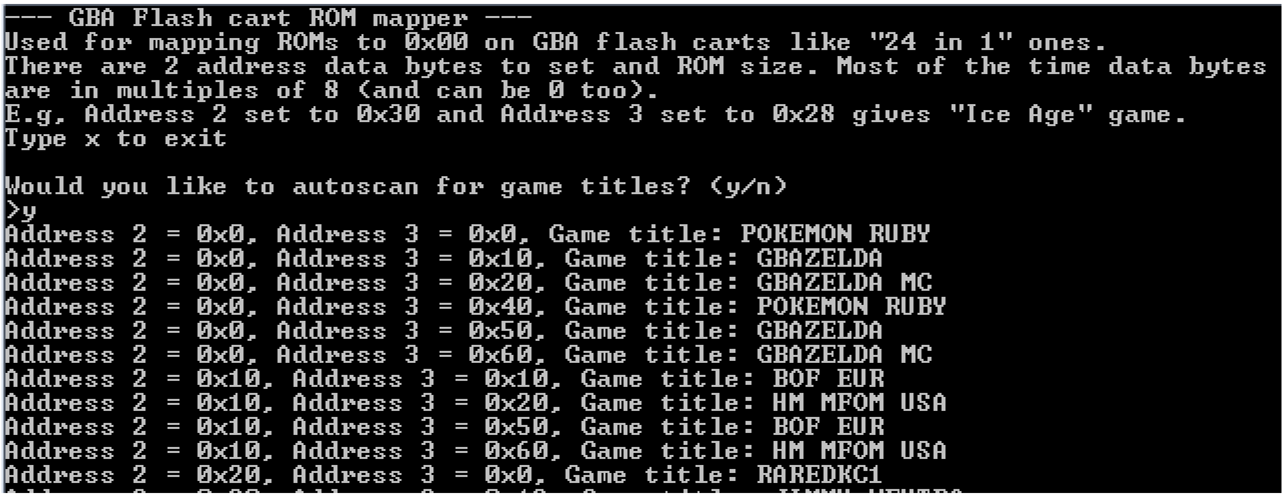


```
--- Other options ---
1. Sachen ROM mapper
x. Exit
>1

--- Sachen ROM mapper ---
Used for mapping ROMs to 0x00 and decoding the Sachen header.
Type x to exit

Enter the ROM start location in Hex: 0x8000

Select the ROM size
1. 32KByte (no ROM banking)
2. 64KByte (4 banks)
3. 128KByte (8 banks)
4. 256KByte (16 banks)
5. 512KByte (32 banks)
>1

Selecting ROM Base 0x02 at Address: 0x08000, Size: 32KByte, ROM Mask 0xFE
Done

--- Dump ROM ---
Dumping ROM to Sachen_0x8000.gb
[             25%             50%             75%            100%]
[################################################################]
Finished
```

For more information please check https://wiki.tauwasser.eu/view/Sachen_Mappers and https://board.byuu.org/viewtopic.php?f=16&t=1557

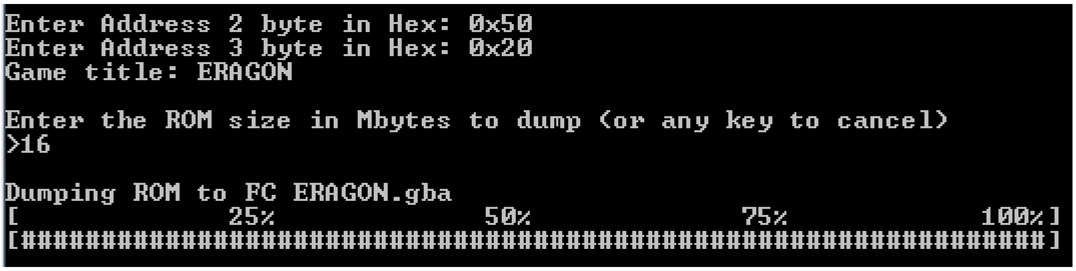
GBA Flash Cart ROM Mapper support

You can use the “Other options” in console program to select ROMs to dump within a GBA Flash Cart similar to the “24-in-1” carts; the ROMs will be dumped into their own file named “FC_<Gametitle>.gba” after you select Address 2 and Address 3/4. Some ROMs may not work properly in emulators.



```
--- GBA Flash cart ROM mapper ---
Used for mapping ROMs to 0x00 on GBA flash carts like "24 in 1" ones.
There are 2 address data bytes to set and ROM size. Most of the time data bytes
are in multiples of 8 (and can be 0 too).
E.g, Address 2 set to 0x30 and Address 3 set to 0x28 gives "Ice Age" game.
Type x to exit

Would you like to autoscan for game titles? (y/n)
>y
Address 2 = 0x0, Address 3 = 0x0, Game title: POKEMON RUBY
Address 2 = 0x0, Address 3 = 0x10, Game title: GBAZELDA
Address 2 = 0x0, Address 3 = 0x20, Game title: GBAZELDA MC
Address 2 = 0x0, Address 3 = 0x40, Game title: POKEMON RUBY
Address 2 = 0x0, Address 3 = 0x50, Game title: GBAZELDA
Address 2 = 0x0, Address 3 = 0x60, Game title: GBAZELDA MC
Address 2 = 0x10, Address 3 = 0x10, Game title: BOF EUR
Address 2 = 0x10, Address 3 = 0x20, Game title: HM MFOM USA
Address 2 = 0x10, Address 3 = 0x50, Game title: BOF EUR
Address 2 = 0x10, Address 3 = 0x60, Game title: HM MFOM USA
Address 2 = 0x20, Address 3 = 0x0, Game title: RAREDKC1
```

Once entering the menu, it will prompt you if you wish to automatically scan the addresses for Game Titles.

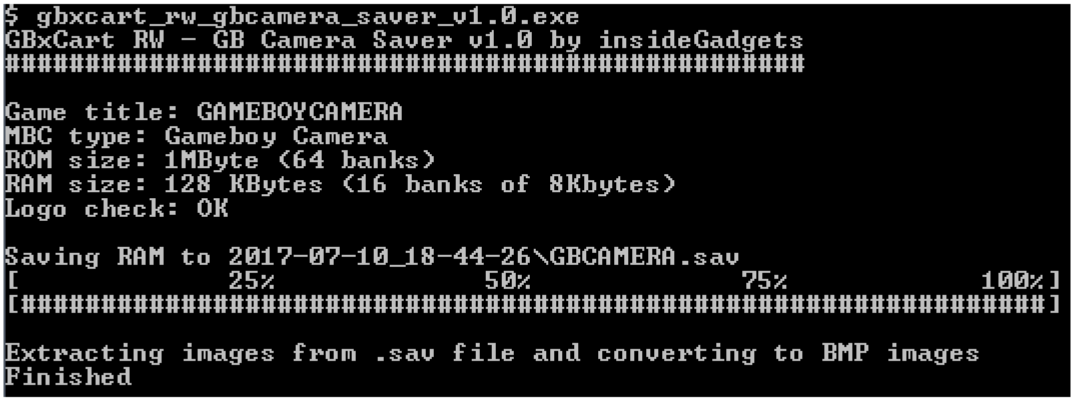


```
Enter Address 2 byte in Hex: 0x50
Enter Address 3 byte in Hex: 0x20
Game title: ERAGON

Enter the ROM size in Mbytes to dump (or any key to cancel)
>16

Dumping ROM to FC ERAGON.gba
[         25%              50%              75%            100%]
[##############################################################]
```

You will be required to input the Address 2 and Address 3/4 in hex and enter the ROM size you wish to dump.

Gameboy Camera Saver

This program is useful if you are constantly using the Gameboy Camera and wish to save the images quickly to BMP files. Found in the \Interface_Programs\GBxCart_RW_GBCamera_Saver_vx.x folder.

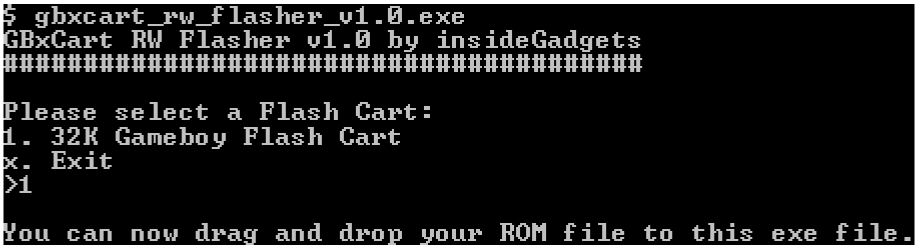


```
$ gbxcart_rw_gbcamera_saver_v1.0.exe
GBxCart RW - GB Camera Saver v1.0 by insideGadgets
##################################################

Game title: GAMEBOYCAMERA
MBC type: Gameboy Camera
ROM size: 1MByte (64 banks)
RAM size: 128 KBytes (16 banks of 8Kbytes)
Logo check: OK

Saving RAM to 2017-07-10_18-44-26\GBCAMERA.sav
[             25%             50%             75%            100%]
[################################################################]

Extracting images from .sav file and converting to BMP images
Finished
```

Automatically creates a folder with the current day/time, saves the GB camera memory to this folder and extracts the active (non-deleted) images from the save to separate BMP files labelled 1.bmp to 30.bmp. It’s a good idea to check the images created do match the images stored on the GB Camera.

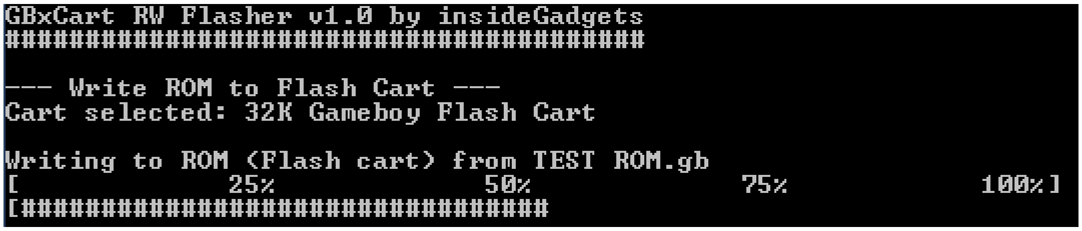
Console Flasher (for writing ROMs to Flash Carts)

This program is useful if you would like drag and drop interface to write ROMs to your flash carts. Please check the list of supported flash carts on the first page of this manual.
Found in the \Interface_Programs\GBxCart_RW_Console_Flasher_vx.x folder.



```
$ gbxcart_rw_flasher_v1.0.exe
GBxCart RW Flasher v1.0 by insideGadgets
########################################

Please select a Flash Cart:
1. 32K Gameboy Flash Cart
x. Exit
>1

You can now drag and drop your ROM file to this exe file.
```

Open the program first and select the Flash Cart you will be writing the ROM to.



```
GBxCart RW Flasher v1.0 by insideGadgets
########################################

--- Write ROM to Flash Cart ---
Cart selected: 32K Gameboy Flash Cart

Writing to ROM (Flash cart) from TEST ROM.gb
[             25%             50%             75%            100%]
[#################################
```

Now you can drag and drop your ROM file to gbxcart_rw_flasher_vx.x.exe as many times as you like.

Raw I/O Access

If you wish to read/write to a special cartridge which the regular functions don't support, you can access the raw I/O of the ATmega8515L on board with the following COM commands. Commands should be sent as characters, a null terminator byte (0) is used to determine the end of some strings sent. No responses will be returned except for reading a PORT.

Enable/Disable common lines RD/WR/MREQ-CS/Reset-CS2 set high after each command
(recommended to disable so you drive these lines yourself)
To enable: M1 To disable: M0

Set pins as inputs/output
To set PB7 as an input: IB0x80 (include a null-terminator byte at the end)
The value set will be applied as DDR &= ~(setValue);

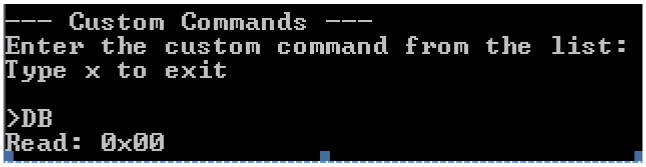
To set PB7 as an output: OB0x80 (include a null-terminator byte at the end)
The value set will be applied as DDR |= (setValue);

Set output low
To set PB7 low: LB0x80 (include a null-terminator byte at the end)
The value set will be applied as PORTB &= ~(setValue);

Set output high
To set PB7 high: HB0x80 (include a null-terminator byte at the end)
The value set will be applied as PORTB |= (setValue);

Read all pins of a PORT
Returns a byte representing all pins of that port, e.g. 128
To read PORTB: DB
Port mappings to Gameboy Slot pins (for PCB v1.1)

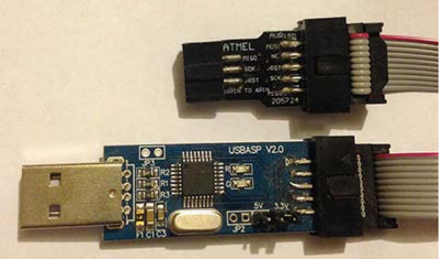
PB0 = A0	PA0 = A8	PC0 = D0/A16
PB1 = A1	PA1 = A9	PC1 = D1/A17
PB2 = A2	PA2 = A10	PC2 = D2/A18
PB3 = A3	PA3 = A11	PC3 = D3/A19
PB4 = A4	PA4 = A12	PC4 = D4/A20
PB5 = A5	PA5 = A13	PC5 = D5/A21
PB6 = A6	PA6 = A14	PC6 = D6/A22
PB7 = A7	PA7 = A15	PC7 = D7/A23
PD6 = *WR	PD4 = *MREQ/CS	PE1 = Audio In
PD5 = *RD	PE2 = *RESET/CS2	



```
--- Custom Commands ---
Enter the custom command from the list:
Type x to exit

>DB
Read: 0x00
```

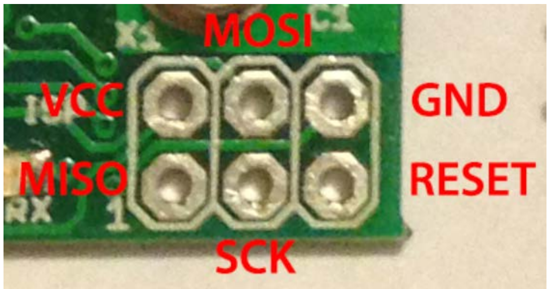
You can use "Custom commands" the console program to input these commands into by selecting option 6 (it will automatically add null terminator bytes when needed).

Re-program the ATmega using an AVR programmer

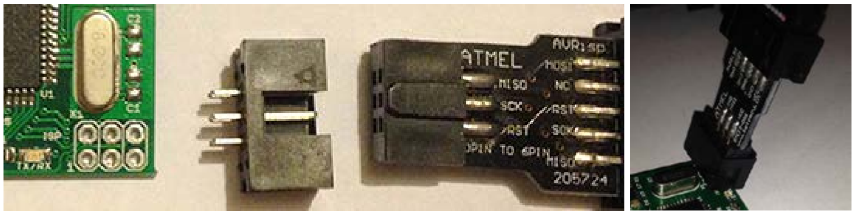


You will require an AVR programmer such as USBTinyISP, USBasp, etc, one that has a 6 pin interface (if the device has a 10 pin interface you can use a 10 pin to 6 pin converter cable). You'll also need to download the Hardware/Firmware zip file from https://www.insidegadgets.com/projects/gbxcart-rw/

Make sure that the voltage selected on the AVR programmer matches the 3.3V or 5V that you have selected on the GBxCart RW board.



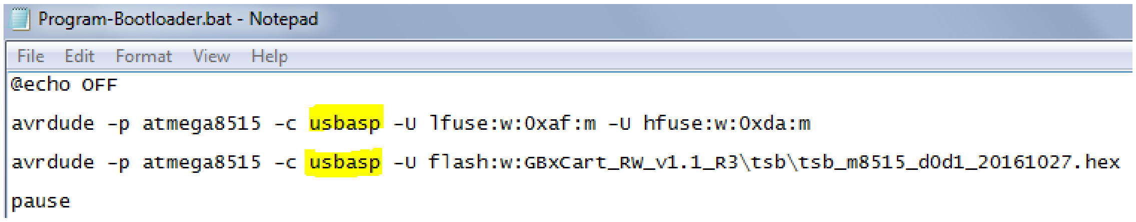
The PCB has an ISP 6 pin interface where you can solder male headers.



If you don't wish to do any soldering or would rather keep it clean, you can buy a DC3-6P connector, plug it into the AVR programmer then insert it into the pins and hold it at an angle so the pins touch the pads.

You now have 2 options, either program the new firmware release directly or program TinySafeBoot which becomes the boot loader so you can easily program any future updates without the need of an AVR programmer.

If you wish to program the boot loader and new firmware release

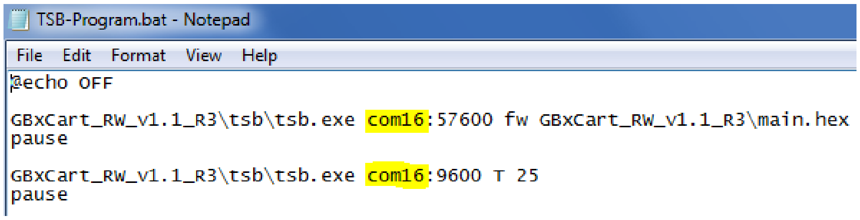


```
@echo OFF
avrdude -p atmega8515 -c usbasp -U lfuse:w:0xaf:m -U hfuse:w:0xda:m
avrdude -p atmega8515 -c usbasp -U flash:w:GBxCart_RW_v1.1_R3\tsb\tsb_m8515_d0d1_20161027.hex
pause
```

Depending on your programmer, you may need to open the /MCU_Files/Program-Bootloader.bat file to change the programmer to suit, by default its set to “usbasp”.

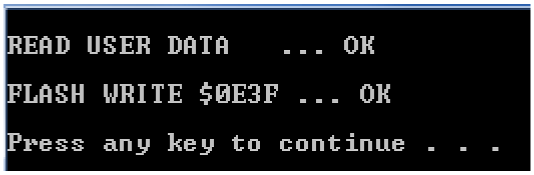
Once you have the programmer pins connected to the board, insert the USB cable for your programmer and then run the /MCU_Files/Program- Bootloader.bat file which should show it uploading and should return “Avrdude Done”.

Please unplug the AVR programmer.



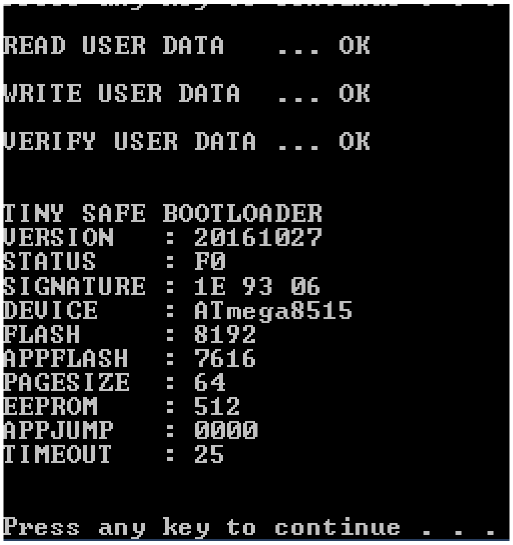
```
@echo OFF
GBxCart_RW_v1.1_R3\tsb\tsb.exe com16:57600 fw GBxCart_RW_v1.1_R3\main.hex
pause
GBxCart_RW_v1.1_R3\tsb\tsb.exe com16:9600 T 25
pause
```

Open up the /MCU_Files/ TSB-Program.bat, you will need to change the COM port to suit.



```
READ USER DATA    ... OK
FLASH WRITE $0E3F ... OK
Press any key to continue . . .
```

Run /MCU_Files/ TSB-Program.bat and you should see the above result. You should also see the LEDs on the board light up.



```
READ USER DATA   ... OK
WRITE USER DATA  ... OK
VERIFY USER DATA ... OK

TINY SAFE BOOTLOADER
VERSION   : 20161027
STATUS    : F0
SIGNATURE : 1E 93 06
DEVICE    : ATmega8515
FLASH     : 8192
APPFLASH  : 7616
PAGESIZE  : 64
EEPROM    : 512
APPJUMP   : 0000
TIMEOUT   : 25

Press any key to continue . . .
```

Unplug and reconnect the USB cable, then press a key and you should see the above result.

The boot loader and new firmware have now been uploaded.

If you wish to program the new firmware release directly without a boot loader
Depending on your programmer, you may need to open the /MCU_Files/Program.bat file to change the programmer to suit, by default its set to "usbasp".

Once you have the programmer pins connected to the board, insert the USB cable for your programmer and then run the /MCU_Files/Program.bat file which should show it uploading and should return "Avrdude Done".

Software Revision History

Console Interface
v1.6 – 26 August 2017

- Remove check for EEPROM if SRAM/Flash found
- Updated check of SRAM/Flash, re-read the first 64 bytes twice (if the cart has an EEPROM, sometimes data lines can come back with random data in the first 64 bytes read).

v1.5 – 17 August 2017

- Added better checks for 512Kbit or 1Mbit Flash
- Added a check for flash sector erases that may take longer than the usual 25ms, waits until byte 0x00 of the sector reads 0xFF. **** Thanks to EGJ for reporting the issue and assisting in troubleshooting*

v1.4 – 13 August 2017

- Fixed bug when writing to a GBA 1Mbit Flash save, wasn't ending the write before switching banks
- Fixed bug where some GBA cartridges would be stuck in Flash ID mode and would keep repeating some bytes over and over again when backing up the save to the PC. We now exit Flash ID mode another way.
  **** Thanks to Black Phoenix for bringing both of the above issues up and for the vast amount of time spent troubleshooting it*

- Fixed bug where specify flash type/size was not taking effect
- Updated Sachen ROM mapper to support v1.1 PCB
- Added support for reading ROMs from GBA "24-in-1" Flash Carts (may not work for all flash carts)

v1.3 – 8 May 2017

- Added alternative method (very slow) to read Gameboy Camera if you aren't able to re-program the firmware to R2
- Added additional check for EEPROMs which seem to allow 4Kbit or 64Kbit reads without any issues
- Added special check for "Chu-Chu Rocket!" SRAM
- Added support for "." in game title
- Fixed bug when reading EEPROM with GBA 32MB carts (firmware update R2 required)

v1.2 – 4 May 2017

- Updated ROM size check from 16MB to 32MB
- Update the check for GBA SRAM/Flash between 256k/512k to correct some carts been detected as 256k

v1.1 – 4 April 2017

- Added custom commands functionality for raw I/O access
- Added Sachen mapper support so you can dump ROMs from the Sachen cart into their own files
  **** Thanks to Voltagex for bringing this up and helping with troubleshooting*
  **** Thanks to Tauwasser for the Sachen information/post explaining how it works*

v1.0 – 11 March 2017

- Initial Release

GUI Interface

v1.7 – 26 August 2017

- Added support for writing ROMs to Gameboy 32K Flash Carts (works with R2+, faster with R3+)
- Updated check of SRAM/Flash, re-read the first 64 bytes twice (if the cart has an EEPROM, sometimes data lines can come back with random data in the first 64 bytes read).
- Changed GUI text font, added “Manual” and “Check for updates” options

v1.6 – 17 August 2017

- Added better checks for 512Kbit or 1Mbit Flash
- Added a check for flash sector erases that may take longer than the usual 25ms, waits until byte 0x00 of the sector reads 0xFF. **** Thanks to EGJ for reporting the issue and assisting in troubleshooting*

v1.5 – 13 August 2017

- Fixed bug when writing to a GBA 1Mbit Flash save, wasn’t ending the write before switching banks
- Fixed bug where some GBA cartridges would be stuck in Flash ID mode and would keep repeating some bytes over and over again when backing up the save to the PC. We now exit Flash ID mode another way.
  **** Thanks to Black Phoenix for bringing both of the above issues up and for the vast amount of time spent troubleshooting it*

v1.4 – 24 June 2017

- Added check to see if device is still connected before “Read Header”, “Read ROM”, etc buttons are pressed
- Added the current mode (GBA or GB/GBC) and firmware version once connected

v1.3 – 8 May 2017

- Added alternative method (very slow) to read Gameboy Camera if you aren’t able to re-program the firmware to R2
- Added additional check for carts with EEPROM that seemed to allow 4Kbit or 64Kbit reads without any issues
- Added special check for “Chu-Chu Rocket!” SRAM size
- Added support for “.” in GBA game title
- Fixed bug when reading EEPROM with GBA 32MB carts (firmware update R2 required)

v1.2 – 4 May 2017

- Fixed a bug when opening/closing COM port that would cause the program to lock up
- Fixed a bug that wouldn't open the "Specify Cart Info" menu when a GBA cart was inserted
- Updated ROM size check from 16MB to 32MB
- Update the check for GBA SRAM/Flash between 256k/512k to correct some carts been detected as 256k

v1.1 – 4 April 2017

- Updated some text labels

v1.0 – 11 March 2017

- Initial Release

GB Camera Saver

v1.0 – 12 July 2017

- Initial Release

Console Flasher

v1.0 – 26 August 2017

- Initial Release - Support for writing ROMs to Gameboy 32K Flash Carts (works with R2+, faster with R3+)

Hardware/Firmware Revision History

PCB v1.1

(Firmware R3) – 26 August 2017

- Now using TinySafeBoot bootloader so ATmega MCU can be reprogrammed without the need for a programmer. (If you purchased a board before 23 August 2017, you will require an AVR programmer, please check "Reprogramming the MCU" section in the manual)
- Added support for writing ROMs to Gameboy 32K Flash Carts

(Firmware R2) – 8 May 2017

- Added extra nop when reading Gameboy Camera SRAM as some of data would change sometimes leading to artifacts on the pictures
- Changed A0-A23 address when preparing to read EEPROM from 0xFFFFFF to 0xFFFF00 to support GBA 32MB carts

(Firmware R1) – 27 April 2017

- Switched from ATmega32A to ATmega8515L, some assigned ports have changed

PCB v1.0

(Firmware R2) – 8 May 2017

- Added extra nop when reading Gameboy Camera SRAM as some of data would change sometimes leading to artifacts on the pictures
- Changed A0-A23 address when preparing to read EEPROM from 0xFFFFFF to 0xFFFF00 to support GBA 32MB carts

(Firmware R1) – 11 March 2017

- Initial Release (Errata - The switch operates in reverse, for example, moving it to the left should select 3.3V but it selects 5V and moving it to the right should select 5V but it selects 3.3V.)