

# Les microcontrôleurs

## Choisir un microcontrôleur

Nous avons plusieurs références de microcontrôleurs (MCU) au lab et pour mieux choisir voici un tableau comparatif de leurs caractéristiques :

	ATTiny44A	ATTiny45	ATMega328P	ATTiny412	ATTiny1614	ATTiny3216	AVR128DB32	ATSAMD11C14U	ATSAMD11D14U	ATSAMD20E17A	ATSAMD21E15	STM32F051K6T6	STM32F030F4P6	STM32F042F4P6
Tension d'alimentation	2.7-5.5V	2.7-5.5V	2.7-5.5V	1.8-5.5V	1.8-5.5V	1.8-5.5V	1.8-5.5V	1.6-3.6V	1.6-3.6V	1.6-3.6V	1.6-3.6V	2.4-3.6V	2.4-3.6V	2-3.6V
Architecture	AVR	AVR	AVR	AVR	AVR	AVR	AVR	ARM	ARM	ARM	ARM	ARM	ARM	ARM
Debug	ISP	ISP	ISP	UPDI	UPDI	UPDI	UPDI	SWD	SWD	SWD	SWD	SWD	SWD	SWD
Fréquence d'horloge	20MHz	20MHz	20MHz	20MHz	20MHz	20MHz	24MHz	48MHz	48MHz	48MHz	48MHz	48MHz	48MHz	48MHz
Flash	4K	4K	32K	4K	16K	32K	128K	16K	16K	128K	32K	32K	16K	16K
SRAM	256B	256B	2K	256B	2K	2K	16K	4K	4K	16K	4K	8K	4K	6K
GPIO	12	6	23	6	12	18	26	12	18	26	26	25	15	16
PWM	2	2	6	1	1	1	4	4	4	8	12	6	6	6
ADC	4	4	8	1	2	2	13	10	10	20	20	16	11	12
DAC	-	-	-	1	3	3	1	1	1	1	1	1	-	-
I2C	1	1	1	1	1	1	2					1	1	1
SPI	1	1	1	1	1	1	2	3	3	6	6	1	1	1
UART	1	1	1	1	1	1	3					2	1	2
Capacitive touch	-	-	-	-	6	12	-	12	18	26	26	13	-	14
USB	-	-	-	-	-	-	-	x	x	x	x	-	-	x
RTC	-	-	-	x	x	x	x	x	x	x	x	x	x	x
CAN	-	-	-	-	-	-	-	-	-	-	-	-	-	1
Compatibilité Arduino	+++	+++	+++	++	++	++	++	+	+	-	-	?	?	+

Attention, tous les microcontrôleurs ne sont pas parfaitement pris en charge par l'environnement Arduino et les bibliothèques associées.

Si votre projet nécessite une connectivité WiFi ou Bluetooth il faudra opter pour un ESP32. Ils ne sont pas représentés dans le tableau car ils sont hors catégorie. Ils sont plus puissants et ont beaucoup plus de mémoire. On peut d'ailleurs les utiliser avec Micro Python.

**Les microcontrôleurs sont une denrée rare et précieuse** actuellement, donc choisir forcément celui qui est le plus puissant ou avec le plus d'E/S est une mauvaise idée. Il faut réfléchir en amont aux caractéristiques du projet et prendre celui qui est le plus adapté.

Les ATTiny44, 45 et les ATMega328P sont des technologies un peu datées (8 bits) mais suffisamment puissantes pour de nombreux projets. Ils ont l'avantage d'être parfaitement pris en charge par l'environnement Arduino. Le protocole utilisé pour uploader du code est toutefois moins

pratique à mettre en oeuvre.

Pour un novice, les MCU les plus polyvalents et les plus faciles à utiliser sont les ATTiny412, 1614 et 3216. Le protocole UPDI pour charger le code est facile à mettre en oeuvre, ils sont relativement puissants et sont compatibles avec la grande majorité des bibliothèques Arduino.

Les SAMD sont plus puissants (32 bits) et ont l'avantage de gérer la connexion USB en natif, mais leur prise en charge par Arduino n'est pas encore optimale.

Les STM32 sont des microcontrôleurs très puissants mais plus complexes à utiliser, car ils nécessitent d'utiliser l'environnement de développement de STM et ses bibliothèques. On a plus de possibilités mais au prix d'une plus grande complexité. Réservé aux utilisateurs aguerris.

## Exemples de projets et documentation

- ATTiny44A ([datasheet](#))
  - [Blinky and button](#)
  - [Phototransistor](#)
- ATTiny45 ([datasheet](#))
- ATmega328P ([datasheet](#))
- ATTiny412 ([datasheet](#))
  - [RGB LED](#)
- ATTiny1614 ([datasheet](#))
- ATTiny3216 ([datasheet](#))
  - [RGB LED](#)
- AVR128DB32 ([datasheet](#))
- ATSAMD11C14U ([datasheet](#))
  - [Blinky](#)
  - [Miniduino](#) (development board)
- ATSAMD11D14U ([datasheet](#))
  - [Development board](#)
- ATSAMD20E17A ([datasheet](#))
- ATSAMD21E15 ([datasheet](#))
- ESP32

- [Development board](#)
- [STM32F051K6T6 \(datasheet / reference manual\)](#)
- [STM32F030F4P6 \(datasheet / reference manual\)](#)
- [STM32F042F4P6 \(datasheet / reference manual\)](#)

# Démarrer avec les ATtiny (1-series)

Quelques ressources pour démarrer avec les ATtiny412, 1614 ou 3216 :

- [Le programmeur / débogueur](#)
  -
- 

Revision #5

Created 20 September 2022 10:16:25 by Stephane Muller

Updated 27 September 2022 14:07:02 by Stephane Muller