

# 5TC option AUD

## Introduction to Embedded Audio Programming

Romain Michon, Tanguy Risset

Labo CITI, INSA de Lyon, Dpt Télécom, GRAME-CNCM



GRAME  
CENTRE NATIONAL  
DE CRÉATION  
MUSICALE, LYON

14 septembre 2023

# Table of Contents

AUD (Embaudio) introduction

Emeraude short presentation

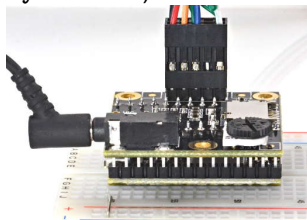
Embedded systems ?

Teensy presentation

Installing Teensy programming env.

# AUD objectives

- Understand basics of **streaming/dataflow programming** (callbacks)
- Understand basics of **embedded micro-controllers** (bare metal, cross-compilation, interrupts, timers, etc.)
- Understand basics of **digital audio programming**.
- Hands-on on a real embedded system : **teensy 4.0** (<https://www.pjrc.com/store/teensy40.html>)



# AUD planning

- AUD web site : <https://embaudio.grame.fr/>
- AUD github : <https://github.com/grame-cncm/embaudio>
- Course organization :
  - $\simeq$  6 course on embedded audio basics on teensy
  - $\simeq$  2 course on embedded systems basics
  - $\simeq$  2 external talks
  - $\simeq$  4 course on project mode + demo presentation

# Table of Contents

AUD (Embaudio) introduction

Emeraude short presentation

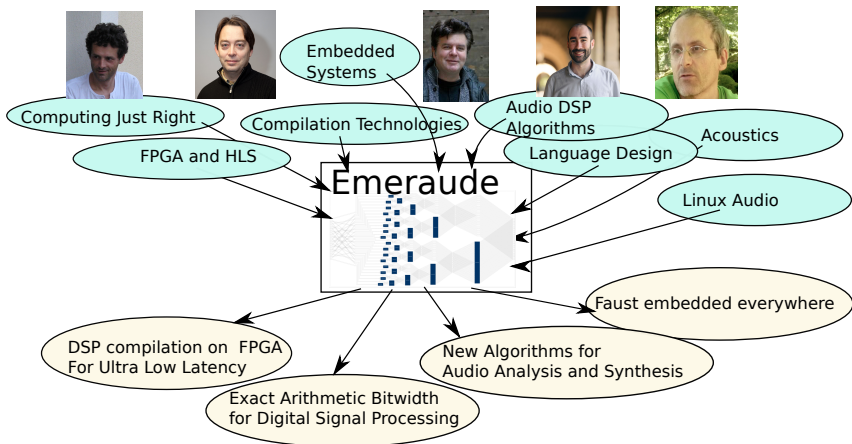
Embedded systems ?

Teensy presentation

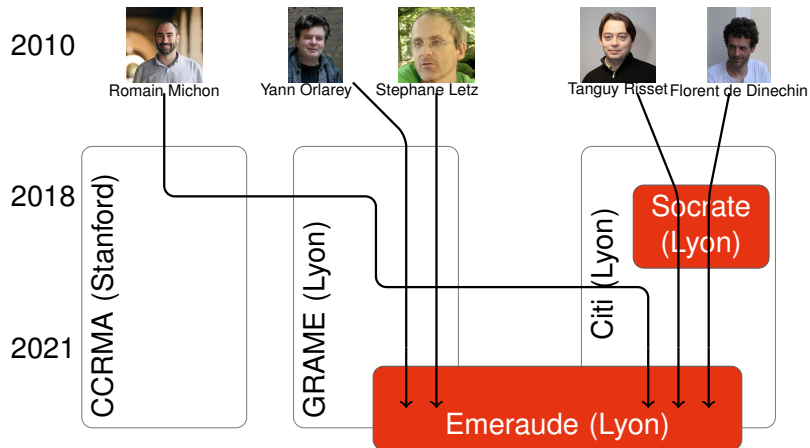
Installing Teensy programming env.

# Emeraude : new Citi Team (2022)

A collaboration between Grame, Insa and Inria



# A focus on Emeraude's origins



## Link with TC

- 2020 creation of 5TC-AUD : Audio on ESP32 LyraT
- 2022 creation of 3TC-SON (projet audio embarqué) based on 5TC-AUD experience.
- Connexion with audio SMEs and academic universities.
- Possible start-up creation



# Table of Contents

AUD (Embaudio) introduction

Emeraude short presentation

Embedded systems ?

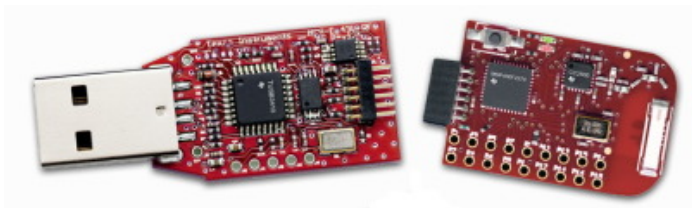
Teensy presentation

Installing Teensy programming env.

# Different types of embedded systems

- Micro-controllers and sensor networks : (example : MSP430)
- Embedded computing devices (Sink nodes, phones, tablets, example : raspberryPi)
- Micro-contrôleur 32 bits with a bunch of RAM (example : Teensy 4.0)

## Example of Small Embedded System



- **eZ430-RF2500** with a MSP430f2274 and a radio chip CC2500
- MSP430 ( $\simeq 1\text{€}$ )
- 16 bits processor, 16Mhz maximum
- 64KB of addressable memory, 1KB RAM
- Usual micro-contrler peripherals
- No MMU
- Low power design

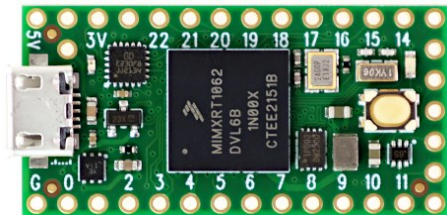
# Example of More powerful Systems

- **Beagleboard** ( $\simeq 60\text{€}$ )
  - Arm Cortex A8 1 GHz
  - DaVinci SoC ARM+DSP
  - Puce graphique 3D
  - 512 MB of DDR SDRAM
  - 4GB SD-Card
  - DVI-D, S-Video, 4 port USB Hub, Stereo In/Out, Ethernet 10/100...
- **Raspberry Pi** ( $\simeq 25\text{€}$ )
  - Broadcom BCM2835, 700 MHz ARM avec FPU
  - GPU Videocore 4
  - RAM 512 Mo.
  - 4GB SD-Card
  - video RCA 2 port USB Hub, Stereo Out, Ethernet 10/100...



## Teensy : intermediate system

- **Teensy 4.0** ( $\simeq 25\text{€}$ , before crisis)
- ARM Cortex M7 (600MHz)
- 1 MB of RAM for execution
- 2 MB of flash memory for storing program
- Audio shield available
- Usual micro-controller peripherals
- No other connectivity (Wifi, Bluetooth, ethernet etc) : need additional boards



# Table of Contents

AUD (Embaudio) introduction

Emeraude short presentation

Embedded systems ?

Teensy presentation

Installing Teensy programming env.

# Teensy presentation

## Welcome to Teensy® 4.0

32 Bit Arduino-Compatible Microcontroller

To begin using Teensy, please visit the website & click [Getting Started](http://www.pjrc.com/teensy).

[www.pjrc.com/teensy](http://www.pjrc.com/teensy)

**Digital Pins**  
digitalRead  
digitalWrite  
pinMode

Red: LED  
Green: Leds  
Blue: Status  
Orange: Ready  
Purple: Write  
Pink: Mode  
Black: No USB

**Analog Pins**  
analogRead

**PWM Pins**  
analogWrite

**Digital Audio**  
Audio Library

**Serial Ports**  
Serial - Serial7

**I2C Port**  
Wire Library

**SPI Port**  
SPI Library

**CAN Bus**  
FlexCAN, M Library

Min (3.6 to 5.5 volts)  
3.3V (200 mA max)  
GND

On/Off Program  
GND  
3.3V  
VBat

All digital pins have 10mA current capability.

## Teensy® 4.0 Back Side

Additional pins and features available on the back side

PWM SCL2 TX16 A10 24 25 A11 RX6 SOA2 PWM  
MOSI1 A12 26 27 A13 SCK1  
RX7 28 29 TX7 PWM  
CRX3 30 31 CTX3  
OUT1B 32 33 MCLK2 PWM

On/Off Program GND 3.3V VBat

Use 3V coin cell for Date & Time and power management features

SD Card (4 bit SDIO)  
SD Library  
SD.begin(BUILTIN\_SDCARD)

Teensy 4.0 signal pins are not 5V tolerant. Do not apply more than 3.3 volts to any pin, except VIn or VUSB.

USB Host  
USBHost\_L39 Library

34 DAT1 MISO2 PWM  
35 DAT0 MISO1 PWM  
GND  
36 CLK CS2 PWM  
37 CMD SCK0 PWM  
38 DAT3 R15 PWM  
39 DAT2 TX5 PWM

Cut to separate VIn from VUSB, if using battery or external power.

VUSB  
D-  
D+

For solutions to the most common issues and technical support, please visit:  
[www.pjrc.com/help](http://www.pjrc.com/help)

Teensy 4.0 System Requirements:  
PC computer with Windows 7, 8, 10 or later  
or Ubuntu Linux 14.04 or later  
or Macintosh OS X 10.8 or later  
USB Micro-B Cable

- 40 input/output signal pins.
- 24 accessible with a solderless breadboard.
- Many serial protocols (I2C, I2S, CAN, SPI and UART)
- 1 MB of RAM for execution
- 2 MB of flash memory for storing program

# PRJC web site

- Teensy has been proposed by Paul Stoffregen (https://www.kickstarter.com/projects/paulstoffregen/teensy-35-and-36)
- It is compatible with arduino compilation suite (most powerful processor targeted by arduino).
- Most teensy software and hardware design are freely available on PRJC website :  
https://www.pjrc.com/store/teensy40.html

The screenshot shows the PRJC website's product page for the Teensy 4.0 Development Board. The page is titled "Teensy® 4.0 Development Board" and features a navigation menu with links for Home, Products, Teensy, Bix, and Forum. A sidebar on the left contains a tree view of the product line, including Teensy 3.2, Teensy 4.0, and various boards like the Teensy 4.0 USB Dongle and Teensy 4.0 USB Dongle (with USB-C). The main content area includes a table of products, update notices, and a list of recommended accessories.

Temporarily Out Of Stock	TEENSY40	Teensy USB Board, Version 4.0 For prototypes, experimentation, and learning	\$23.80
Temporarily Out Of Stock	TEENSY40_LOCK	Lockable Teensy USB Board, Version 4.0 For commercial products and secure applications. <a href="#">See Code Samples for Lockable Teensy Details</a>	\$23.80

**Recommended Accessories:** [USB Cable - Pro 14x1 \(7\)](#)

**Updates:**  
**November 7, 2022:** Chips ordered from TSMC and Teensy 4.0 is in production. A small number will be completed on an expedited schedule by November 22. The bulk of this production batch will arrive between December 12 to 23. Even more chips are ordered and we expect to have even more Teensy 4.0 throughout January.  
**September 7, 2022:** More Teensy 4.0 will complete production next week, but all will be sold by distributors. [More Details on this Update](#).  
**June 21, 2022:** Chip shortages are anticipated to impact Teensy 4.0 more than Teensy 4.1 between August to December 2022. If you are developing a project which will need more Teensy 4.0 boards before 2023, we recommend testing with a spare and other measures to allow a Teensy 4.1 to be used as a substitute. In case Teensy 4.0 is in short supply later this year.

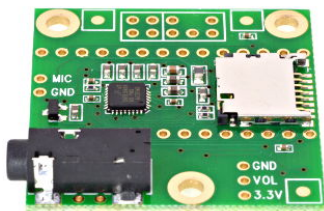
**Inventory Status:** **Out Of Stock**  
Last physical count: Oct 18, 2022

**Sections On This Page:**  
[Photos](#) - [Specifications](#) - [Software](#) - [Processor](#) - [Pin](#) - [Digital Pins](#) - [Analog Pins](#) - [Communication](#) - [Drivers](#) - [Audio](#) - [LEDs](#) - [Energy](#) - [Power](#) - [Memory](#) - [Programming](#) - [Code Security](#) - [Special Features](#) - [Technical Information](#)

**Photos**



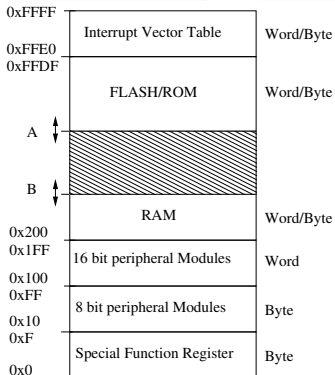
# Teensy audio shield



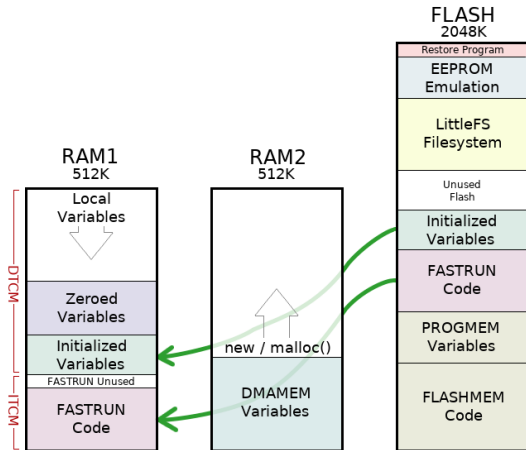
- Audio adaptor board provided by PJRC (soldered by profs)
- Low power stereo codex (NXP Semiconductors SGTL5000 codec) and a SD card reader.
- [https://www.pjrc.com/store/teensy3\\_audio.html](https://www.pjrc.com/store/teensy3_audio.html)
- the I2C (or I<sup>2</sup>C : Inter-Integrated Circuit) protocol is used to configure the codec (sample rate, input and output pins etc.)
- the I2S (or I<sup>2</sup>S : Inter-IC Sound) is used to transfer samples bit by bit in both direction (i.e. from and to the teensy)
- The audio shield comes with the Teensy Audio Library ([https://www.pjrc.com/teensy/td\\_libs\\_Audio.html](https://www.pjrc.com/teensy/td_libs_Audio.html))

# Reminder : MSP439F2274 Memory Map (64KB)

- 0x0000 to 0x01FF : peripherals
- 0x0200 to B=0x05FF : RAM (1KB), Data and Stack
- 0x0C00 to 0x0FFF : Boot mem (1KB, ROM).
- 0x1000 to 0x10FF : byte info. mem. (256 bytes, Flash)
- 0x8000 to 0xFFFF : Code (32 KB, Flash).
  - where : 0xFFE0 to 0xFFFF : interrupt vectors

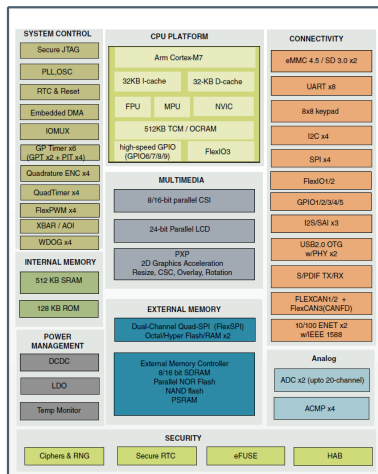


# Teensy Mempoyp Map (1MB address space)



- RAM1 is accessed as tightly coupled memory
- RAM2 is accessed with a DMA (for large arrays)
- Pragmas in variables declaration (e.g. DMAMEM) for mapping variables to memory zones.

# Teensy processor : NXP i.MX RT1062



- CPU : ARM Cortex-M7 (ARMv7)
- dual-issue superscalar processor (2 instructions per cycle)
- Hardware Floating Point Unit
- 64 bit ICTM (RAM1)
- Data and Instr. caches (RAM2)
- DSP extensions (SIMD, MAC, ...)
- USB used to
  - program (JTAG)
  - Serial communication (UART/Serial, Midi, mouse, etc)

# Table of Contents

AUD (Embaudio) introduction

Emeraude short presentation

Embedded systems ?

Teensy presentation

Installing Teensy programming env.

# Teensyduino install

- Install arduino from here :  
`https://www.arduino.cc/en/software.`  
(version 1.8.19 recommended)
- Install teensyduino here :  
`https://www.pjrc.com/teensy/td_download.html`
- Clone embaudio git here :  
`https://github.com/grame-cncm/embaudio`
- Copy \$EMBAUDIO/examples/teensy/libraries/mydsp to  
\$ARDUINOPATH/librairies
- In arduino :
  - Select :  
Tools -> Board Manager -> Teensyduino -> Teensy 4.0  
Tools -> USB Type -> Serial

# Au Boulot...



The screenshot shows the Arduino IDE interface. The title bar reads "EchoBoth | Arduino 1.8.19". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". The toolbar contains icons for saving, undo, redo, and zooming. The sketch editor displays the following code:

```
EchoBoth

/* UART Example, any character received on either the real
  serial port, or USB serial (or emulated serial to the
  Arduino Serial Monitor when using non-serial USB types)
  is printed as a message to both ports.

  This example code is in the public domain.
  */

// set this to the hardware serial port you wish to use
#define HWSERIAL Serial1

void setup() {
  Serial.begin(9600);
  HWSERIAL.begin(38400);
}

void loop() {
  int incomingByte;

  if (Serial.available() > 0) {
```

The status bar at the bottom indicates "Teensy 4.0, Serial, 600 MHz, Faster, US English on /dev/ttyACM0".