



Prototypage IoT avec 6TRON

Partie 2 : Cas d'Étude

20 janvier 2025 – Thomas Habrant



Partie 2 : Cas d'Étude

Le Prototype

- Station Météo Connectée
- Cartes 6TRON utilisées

Microcontrôleurs

- MCU et MPU
- Arm
- Les Cortex-M
- Le STM32L4

Notions utiles

- Les GPIO
- Les IRQ
- Les Timers
- Principaux périphériques

Travaux Pratiques



Prototype : Station Météo Connectée

Objectif

Ajouter de fonctionnalités "intelligentes" à un objet simple

Mesurer

- Température ambiante
- Humidité relative
- Pression atmosphérique

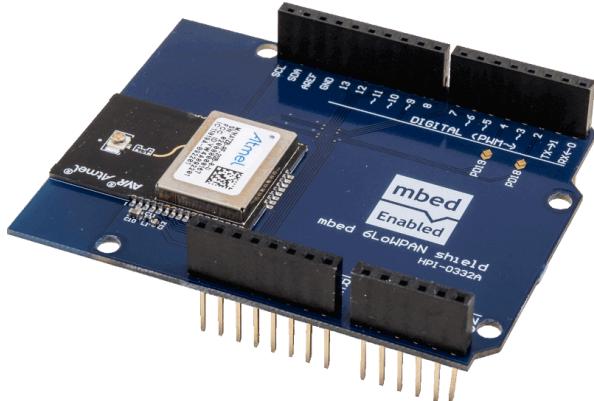
Et Augmenter

- Suivi à distance
- Actions automatisées et paramétrables (arrosage, lumières, etc.)
- Accès à des sources de données tierces (météo par ex.)



Prototype - Cartes 6TRON utilisées

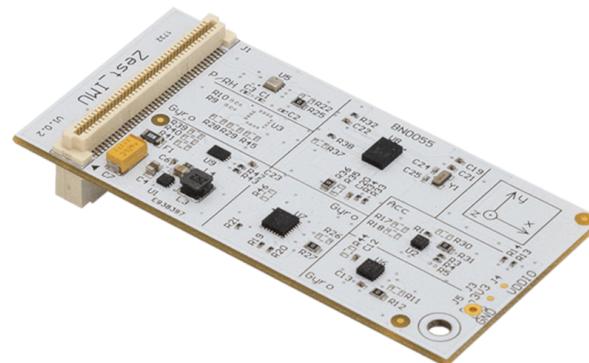
6LoWPAN Shield



Communication en 6LoWPAN

Réseau maillé IPv6

Zest IMU



Carte capteurs
Environnementaux
Accéléromètres
Magnétomètres
Etc.

Zest Core STM32L4



Carte microcontrôleur
Exécute l'application
Pilote le module radio
et les capteurs



Microcontrôleurs - MCU et MPU

MCU

- "Tout en un"
CPU, Mémoires, Périphériques, IOs
- Tout intégré
- Plus facile à intégrer
- Souvent moins cher
- Très rapide à démarrer (ms)
- Faible consommation
- Fréquence plus faible
- Capacités de calcul moindres

MPU

- Unité de calcul
- Mémoires et périphériques externes
- Intégration complexe
- Fréquence de calcul élevée
- Souvent multicœurs
- Puissance de calcul bien supérieure
- Utilisation possible d'un OS : Linux



Microcontrôleurs - MCU ou MPU ?

Dépend de la tâche à réaliser.

Microcontrôleur pour les tâches simples, **basse consommation**,
ou si besoin **temps réel** fort comme la sécurité ou le contrôle moteur

Microprocesseur pour des systèmes complexes, avec système d'exploitation
et/ou interfaces graphiques, comme le système multimédia d'une voiture, un
drone avec caméra, etc.



Microcontrôleurs - Arm

Société britannique fondée en 1990 spécialisée dans le développement de blocs de propriété intellectuelle (IP).

Fournit tout l'écosystème logiciel pour le développement de SoC et leur mise en œuvre.

En chiffres :

- Leader mondial sur les IP de semi-conducteurs
- 70% de la population mondiale est utilisatrice
- 90% des smartphones
- 125 milliards de composants vendus

Concurrents : Microchip (MCU), Intel/AMD (MPU)



Microcontrôleurs - Offre Arm

Cortex - A

Highest performance

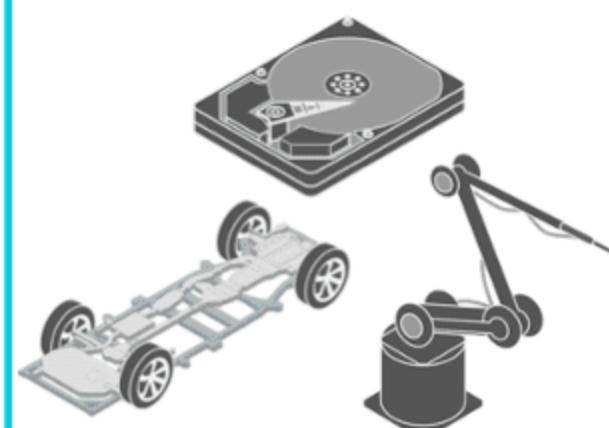
Optimised for
rich operating systems



Cortex - R

Fast response

Optimised for
high performance,
hard real-time applications



Cortex - M

Smallest/lowest power

Optimised for
discrete processing and
microcontrollers





Microcontrôleurs - Le STM32L4

Parallel Interface

FSMC 8-/16-bit
(TFT-LCD, SRAM, NOR, NAND)

Display

8 x 40 LCD driver

Timers

17 timers including:
2 x 16-bit advanced motor control
timers
2 x ULP timers
7 x 16-bit-timers
2 x 32-bit timers

I/Os

Up to 114 I/Os
Touch-sensing controller

ARM®
Cortex®-M4
80 MHz
FPU
MPU
ETM
DMA

ART Accelerator™

Up to
1-Mbyte Dual
Bank Flash
memory with ECC

Up to
128-Kbyte RAM

Connectivity

USB OTG,
1x SD/SDIO/MMC, 3 x SPI,
3 x I²C, 1x CAN, 1 x Quad-SPI,
5 x USART + 1 x ULP UART,
1 x SWP

Digital

AES (256-bit), TRNG, 2 x SAI, DFSDM
(8 channels)

Analog

3 x 16-bit ADC, 2 x DAC,
2 x comparators, 2 x op amps
1 x temperature sensor



Notions Utiles - Les GPIO

GPIO

- General Purpose Input/Output
 - Pins d'entrées et sorties des signaux du microcontrôleur
- Configurables par l'utilisateur
- En général, chaque pin peut être reliée à différents périphériques internes et donc avoir plusieurs fonctions

Dans Mbed OS, utiliser les classes `DigitalIn`, `DigitalOut`, etc., ou les classes dédiées à une fonction spécifique.



Interruption matérielle

- IRQ : Interrupt request
 - Interruption matérielle, déclenchée par un périphérique
- Interrompt le fonctionnement du MCU et appelle le gestionnaire d'interruption
- Permet la gestion d'évènements asynchrones
- Évite le polling

Dans Mbed OS, utiliser la classe `InterruptIn` pour surveiller le changement d'état d'une pin



Notions Utiles - Principaux périphériques

Timer

- Périphérique de comptage d'unités de temps
- Basé sur la fréquence interne du microcontrôleur
- Configurable (diviseur, PLL, seuil, etc.)
- Peut générer des interruptions

Dans Mbed OS, utiliser les classes `Timer` et `Ticker`



Notions Utiles - Principaux périphériques

ADC : Analog to Digital Converter

- Convertit un signal analogique (tension) en signal numérique
- Mesure comprise entre 0 V et la référence de tension,
souvent la tension d'alimentation : 3,3 V
- Résolution de 16 bits pour le STM32L4 : $3,3 / 2^{16} = 0,000050$ V

DAC : Digital to Analog Converter

- Génère un signal analogique à partir d'une valeur numérique

Dans Mbed OS, utiliser les classes `AnalogIn` et `AnalogOut`



Travaux Pratiques

Polling du bouton

- Lire l'état du bouton et afficher l'état du bouton sur la sortie standard
- Allumer la LED quand le bouton est appuyé, l'éteindre sinon

Utiliser une interruption

- Remplacer la lecture en polling par l'utilisation d'IRQ

Ajouter un Timer

- Compter et afficher la durée de l'appui sur le bouton

Ouvrir la liaison série pour visualiser les messages de la cible



Travaux Pratiques

Contrôle de la LED par un ticker

- Faire clignoter la LED à une certaine fréquence, **sans** boucle infinie

Piloter la fréquence de clignotement

- Utiliser le bouton pour changer la fréquence de clignotement à chaque appui

■ Ne pas oublier de créer des commits pour chaque étape !



Merci !

Des questions ?

t.habrant@catie.fr