



User Manual
version 0-01-alpha

April 25th, 2011

FAT FuriousAdvanceTracker
version 0-01-alpha
Code and design: Cyril Brouillard

Thanks to **Guillaume Dualé** and **Aline Bouquillon**
for their patience and encouragement.

Manual translation by Joan Dolç
Errors and comments to ds_joan@yahoo.es

Contents

1 Introduction

1.1 Introduction and notice	3
1.2 Why FAT ?	4
1.3 How to use FAT	4
1.4 A first sound	5
1.5 Data structure	6
1.6 Gameboy Advance technical data	7

2 Screens

2.1 Screen menu	8
2.1.1 Commands	9
2.2 Song screen	9
2.2.1 Commands	9
2.3 Blocks screen	10
2.3.1 Commands	11
2.4 Notes screen	12
2.4.1 Commands	13
2.5 Instrument screen	14
2.5.1 Commands	15
2.5.2 Pulse type instrument	16
2.5.3 Wave type instrument	18
2.5.4 Noise type instrument	18
2.5.5 Sample type instrument	19
2.6 Project screen	19
2.6.1 Commands	20
2.7 Effects screen	20
2.8 Live screen	20
2.9 Compose screen	21
2.9.1 Commands	22

3 Effects and commands	23
-------------------------------	-----------

1 Introduction

1.1 Introduction and notice

FAT is a software for musical composition conceived for the Gameboy Advance. With **FAT** you can compose music directly on your handheld using its technical capabilities. **FAT** is laid out in the simplest possible manner. However, it is necessary to learn how to use it before you can exploit all its possibilities.

Attention! Version 0-01 (the one currently available) is an early version. The software is probably full of bugs and unfinished. Paris was not built in a day: **FAT** will evolve during the coming months and years (or at least I hope so). In case you detect a bug or any strange behavior, don't hesitate! Send a bug report to <http://www.furiousadvancetracker.com/bugtracker/>. It won't take you more than 5 minutes and it will help me improve the software. Thanks in advance! (and a kiss too!)

Those who have already a foot in chiptune music composition won't take long to notice the huge resemblance to LSDJ (LittleSoundDJ). I have in fact regarded LSDJ as THE model, although I have added my personal touch. There are also some features which are completely original, some have been revisited and some have disappeared.

Of course I don't intend any pretense to say that **FAT** is anywhere near LSDJ. However, I do hope that some day **FAT** will become a reference for chiptune composition on the GBA.

So, all in all, big thanks to you who are reading this manual probably to try out **FAT**. I hope you enjoy it!

1.2 Why FAT?

In 2011, there exist already several solutions for composing music on a Gameboy Advance:

- Nanoloop 2.3
- M4G tracker
- and probably others I couldn't find...

I find none of these solutions viable (although they are perfectly functional!).

– Nanoloop 2.3 is an excellent program, but its availability is almost null (and by the effects of the "market", its price prohibitive). It is in the end a solution difficult to get a hold of.

– M4G tracker is also excellent, but its development seems to have been abandoned and not all of its planned features are complete. I have also been unable to make it work.

– As for the others, I haven't found them...

Finally, programming your own tracker is at least tons of fun too!

1.3 How to use FAT

FAT is provided as a .gba file. This file can be used directly on your computer with an emulator (VirtualBoy Advance for instance).

But the advantage of having a Gameboy is the portability! And an emulated sound will never equal the original one. To execute **FAT** on a real Gameboy Advance, you will need:

- A real Gameboy (duh!)
- An EZ-Flash or SuperCardToSD cartage, or others... see here for instance: <http://www.r4i-r4.co.uk>
- An SD card that fits the cartage

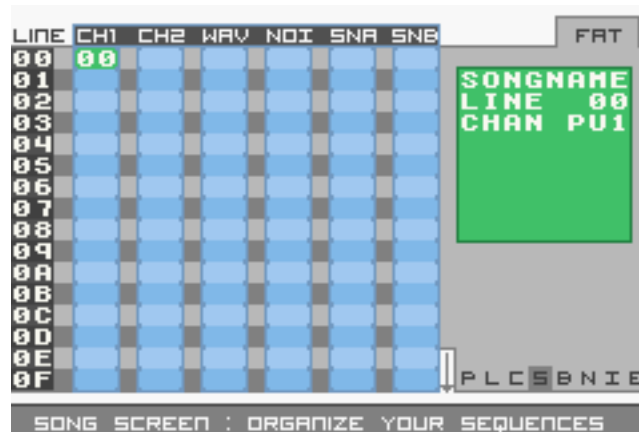
After having invested on these, all you need to do is copy the file fat.gba to the SD card.

*Notice: this version of **FAT** is experimental. It's unwise to spend a lot of money in order to test on a GBA. Have patience.*

1.4 A first sound

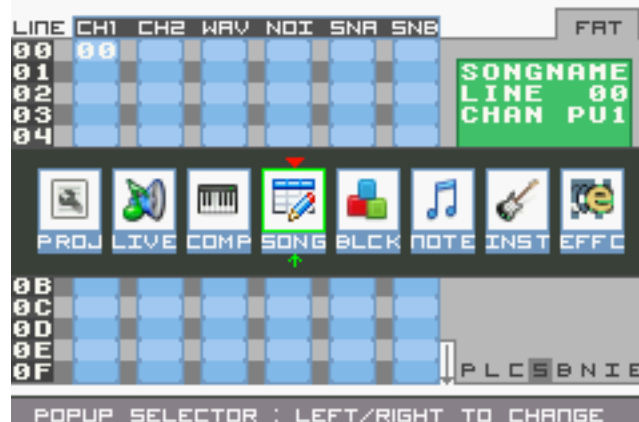
Let's start simple! We are going to create our first sound in **FAT**.

After starting **FAT** up, you will see the Song screen. Don't move the cursor. Just press A once. A "00" will appear in the top left part of the table: that's a sequence.



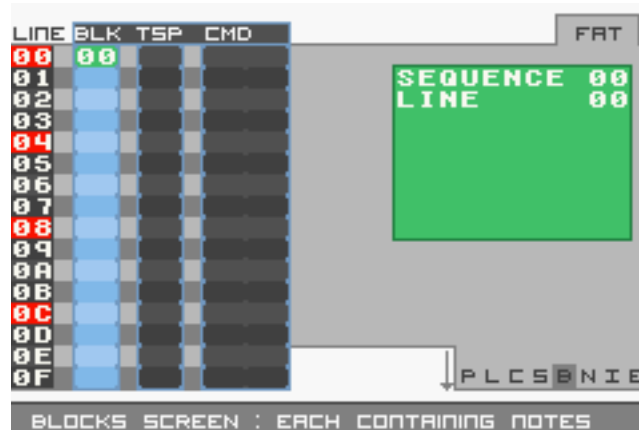
The Song screen

Next press and hold SELECT and move RIGHT once. Release SELECT. The interface will change.



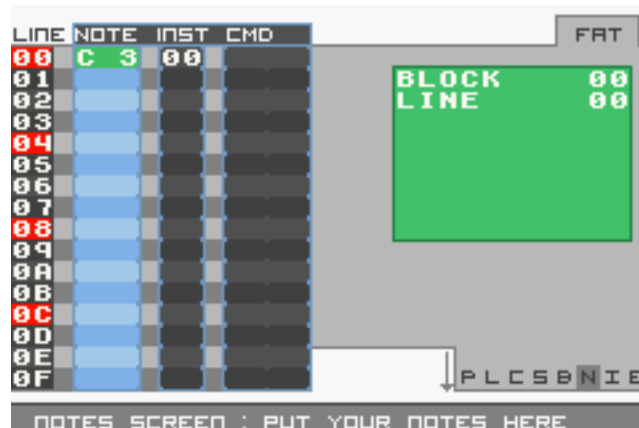
The Screen menu

Again without moving the cursor, press A once: "00" appears once again. This time it represents a block.



Block screen

Press and hold SELECT again and go RIGHT again. Release SELECT and the interface changes. Press A once and you will see "C3" appear. This is your first note! Press START and voila your first sound in **FAT**! Press START again to stop playing.



Notes screen - Press START!

1.5 Data structure

We have approached the concepts of "sequences", "blocks" and "notes". **FAT** is, in general terms, organized the same way as LSDJ:

- 6-column table of sequences: the GBA can play 6 sounds at once.
- Each sequence there's a maximum of 16 blocks.
- Each block contains up to 16 notes.

The way of organizing a song resembles a Russian doll. Sequences contain blocks which in turn contain notes. When **FAT** reads a sequence, it goes from one block to the other and reads all the notes one by one.

To avoid every note sounding the same, we separate them in "instruments". Each instrument contains a certain number of parameters. If you change the values of these parameters, the note will sound differently. You also need to know there are several types of instruments: the sound channels in the GBA allow us to do different things to each of them.

1.6 Gameboy Advance technical data

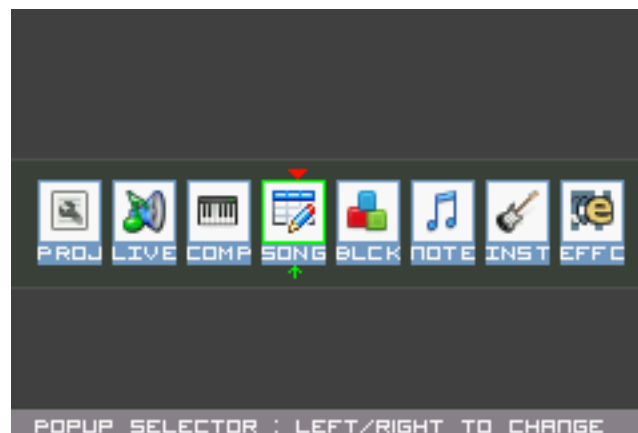
This part of the documentation is still in the works

2 Screens

To organize all the data in a song, **FAT** is divided in several screens. Each screen has its own useful features.

2.1 Screen menu

The screen menu (always accessible through the SELECT button) allows to change the currently active screen.



The screen menu (isolated in this image)

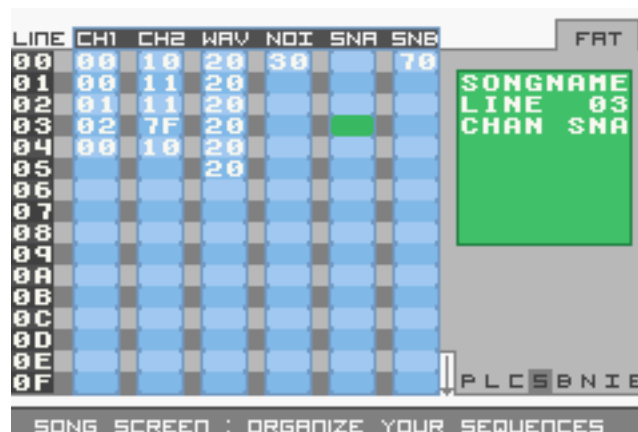
Simply imagine this menu as a "task bar". Change the screen you're on by moving LEFT or RIGHT and releasing SELECT to enter the desired screen.

2.1.1 Commands

SELECT (hold)	Show the menu
SELECT (release)	Validate and select a screen
LEFT / RIGHT	Move the selection cursor

2.2 Song screen

This screen serves as a "sequencer". It is here where you will organize all your sequences. Once the playhead is in motion, **FAT** reads all the sequences line by line. When it finds an empty space, **FAT** will try to go back to the first line.



The song screen shows a large table

2.2.1 Commands

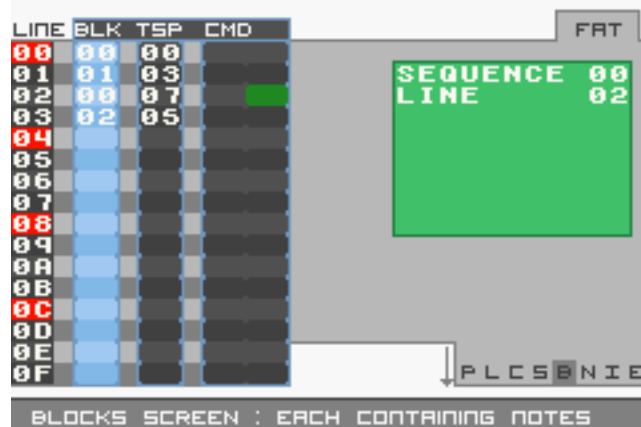
SELECT	Show the screen menu
START	Start stop the playhead from the cursor's position
L + UP	Go to start

L + DOWN	Go to the end
R + UP	Page up
R + DOWN	Page down
A	Put the current sequence in the selected cell (defaults to 00)
A + DIRECTION	Change the current sequence and put it in the selected cell
L + A	Find the next available sequence and put it in the selected cell
R + A	<i>Currently unused</i>
B	If the cell is not empty, cut its contents
B	If the cell is empty, paste
L + B	If the cell is empty, a previously cut sequence and increase its number
L + B	If the cell is not empty, increase its number and copy its contents

2.3 Blocks screen

In this screen you can edit the contents of a sequence: each sequence can contain from 0 to 16 blocks. When **FAT** plays the sequence, it will start reading the first block and go down until the last empty space. Then **FAT** will go back to the beginning. If there are no blocks in this sequence, **FAT** will skip it.

To simplify, imagine this screen as a list of blocks contained in a sequence (the number of the sequence is shown in the right side of the screen).



Blocks screen

2.3.1 Commands

SELECT	Show the screen menu
START	Start stop the playhead within the sequence from the cursor's position
L + UP	Go to start
L + DOWN	Go to the end
A	Put the current block in the selected cell
A + DIRECTION	Change the block number in the selected cell
L + A	Find the next available block number and put it in the cell
R + A	<i>Currently unused</i>
B	If the cell is not empty, cut the block
B	If the cell is empty, paste a previously cut block
L + B	If the cell is not empty, increase the block number and copy the cell's contents

L + B	If the cell is empty, paste the previously cut block and increase its number
R + LEFT / RIGHT	Change the sequence being edited
R + UP / DOWN	Jump to the next or previous sequence in the sequence list (song screen)

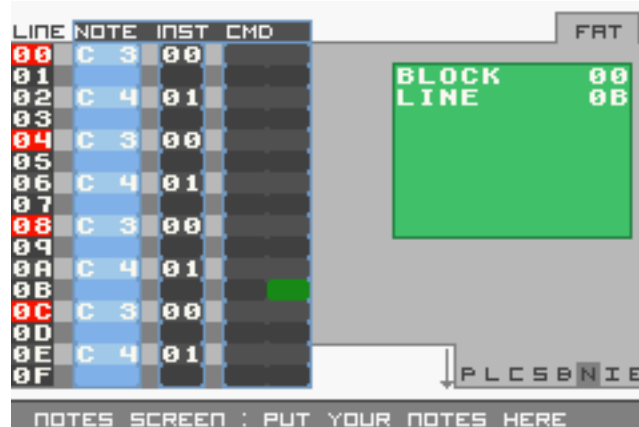
2.4 Notes screen

In this screen you will write... your notes! You can only write one note per line. A block contains 16 lines, so you can write up to 16 notes.

The anglo-saxon notation is used to show the notes. So:

DO C
RE D
MI E
FA F
SOL G
LA A
SI B

Each note is linked to an instrument (the INST column). Each instrument possesses a certain number of parameters which allow us to make the note sound in a certain manner. We will see in the next section how to set up our instruments.



Notes screen

2.4.1 Commands

SELECT	Show the screen menu
START	Start stop the playhead within the block from the cursor's position
L + UP	Go to start
L + DOWN	Go to the end
A	Write a note (C3 by default)
A + UP / DOWN	NOTE COLUMN – Change the octave for note under the cursor
A + LEFT / RIGHT	NOTE COLUMN – Change the tone of the note under the cursor
A + DIRECTION	INSTRUMENT COLUMN – Change the instrument number used for the selected note
A + DIRECTION	CMD COLUMN – Change the command value
L + A	INSTRUMENT COLUMN – Find a new instrument number and write it
R + A	Currently unused

R + LEFT / RIGHT	Change the block being edited
R + UP / DOWN	Jump to the next or previous block in the current sequence
B	If the cell is not empty, cut the note
B	If the cell is empty, paste a previously cut note

2.5 Instrument screen

This screen is divided in 4 tabs: there are 4 types of instruments in **FAT**.

- The Pulse type
- The Wave type
- The Noise type
- The Sample type

*Note that this version of **FAT** does not implement the Sample type yet. In the future, this category will likely be renamed or subdivided in 2 parts: the samples included in the ROM and a synthesizer.*

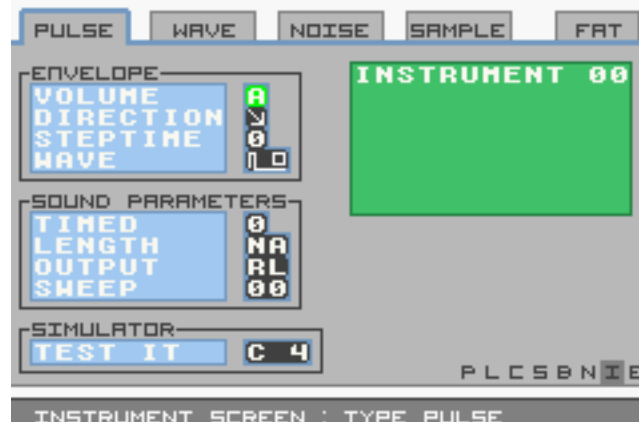
Each instrument's notes will be read **only on a specific channel**. Remember the Song screen contains 6 columns. These 6 columns represent in fact each of the 6 channels of the GBA. So:

- Pulse instruments are designated to channels 1 and 2: CH1 and CH2
- Wave instruments are used in channel 3 only: WAV
- Noise are used in channel 4: the NOI channel
- Finally, Sample instruments work on channels 5 and 6: SNA and SNB

2.5.1 Commands

SELECT	Show the screen menu
START	Start stop the playhead within the currently being edited block
A + DIRECTION	Change the values under the cursor
R + LEFT / RIGHT	Change the current instrument
NOTE THIS IS BUGGY AT THE MOMENT	
L + LEFT / RIGHT	Change the instrument type

2.5.2 Pulse type instrument



Pulse instrument

Envelope volume: From 0 to F. This value allows to change the volume of the instrument's notes.

- 0: Mute
- F: Full volume!

Envelope direction: Arrow up or down represents if the note is ascending or descending.

Envelope steptime: From 0 to 7. Determines the length of the wave (roughly).

- 0: Parameter will be ignored
- 1: Shortened
- 7: Very short

Envelope wave: 0 to 3. Modifies the shape of the wave played by the Gameboy Advance.

Sound parameters timed: 0 or 1.

- 0: The sound is not timed
- 1: The sound is timed

Sound parameters soundlength: From 0 to 3F. This parameter is not accessible unless “Timed” is set to 1.

- 0: The sound is “infinite”
- 3F: The sound is very (very... a.k.a “too”) short

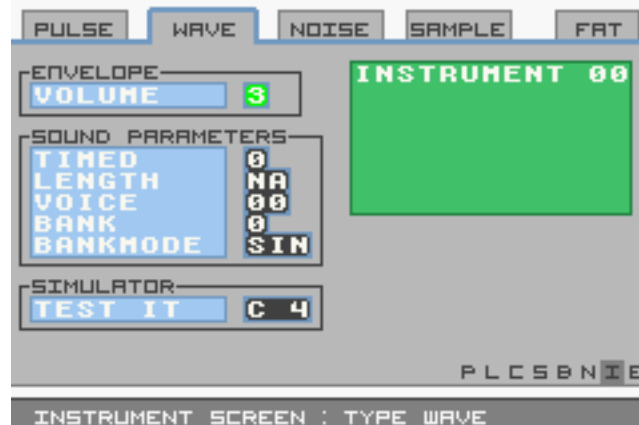
Parameter output: This parameter is currently unused.

Parameter sweep: From 0 to 7F. This adds a given effect to the note (still to be tested).

Simulator: This feature is not ready yet. It’s purpose will be to test directly a note with the currently selected instrument parameters. This space will allow to change the tone of the test note.

2.5.3 Wave type note

This instrument type is not stable enough yet: it can yield unexpected results for the time being. All this will be fixed in coming versions.



Wave type

Envelope volume: From 0 to 4. This value allows to adjust the volume of the instrument's notes.

Sound parameters timed: 0 or 1.

- 0: The sound is not timed
- 1: The sound is timed

Sound parameters length: From 0 to FF. This parameter is only accessible if “Timed” is set to 1.

- 00: The sound is “infinite”
- EB: The sound begins to be short
- FF: The sound is very (very very) short

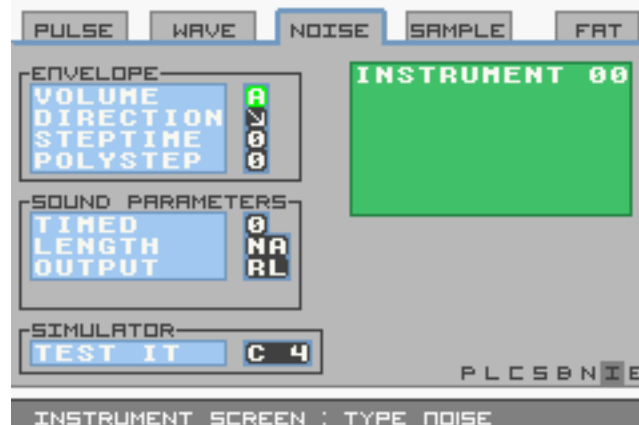
Sound parameters voice: From 0 to 17. **FAT** boasts 23 types of WAV instruments (in hexadecimal 17=23). For the time being, these instruments are not customizable. They will be in next versions.

Sound parameters bank: 0 or 1. The WAV channel in the GBA is capable of loading 2 banks into memory. You can decide which bank will be played. Note that the bank will be directly loaded depending on the “Voice” parameter above (each “Voice” contains 2 banks).

Sound parameters bankmode: “SIN” or “DUA”. The WAV channel is capable of synthesizing both banks in memory (DUA = Dual) or just one of them (SIN = Single). Note that if you set this parameter to SIN, **FAT** will use the bank selected in the “Bank” parameter above.

Simulator: This feature is not ready yet. It's purpose will be to test directly a note with the currently selected instrument parameters. This space will allow to change the tone of the test note.

2.5.4 Noise type instrument



Noise type

Envelope volume: From 0 to F. This value adjusts the volume of this instrument's notes.

- 0: Mute
- F: Full volume!

Envelope direction: Arrow up or down represents whether the sound is ascending or descending.

Envelope steptime: From 0 to 7. Determines the length of the wave (roughly).

Envelope polystep: 0 or 1. Modifies the manner in which the noise generator works.

- 0: Simple noise
- 1: Grinding noise

Try it out!

Sound parameters timed: 0 or 1.

- 0: The sound is not timed
- 1: The sound is timed

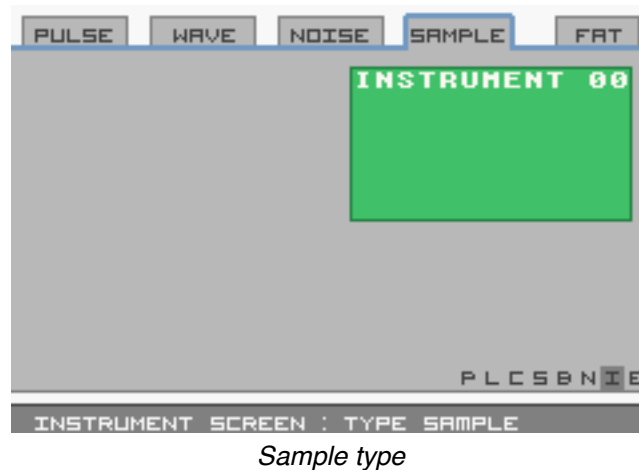
Sound parameters length: From 0 to FF. This parameter is only accessible if "Timed" is set to 1.

- 00: The sound is “infinite”
- EB: The sound begins to be short
- FF: The sound is very (very very) short

Parameter output: This parameter is currently unused.

Simulator: This feature is not ready yet. It’s purpose will be to test directly a note with the currently selected instrument parameters. This space will allow to change the tone of the test note.

2.5.5 Sample type instrument



This instrument type it's not available yet. In a future version you will be able to create here instruments divided in sample collections or instruments that will allow to make FM synthesis.

2.6 Project screen

The Project screen is used to manage the properties of your song. Right now, only the “Transpose” and “Tempo” properties are functional.



The project screen – not much at the moment

2.6.1 Commands

SELECT	Show the screen menu
A + DIRECTION	Change the value under the cursor
START	Start / stop the playhead as in the sequencer

2.7 Effects screen

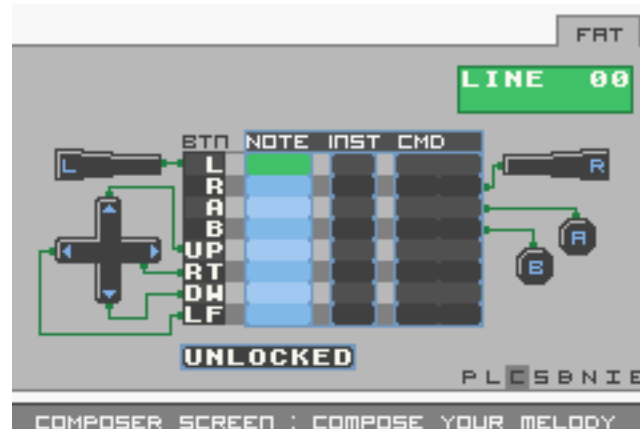
This screen is not programmed yet. For those who are familiar with LSDJ, it will be equivalent to the TABLES.

2.8 Live screen

The Live mode is not available yet. This mode will allow to tamper with your song in real time to make it sound never the same.

2.9 Composer screen

Imagine this screen as a mini-piano: the Composer will transform your Gameboy Advance into a sound improvisation machine!



Composer screen

The principle is simple. This screen shows 8 notes. Each of these notes is bound to a button (only the SELECT and START buttons are not used). If you look to the bottom side of the screen you will see the word “UNLOCKED”: this means you are in edit mode and you can write notes as you would in a Block (with A).

Once you have written all your notes (which can each have a different instrument), press START. The mode will be set to “LOCKED”.

Press a button: the assigned note is played in real time. Press another button for another note. Simple, isn't it?

Also note that this data will be saved for each of your songs. You can prepare your set of notes at home and jam live without effort!

Try it! Compose some sequences and then try to play them in Composer at the same time!

2.9.1 Commands

SELECT	Show the screen menu
START	Change the mode (UNLOCKED / LOCKED)
BUTTONS	LOCKED – Play the corresponding note
A + DIRECTION	UNLOCKED – Change the value for a note / instrument

3 Effects and commands

I'm sorry! There are no effects nor commands in this version. You will have to manage with instrument parameters.