

www.sylvainmahe.xyz

LE BLOG

de Sylvain Mahé

contact@sylvainmahe.xyz



Article: Sylvain Mahé

contact@sylvainmahe.xyz

[Retour](#)

[Suite](#)

Comment programmer avec MODULE

Pour programmer avec MODULE, vous avez à disposition **26 classes** organisées par fonctions (sous la forme de fichiers d'en-tête et de fichiers C++). Toutes ces classes sont à la fois complètement indépendantes les unes des autres (au niveau du code source C++) et peuvent aussi fonctionner toutes ensembles sans problèmes d'interactions.

Les classes de MODULE:

GpioRead.h
GpioWrite.h
AnalogRead.h
InterruptRead.h
PwmRead.h
PwmWrite.h
SoundWrite.h

Timer.h
Period.h
Delay.h

Math.h
Iteration.h
Average.h
Filter.h
Hysteresis.h
Random.h

Max7219.h
Mpu6050.h
Bno055.h
Hmc5883l.h
Bmp180.h
Nrf24l01p.h

Network.h

Memory.h
Power.h
Tool.h

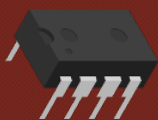
Pour comprendre comment utiliser les classes de MODULE:

- Vous pouvez lire les exemples que je vous propose (voir plus bas) qui s'étofferont au fil du temps (disponibles dans la section "Exemples simples à l'aide de MODULE" en page d'accueil).
- Vous pouvez copier le code source des mes projets.

Ou bien il est possible de regarder dans le répertoire **module** (dans l'archive de MODULE) le contenu des fichiers d'en-tête **.h** des classes correspondantes au microcontrôleur que vous souhaitez programmer, ce qui vous donnera une bonne idée de comment utiliser les fonctions et variables si vous êtes déjà bien familiarisé à la programmation C++.

Les exemples pour l'ATmega1284P:

Historiquement, j'ai débuté la programmation de MODULE avec l'ATmega328P et finalisé un certain nombre de projets avec. Mais depuis j'ai conçu un automate programmable plus intéressant sur bons nombres de points qui peut être équipé (entre autres) de l'ATmega1284P (que j'utilise beaucoup), ce qui explique que les exemples que vous pouvez lire soient orientés vers ce microcontrôleur.



www.sylvainmahe.xyz

LE BLOG

de Sylvain Mahé

contact@sylvainmahe.xyz



[Retour](#)

[Suite](#)

*Tous les exemples proposés dans la section "Exemples simples à l'aide de MODULE" en page d'accueil sont écrits pour l'automate programmable **MODULABLE M32 équipé d'un ATmega1284P**, mais bien évidemment cela reste de votre choix de programmer pour l'un des 7 autres microcontrôleurs pris en charge par MODULE !*

Si par exemple vous souhaitez programmer pour le microcontrôleur ATmega328P plutôt que l'ATmega1284P, il vous suffit de modifier les chemins d'accès aux fichiers d'en-tête **1284p** par **328p** dans les exemples proposés, ainsi que d'adapter le numéro de certains ports de l'automate programmable par rapport au microcontrôleur choisi.

*Dans tous les cas, **vous disposez exactement des mêmes fonctionnalités avec MODULE** qu'il s'agisse de programmer l'ATmega48P, l'ATmega88P, l'ATmega168P, l'ATmega328P, l'ATmega164P, l'ATmega324P, l'ATmega644P ou l'ATmega1284P.*

Seul **le nombre de ports disponibles** par fonctionnalité et **leurs distribution sur les automates programmables** peut varier (c'est-à-dire les ports reliés aux fonctions USART, SPI, TWI, PWM, convertisseur analogique/numérique, le nombre de GPIO, etc...).

Libre à vous de choisir d'utiliser l'automate programmable **MODULABLE M20** qui peut être équipé des microcontrôleurs ATmega48P, ATmega88P, ATmega168P, ou ATmega328P, ou l'automate programmable **MODULABLE M32** qui peut être équipé des microcontrôleurs ATmega164P, ATmega324P, ATmega644P ou ATmega1284P.

Pour connaître précisément les caractéristiques de ces deux automates programmables et télécharger les plans de fabrication, je vous conseille d'aller voir dans les sections "Téléchargements" et "Fabrications et divers réalisations" en page d'accueil.

design du blog: sylvain mahé