

MINF
Mise en œuvre
des microcontrôleurs PIC32MX

Chapitre 1

Introduction

✂ T.P. PIC32MX

Christian HUBER (CHR)
Serge CASTOLDI (SCA)
Version 1.81 novembre 2021

CONTENU DU COURS T.P.

1. Introduction (ce chapitre)
2. Prise en main MPLABX, XC32 et Harmony
3. Prise en main MPLABX et ICD, programmation et debug
4. Gestion des E/S
5. Timers, interruptions et PWM
6. Utilisation des entrées de capture
7. Utilisation de l'USART
8. Gestion du bus I2C (machine d'état)
9. Gestion du bus USB
10. Gestion de l'Ethernet et de TCP/IP

CONTENU DE L'INTRODUCTION

<i>Remarque préliminaire</i>	<i>1</i>
1. Introduction	2
1.1. Pourquoi le PIC32MX	2
1.2. Apprentissage d'une nouvelle famille de microcontrôleurs	3
1.3. Développement et mise au point des programmes	4
1.4. Harmony : apports et contraintes	4
1.5. Le PIC32MX795F512L en bref	5
1.5.1. Schéma bloc	5
1.5.2. Boîtier et brochage	6
1.5.3. Boîtier et brochage de la version H	7
1.5.4. Quelques caractéristiques	8
1.5.5. USB, Ethernet et CAN selon modèle	9
1.5.6. Organisation de la mémoire	10
1.6. Documentations à disposition	11
1.6.1. Documentation PIC32MX	11
1.6.2. Documentation du Kit PIC32MX	11
1.6.3. Documentation compilateur et environnement	11
1.6.4. Documentation Harmony	12
1.6.5. Documentation dans l'environnement MPLABX	12
1.7. Historique des versions	12
1.7.1. V1.0 2013	12
1.7.2. V1.5	12
1.7.3. V1.6	12
1.7.4. V1.7 octobre 2016	12
1.7.5. V1.8 novembre 2017	12
1.7.6. V1.81 novembre 2021	12

REMARQUE PRÉLIMINAIRE

Le présent cours est illustré à travers l'utilisation des outils de Microchip, dont notamment le framework Harmony. Au fil de l'évolution des versions, il est possible que quelques différences existent entre la version utilisée lors de l'élaboration et la version actuelle lors de la lecture. Les principes présentés restent applicables.

Dans la mesure du possible, la version utilisée pour les exemples sera indiquée.

Le lecteur est invité à se reporter à la documentation de sa version disponible sous :

<Répertoire Harmony>\v<n>\doc

1. INTRODUCTION

Ce cours a pour objectif d'offrir un support pour les manipulations de laboratoire et les projets utilisant les microcontrôleurs PIC32MX et en particulier le PIC32MX795F512L équipant le starter-kit développé dans le cadre de l'Ecole Supérieure.

1.1. POURQUOI LE PIC32MX

L'introduction de l'utilisation d'un microcontrôleur 32 bits moderne dans le cours MINF est indispensable pour permettre aux étudiants d'étendre leurs compétences dans ce domaine. Le choix des familles de microcontrôleurs est vaste.

Dans le domaine des microcontrôleurs 32 bits à usage général, une architecture domine actuellement le marché : les microcontrôleurs à cœurs ARM. Le modèle économique d'ARM est particulier dans le sens où il n'est pas basé sur la fabrication de microcontrôleurs, mais sur la conception de propriétés intellectuelles (IP, Intellectual Properties). Ainsi, de nombreux fabricants intègrent des cœurs ARM dans leurs familles de microcontrôleurs et ajoutent leurs périphériques propres.

Il existe un concurrent, moins connus que ARM : les cœurs MIPS. Bien évidemment, s'agissant du fonctionnement du cœur du microcontrôleur, les concepts généraux sont similaires. Remarquons que le modèle économique de MIPS est également basé sur la revente d'IP.

La famille de microcontrôleurs choisie pour illustrer ce cours est la famille des PIC32MX de Microchip, basée sur un cœur MIPS. Les raisons de ce choix sont multiples :

- Famille 32 bits à usage général.
- Support de l'USB et d'Ethernet.
- Disponibilité d'outils de développement hardware (sonde de debug) et software (compilateur, IDE) gratuits ou peu coûteux.
- Migration depuis la famille précédemment utilisée (PIC18). Quand bien même il s'agissait d'une famille du même fabricant, le cœur était différent (8 bits), et l'IDE utilisé était également différent. Toutefois, la réutilisation des outils de debug et la connaissance de certains périphériques, qu'on retrouve identiques dans le PIC32, facilite la transition.

Voici les spécificités et champs d'applications du PIC32MX (en particulier le PIC32MX795F512L) par rapport aux autres membres de la famille des PIC32 basés sur cœur MIPS :

		Memory Flash/SRAM	Automotive	Connectivity	Functional Safety	Graphics	Motor Control	Security	Ultra-Low Power
PERFORMANCE ↑	PIC32MZ EF FPU MIPS32® M-Class, 252 MHz	512-2048 KB/ 128-512 KB	●	●	●	●		●	
	PIC32MZ DA MIPS32 microAptiv™, 200 MHz	1024-2048 KB/ 256-640 KB		●	●	●		●	
	PIC32MK MIPS32 microAptiv, 120 MHz	256-1024 KB/ 128-256 KB	●	●	●	●	●		
	PIC32MX 3/4 MIPS32 M4K®, 80-120 MHz	32-512 KB/ 8-128 KB		●	●				
	PIC32MX 5/6/7 MIPS32 M4K, 80 MHz	64-512 KB/ 16-128 KB		●	●				
	PIC32MX 1/2 XLP MIPS32 M4K, 72 MHz	128-256 KB/ 32-64 KB		●	●				●
	PIC32MX 1/2/5 MIPS32 M4K, 50 MHz	16-512 KB/ 4-64 KB		●	●				
	PIC32CM MC Arm® Cortex®-M0+, 48 MHz	64-128 KB/ 8-16 KB			●		●		
	PIC32MM MIPS32 microAptiv UC, 25 MHz	16-256 KB/ 4-32 KB	●						●

Tiré de www.microchip.com, " 32-bit PIC® Microcontrollers (MCUs) », consulté le 9.11.2021

Notons que Microchip fabrique également des microcontrôleurs à cœur ARM, et que les différentes familles peuvent être programmées à l'aide de l'IDE MPLAB X.

1.2. APPRENTISSAGE D'UNE NOUVELLE FAMILLE DE MICROCONTRÔLEURS

Finalement, le choix d'une famille de microcontrôleurs utilisée pour l'apprentissage et une affaire d'habitudes et de « religion ». Par rapport à une autre famille, les outils changeront, mais les concepts resteront applicables.

Tout changement de famille de microcontrôleurs nécessite forcément une certaine quantité de mises à jour et apprentissage par rapport à n'importe quelle autre famille de microcontrôleurs ou fabricant.

L'utilisation de l'architecture MIPS 32 bits et du compilateur Microchip ainsi que du framework Harmony vont se traduire par un apprentissage d'un nouvel environnement au niveau des détails pour gérer des éléments connus tels que les entrées-sorties, timers, PWM, interruptions et autres.

1.3. DÉVELOPPEMENT ET MISE AU POINT DES PROGRAMMES

Au niveau du compilateur, le choix s'est porté sur le XC32 de Microchip, génération basée sur le compilateur universel gnu et supportant C et C++.

Pour faire le lien entre l'environnement MPLABX IDE et le microcontrôleur cible, une sonde de debug (ICD : In-Circuit Debugger) de Microchip sera utilisée. Cet outil s'intègre naturellement à MPLABX.

1.4. HARMONY : APPORTS ET CONTRAINTES

Microchip présente Harmony comme : MPLAB Harmony Integrated Software Framework. Il s'agit d'un ensemble de logiciels comportant principalement les bibliothèques de périphériques, mais aussi des exemples complets permettant par exemple de gérer une pile TCP/IP. Les contraintes sont une structure lourde pour mettre en place un projet ainsi qu'une évolution des bibliothèques périphériques incompatible avec les anciennes bibliothèques de Microchip (MLA).

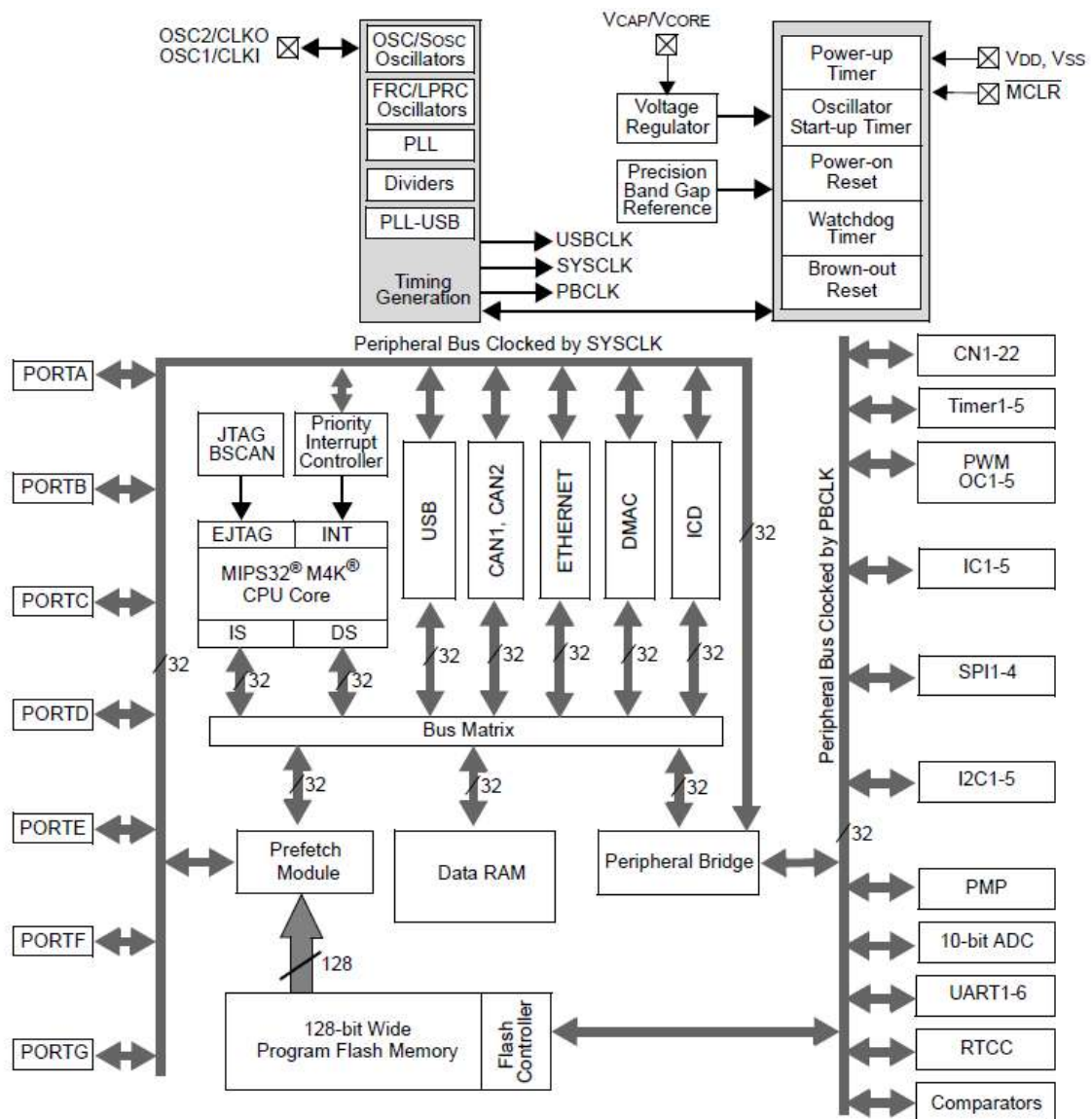
Un grand avantage est le MHC (Microchip Harmony Configurator). Il s'agit d'un assistant permettant de générer un projet complet avec la mise en œuvre des périphériques choisis.

☺ L'utilisation du MHC allège passablement le travail et réduit l'étude des nombreuses fonctions de la nouvelle PLIB.

☺ L'intégration d'un BSP spécifique est possible depuis Harmony 1.06 et MPLAB X 3.x, ce qui simplifie encore la génération de projet.

1.5. LE PIC32MX795F512L EN BREF

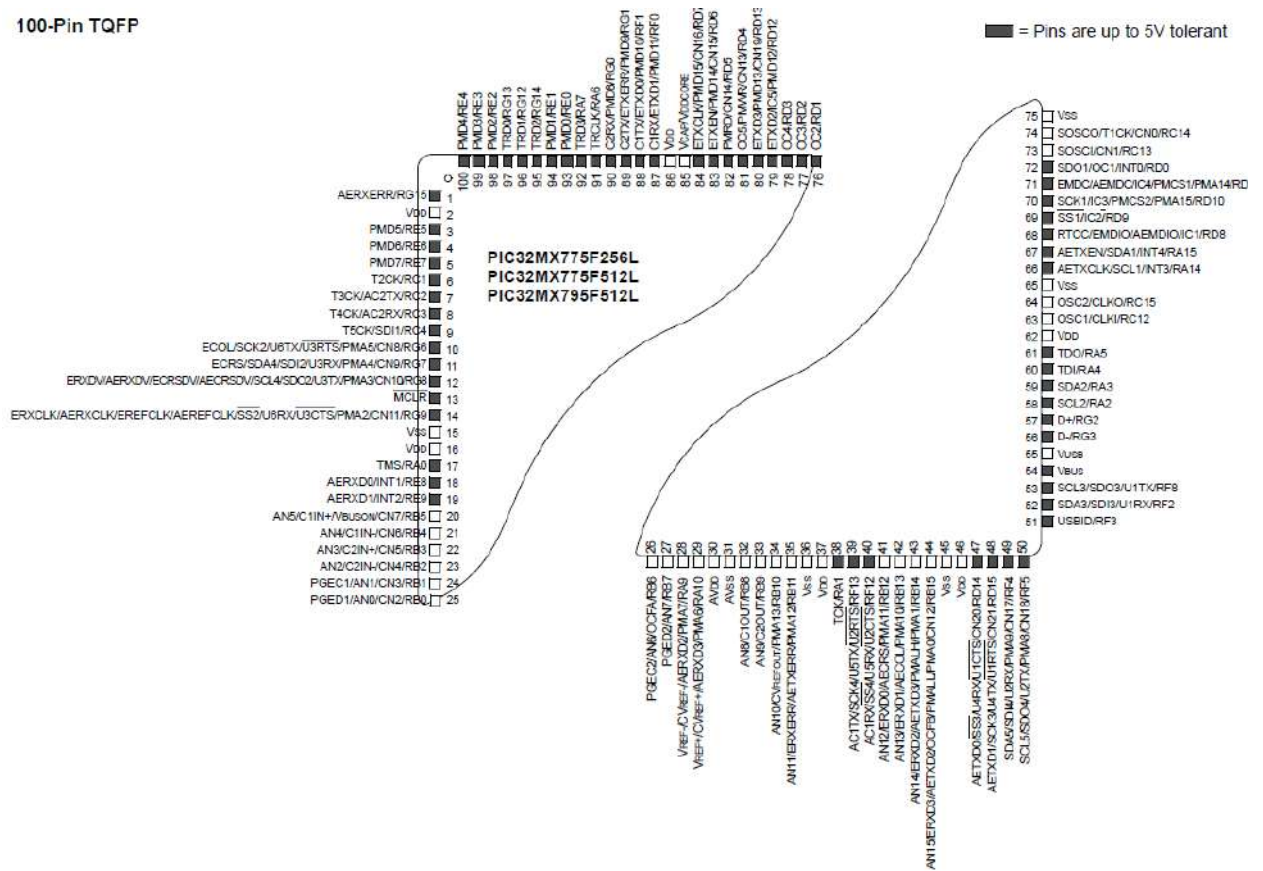
1.5.1. SCHÉMA BLOC



1.5.2. BOÎTIER ET BROCHAGE

Le boîtier choisi pour le kit est le TQFP de 100 broches. Le L signifie large.

100-Pin TQFP

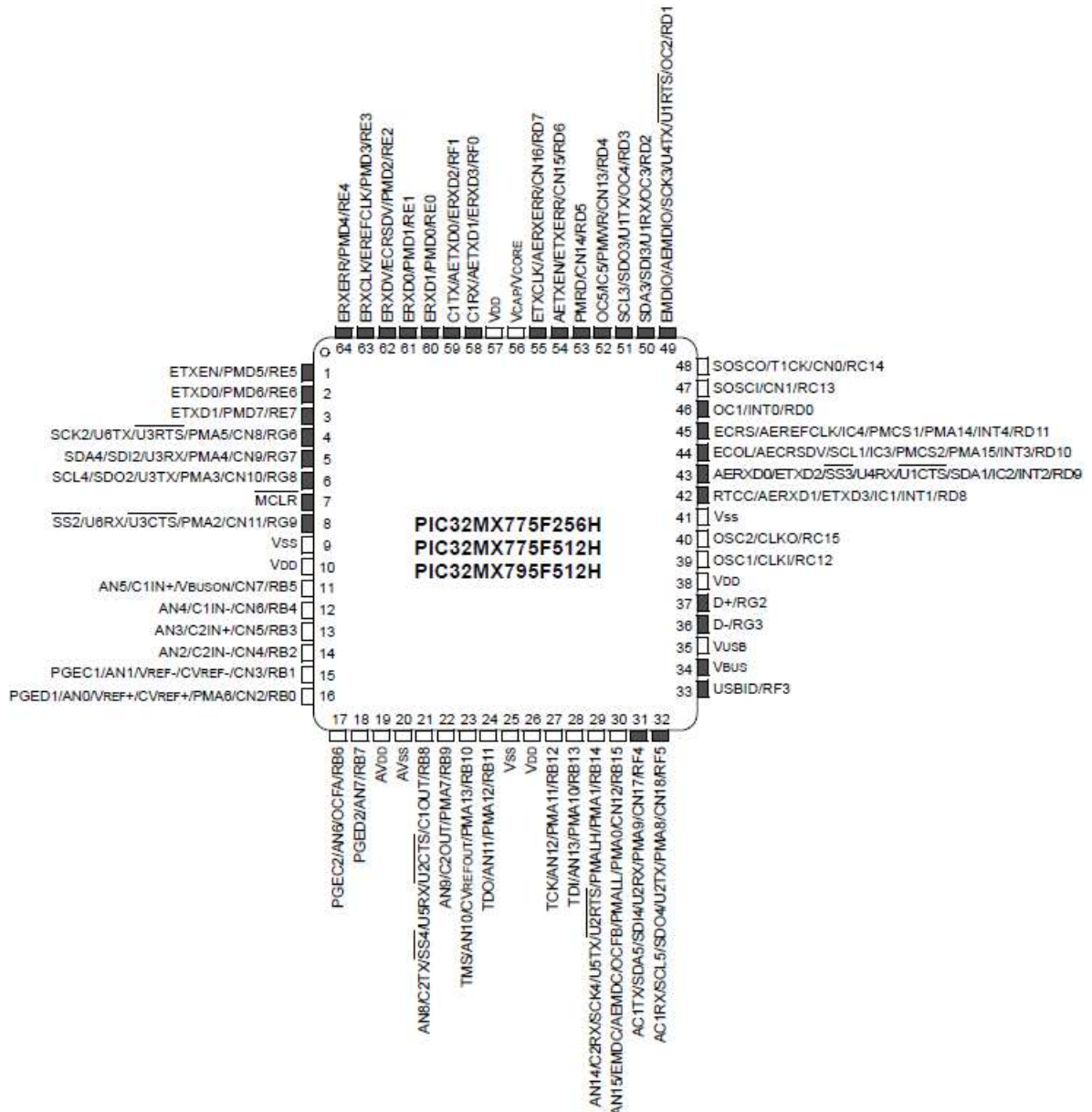


1.5.3. BOÎTIER ET BROCHAGE DE LA VERSION H

Le PIC32MX795F512H (H = half) dispose des mêmes fonctionnalités avec une réduction du nombre d'entrées-sorties à cause du boîtier TQFP de 64 broches.

64-Pin TQFP

■ = Pins are up to 5V tolerant



1.5.4. QUELQUES CARACTÉRISTIQUES



PIC32MX5XX/6XX/7XX

High-Performance, USB, CAN and Ethernet 32-bit Flash Microcontrollers

High-Performance 32-bit RISC CPU:

- MIPS32® M4K® 32-bit core with 5-stage pipeline
- 80 MHz maximum frequency
- 1.56 DMIPS/MHz (Dhrystone 2.1) performance at zero Wait state Flash access
- Single-cycle multiply and high-performance divide unit
- MIPS16e® mode for up to 40% smaller code size
- Two sets of 32 core register files (32-bit) to reduce interrupt latency
- Prefetch Cache module to speed execution from Flash

Microcontroller Features:

- Operating voltage range of 2.3V to 3.6V
- 64K to 512K Flash memory (plus an additional 12 KB of Boot Flash)
- 16K to 128K SRAM memory
- Pin-compatible with most PIC24/dsPIC® DSC devices
- Multiple power management modes
- Multiple interrupt vectors with individually programmable priority
- Fail-Safe Clock Monitor mode
- Configurable Watchdog Timer with on-chip Low-Power RC oscillator for reliable operation

Peripheral Features:

- Atomic SET, CLEAR and INVERT operation on select peripheral registers
- Up to 8-channels of hardware DMA with automatic data size detection
- USB 2.0-compliant full-speed device and On-The-Go (OTG) controller:
 - Dedicated DMA channels
- 10/100 Mbps Ethernet MAC with MII and RMII interface:
 - Dedicated DMA channels
- CAN module:
 - 2.0B Active with DeviceNet™ addressing support
 - Dedicated DMA channels
- 3 MHz to 25 MHz crystal oscillator

Peripheral Features (Continued):

- Internal 8 MHz and 32 kHz oscillators
- Six UART modules with:
 - RS-232, RS-485 and LIN support
 - IrDA® with on-chip hardware encoder and decoder
- Up to four SPI modules
- Up to five I²C™ modules
- Separate PLLs for CPU and USB clocks
- Parallel Master and Slave Port (PMP/PSP) with 8-bit and 16-bit data, and up to 16 address lines
- Hardware Real-Time Clock and Calendar (RTCC)
- Five 16-bit Timers/Counters (two 16-bit pairs combine to create two 32-bit timers)
- Five Capture inputs
- Five Compare/PWM outputs
- Five external interrupt pins
- High-speed I/O pins capable of toggling at up to 80 MHz
- High-current sink/source (18 mA/18 mA) on all I/O pins
- Configurable open-drain output on digital I/O pins

Debug Features:

- Two programming and debugging Interfaces:
 - 2-wire interface with unintrusive access and real-time data exchange with application
 - 4-wire MIPS® standard enhanced Joint Test Action Group (JTAG) interface
- Unintrusive hardware-based instruction trace
- IEEE Standard 1149.2 compatible (JTAG) boundary scan

Analog Features:

- Up to 16-channel, 10-bit Analog-to-Digital Converter:
 - 1 Msps conversion rate
 - Conversion available during Sleep and Idle
- Two Analog Comparators

1.5.5. USB, ETHERNET ET CAN SELON MODÈLE

TABLE 3: PIC32 USB, ETHERNET AND CAN – FEATURES

USB, Ethernet and CAN																	
Device	Pins	Program Memory (KB)	Data Memory (KB)	USB	Ethernet	CAN	Timers/Capture/Compare	DMA Channels (Programmable/ Dedicated)	UART ^(2,3)	SPI ⁽³⁾	I ² C™ ⁽³⁾	10-bit 1 Msps ADC (Channels)	Comparators	PMP/PSP	JTAG	Trace	Packages ⁽⁴⁾
PIC32MX764F128H	64	128 + 12 ⁽¹⁾	32	1	1	1	5/5/5	4/6	6	3	4	16	2	Yes	Yes	No	PT, MR
PIC32MX775F256H	64	256 + 12 ⁽¹⁾	64	1	1	2	5/5/5	8/8	6	3	4	16	2	Yes	Yes	No	PT, MR
PIC32MX775F512H	64	512 + 12 ⁽¹⁾	64	1	1	2	5/5/5	8/8	6	3	4	16	2	Yes	Yes	No	PT, MR
PIC32MX795F512H	64	512 + 12 ⁽¹⁾	128	1	1	2	5/5/5	8/8	6	3	4	16	2	Yes	Yes	No	PT, MR
PIC32MX764F128L	100	128 + 12 ⁽¹⁾	32	1	1	1	5/5/5	4/6	6	4	5	16	2	Yes	Yes	Yes	PT, PF, BG
PIC32MX775F256L	100	256 + 12 ⁽¹⁾	64	1	1	2	5/5/5	8/8	6	4	5	16	2	Yes	Yes	Yes	PT, PF, BG
PIC32MX775F512L	100	512 + 12 ⁽¹⁾	64	1	1	2	5/5/5	8/8	6	4	5	16	2	Yes	Yes	Yes	PT, PF, BG
PIC32MX795F512L	100	512 + 12 ⁽¹⁾	128	1	1	2	5/5/5	8/8	6	4	5	16	2	Yes	Yes	Yes	PT, PF, BG

Legend: PF, PT = TQFP MR = QFN BG = XBGA

Note 1: This device features 12 KB boot Flash memory.

Note 2: CTS and RTS pins may not be available for all UART modules. Refer to the “Pin Diagrams” section for more information.

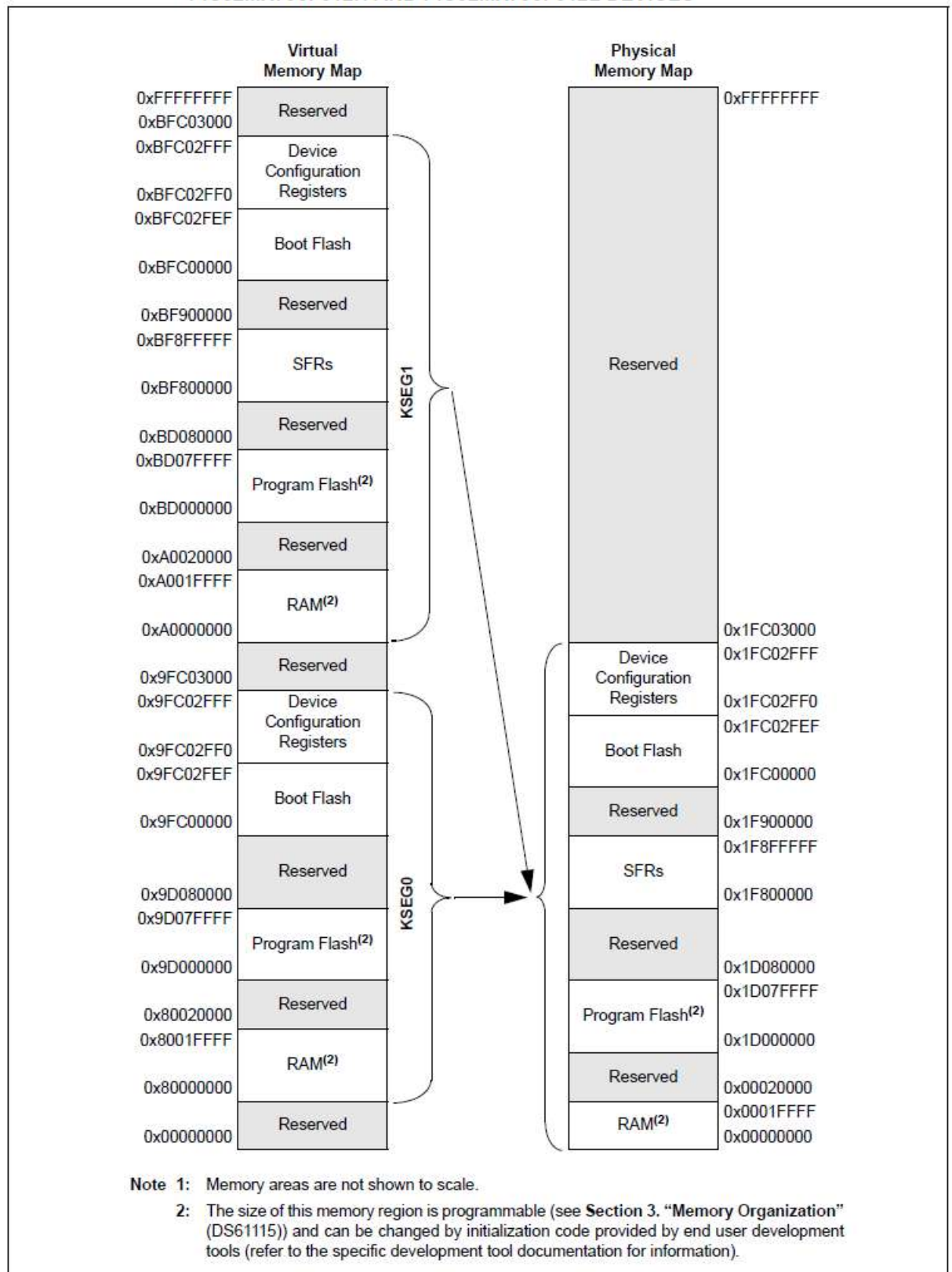
Note 3: Some pins between the UART, SPI and I²C modules may be shared. Refer to the “Pin Diagrams” section for more information.

Note 4: Refer to 32.0 “Packaging Information” for more information.

👉 Le PIC32MX795F512H, le petit frère avec 64 broches, est un bon choix pour certains projets nécessitant l'ethernet et/ou l'USB. Si ces fonctionnalités ne sont pas nécessaires, on pourra s'orienter vers une autre série aux fonctionnalités plus réduites.

1.5.6. ORGANISATION DE LA MÉMOIRE

FIGURE 4-6: MEMORY MAP ON RESET FOR PIC32MX695F512H, PIC32MX695F512L, PIC32MX795F512H AND PIC32MX795F512L DEVICES



1.6. DOCUMENTATIONS À DISPOSITION

1.6.1. DOCUMENTATION PIC32MX

Vous trouverez l'ensemble de la documentation des PIC32MX sur le réseau sous ...\\PROJETS\\SLO\\1102x_SK32MX775F512L\\Datasheets\\PIC32 Family Reference Manual.

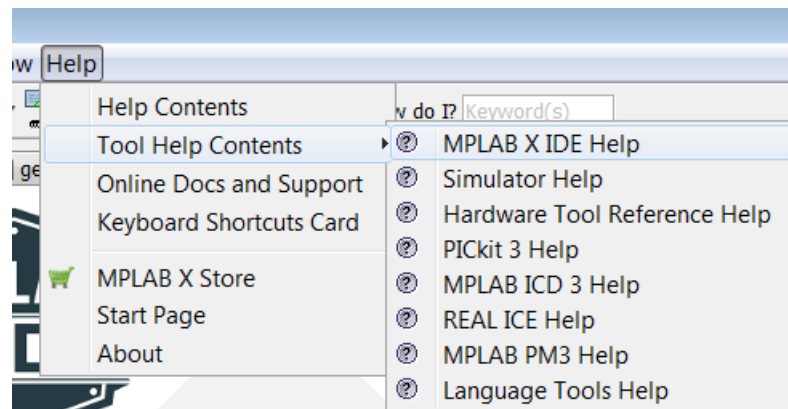
Remarque le projet 1102x est la première version du kit PIC32MX qui n'est plus utilisé.

1.6.2. DOCUMENTATION DU KIT PIC32MX

Vous trouverez le schéma du kit sur le réseau sous ...\\PROJETS\\SLO\\1102x_SK32MX775F512L\\Hardware\\11020_SK32MX775F512L

1.6.3. DOCUMENTATION COMPILATEUR ET ENVIRONNEMENT

Dans MPLAB X, l'aide est accessible à partir du menu :



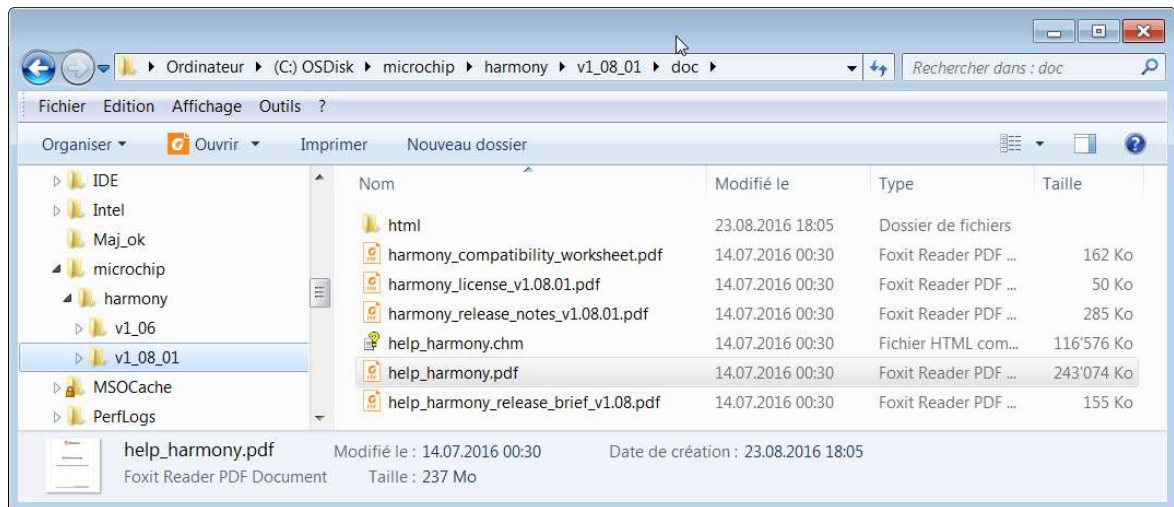
Pour le compilateur XC32, sa documentation se trouve sous :

<Répertoire xc32>\\v<n>\\docs.

En particulier dans le fichier MPLAB-XC32-Users-Guide.pdf.

1.6.4. DOCUMENTATION HARMONY

La documentation Harmony est présente dans le sous répertoire \doc de l'installation de Harmony.



1.6.5. DOCUMENTATION DANS L'ENVIRONNEMENT MPLABX

Dans l'ensemble de documentations accessible par le menu Help du MPLABX, on trouve la section **XC32 ToolChain** et la sous-section XC32 Peripheral Librairies qui correspond à l'ancienne PLIB.

😊 Lorsque l'on utilise le MHC, une aide contextuelle est fournie automatiquement.

1.7. HISTORIQUE DES VERSIONS

1.7.1. V1.0 2013

Traite de MPLABX et XC32 sans Harmony (ancienne plib)

1.7.2. V1.5

Traite de MPLABX et XC32 avec Harmony (nouvelle plib)

1.7.3. V1.6

Traite de MPLABX 3.10 et XC32 avec Harmony 1.06

1.7.4. V1.7 OCTOBRE 2016

Traite de MPLABX 3.40 et XC32 avec Harmony 1.08

1.7.5. V1.8 NOVEMBRE 2017

Traite de MPLABX 3.61 et XC32 avec Harmony 1.11 (sauf indication contraire dans le document). Reprise et relecture par SCA

1.7.6. V1.81 NOVEMBRE 2021

Compléments choix du PIC32MX.