

la documentation des microcontrôleurs sans peine (3)

schémas de principe, et autres documents

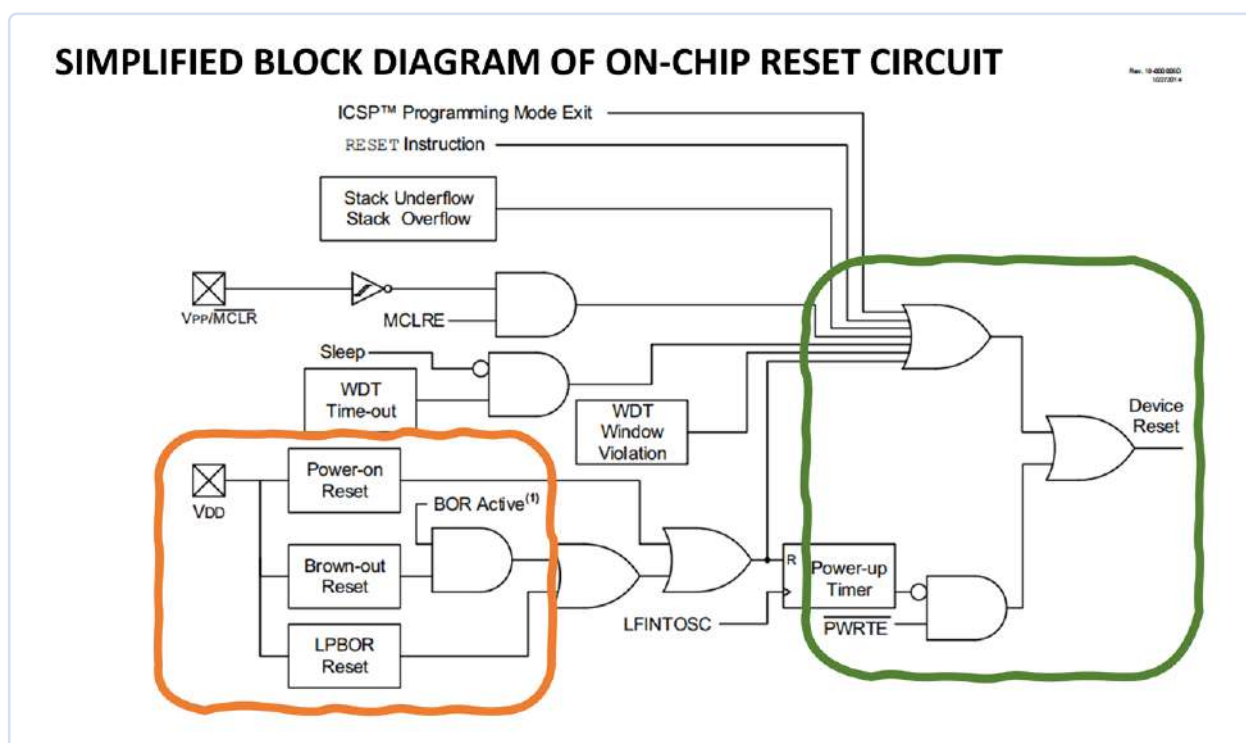


Figure 1. Le circuit de réinitialisation offre très peu d'options de configuration, certaines sources de réinitialisation étant liées à la broche d'alimentation VCC. (Source : Microchip Technology)

Stuart Cording (Elektor)

Dans les parties précédentes de cette série d'articles [1], nous avons abordé les fonctionnalités des registres et examiné le schéma de principe d'une horloge. Nous présentons ici d'autres schémas en précisant aussi où trouver le reste de la documentation nécessaire.

Cette fois-ci encore, nous allons nous focaliser spécifiquement sur le microcontrôleur 8 bits PIC16F18877 de Microchip Technology et nous vous conseillons donc de télécharger la fiche technique [2] si vous souhaitez suivre notre démarche.

Commençons par la réinitialisation

Si le périphérique d'horloge est la partie la plus importante d'un microcontrôleur, la mise en œuvre du circuit de réinitialisation est le deuxième bloc essentiel. Tout comme l'horloge, il ne doit être configuré qu'une seule fois (s'il existe des options de configuration), et vous éviterez bien des problèmes ultérieurs si vous comprenez bien ce qui peut provoquer une réinitialisation.

Dans l'exemple de la page 100 (**figure 1**), nous pouvons voir que deux broches (représentées par des carrés contenant une croix,

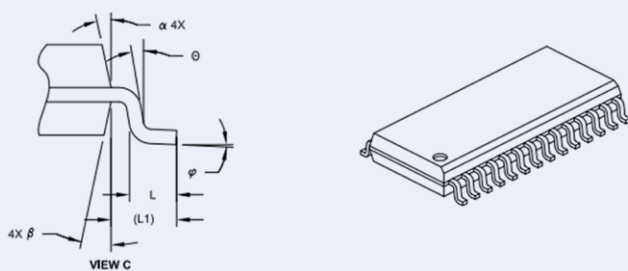
Rev. 10-000017
4-2015 (2015)

Représentés en orange, une série de mécanismes de réinitialisation, notamment la détection de la mise sous tension et les baisses de tension électrique (*brown-out*), sont liés à la broche d'alimentation du microcontrôleur. Dans le cas présent, le dispositif propose un contrôle limité, toutes les sources d'interruption étant pour l'essentiel des signaux de réinitialisation possibles (représentés en vert). Comme nous l'avons vu dans un précédent article avec le registre **STATUS**, il est possible dans certains cas de déterminer la cause de la réinitialisation précédente.

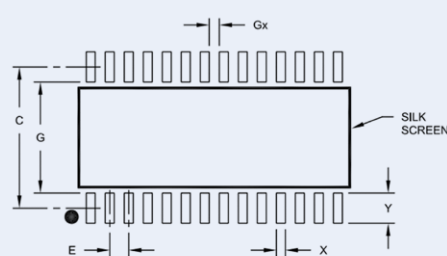
Bien que vous puissiez reconnaître la bascule D [3], sur le côté droit, connectée à la broche de sortie, différents blocs du schéma sont tout simplement des carrés. Les représentations de certains, comme le prédiviseur (prescaler) et le postdiviseur (postscaler), vont de soi. Il en va de même pour le bloc **SYNC**, mais un développeur devra probablement lire le texte d'accompagnement en détail pour comprendre pleinement sa fonction. Les diagrammes et le texte vont ainsi souvent de pair dans les fiches techniques. Il est parfois nécessaire d'écrire quelques lignes de code de test pour vraiment comprendre le fonctionnement de certains périphériques.

Le boîtier SOIC illustré à la **figure 3** est présenté *pages 645 et 646*. Pour les concepteurs de circuits imprimés, l'information la plus utile aujourd'hui est peut-être la hauteur du boîtier si la production du dispositif est en volume limité (ils connaissent déjà la largeur et la longueur). De nos jours, de nombreux boîtiers comportent également une « pastille métallique » intégrée qui doit être soudée au circuit imprimé pour une meilleure dissipation thermique. Ce n'est pas le cas ici. Sinon, des indications sur la manière de le connecter seraient données. Vous devrez peut-être aussi vérifier si cette pastille est connectée à la masse de l'appareil ou à une autre broche.

- **Jeu d'instructions du processeur** : si le nombre d'instructions est faible (moins de 60 à 70), elles peuvent toutes être incluses, avec une explication, dans la fiche technique. Si ce n'est pas le cas, il existe probablement un document séparé dans lequel chaque instruction sera expliquée en détail.
- **Exemples de code** : ils apparaissent plus souvent lorsque le microcontrôleur est encore essentiellement programmé en assembleur. Les exemples de code en langage de haut niveau, le C par exemple, n'ont guère de sens, puisque le compilateur définit les instructions précises utilisées.
- **Exemples de circuits** : ils sont le plus souvent associés à des fonctionnalités propres au microcontrôleur sélectionné, comme l'oscillateur ou l'alimentation, ou à des interfaces



Dimension	Limits	MILLIMETERS		
		MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	1.27 BSC		
Overall Height	A	-	-	2.65
Molded Package Thickness	A2	2.05	-	-
Standoff	§	0.10	-	0.30
Overall Width	E	10.30 BSC		
Molded Package Width	E1	7.50 BSC		
Overall Length	D	17.90 BSC		
Chamfer (Optional)	h	0.25	-	0.75
Foot Length	L	0.40	-	1.27
Footprint	L1	1.40 REF		
Lead Angle	θ	0°	-	-
Foot Angle	φ	0°	-	8°
Lead Thickness	c	0.18	-	0.33
Lead Width	b	0.31	-	0.51
Mold Draft Angle Top	α	5°	-	15°
Mold Draft Angle Bottom	β	5°	-	15°



Dimension	Limits	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	1.27 BSC		
Contact Pad Spacing	C	9.40		
Contact Pad Width (X28)	X	0.60		
Contact Pad Length (X28)	Y	2.00		
Distance Between Pads	Gx	0.67	-	-
Distance Between Pads	G	7.40	-	-

Figure 3. Les plans du boîtier indiquent ses dimensions (à gauche) ainsi que la configuration recommandée pour le circuit imprimé (à droite). (Source : Microchip Technology)

légèrement plus complexes qui ont des exigences spécifiques en matière de charge de signaux, comme l'USB. Toutefois, il est très probable que ces circuits soient abordés plus en détail dans une note d'application relative au sujet.

Quelles sont les informations généralement absentes d'une fiche technique de microcontrôleur ?

Les éléments suivants sont normalement exclus des fiches techniques et figurent dans d'autres documents. C'est généralement parce qu'ils concernent un sujet commun à une large gamme de microcontrôleurs.

- Flashage de la mémoire du microcontrôleur : ce sujet est normalement traité séparément pour la programmation des microcontrôleurs dans le cadre de la production de masse.
- Utilisation d'outils de développement : le compilateur, l'environnement de développement intégré (IDE) et les outils de débogage disposent de leur propre documentation.
- Explication détaillée du cœur du processeur : ce point fait souvent l'objet d'un document séparé, en particulier pour les cœurs 16 et 32 bits.
- Explication détaillée des périphériques complexes : les périphériques USB, les interfaces graphiques et les périphériques Ethernet font généralement l'objet de documents distincts, car l'inclusion de chacun d'entre eux pourrait doubler la taille de la fiche technique du microcontrôleur.
- Errata : cette partie répertorie les erreurs contenues dans la documentation, ou les solutions de contournement pour résoudre les bogues matériels non corrigés.

Quelles sont les autres documentations mises à votre disposition ?

Outre la fiche technique, les documents suivants sont généralement proposés :

- « Manuel de référence de la gamme » : alors que la fiche technique vous indique précisément ce qu'un microcontrôleur

peut faire, ces documents offrent une vue globale de ce que peuvent accomplir tous les microcontrôleurs d'une même gamme.

- Notes d'application : cette partie explique en détail comment utiliser des périphériques spécifiques, ou un groupe de périphériques, pour mettre en œuvre une application ou une interface. Si la fiche technique peut décrire, par exemple, comment utiliser un périphérique CAN (Controller Area Network), une note d'application indiquera plutôt comment l'exploiter dans le cadre d'un réseau CAN, avec des conseils sur les protocoles logiciels de haut niveau et la sélection d'émetteurs-récepteurs appropriés.
- Spécification de programmation : pour la programmation dans des environnements de production de masse, elle explique en détail les tensions et les temporisations requises, ainsi que le ou les protocoles éventuels utilisés pour l'interface de programmation.

Comment assimiler ces documentations volumineuses ?

Malheureusement, il n'existe pas de moyen particulier pour assimiler rapidement toutes les informations associées à un microcontrôleur et à l'ensemble de ses outils. Si vous débutez, il est probablement préférable de lire la fiche technique parallèlement à un livre ou à un article sur le microcontrôleur concerné. Elektor ne propose pas seulement des livres pour ceux qui s'intéressent aux microcontrôleurs PIC ; il existe également des livres de démarrage pour les microcontrôleurs STM32 [4] et ARM [5] en général. Et pourquoi ne pas essayer un simple projet MSP430 [6] ?

La combinaison d'exemples pratiques et de deux types de documents explicatifs écrits (un livre/article et la fiche technique elle-même) pourra vous aider. Si vous êtes plus avancé(e), la meilleure approche consiste à vous focaliser sur les sections qui concernent le cœur du processeur, l'arborescence d'horloge et le bloc de réinitialisation, puis les périphériques que vous souhaitez utiliser. Vous devrez ensuite vous familiariser avec la

documentation de la chaîne d'outils du compilateur. Faites également bon usage des bibliothèques logicielles et des exemples pour améliorer votre compréhension. Si vous avez besoin d'aide supplémentaire, posez des questions sur les forums.

Les fiches techniques peuvent être difficiles à lire et à comprendre, et elles ne sont pas particulièrement enthousiasmantes sur le plan littéraire. Pour autant, elles sont (la plupart du temps) exactes et constituent une description technique de la fonctionnalité (la section « errata » peut servir à déterminer la confiance à accorder à la fiche consultée). Gardez-les à portée de main, et, avec le temps, vous comprendrez bien comment elles sont construites et formulées. ◀

VF : Pascal Godart — 230286-04

Des questions, des commentaires ?

Envoyez un courriel à l'auteur (stuart.cording@elektor.com).



Produits

- **T. Hanna, *Microcontroller Basics with PIC*, Elektor 2020**
<https://elektor.fr/19188>
- **A. Pratt, *Programming the Finite State Machine*, Elektor 2020**
<https://elektor.fr/19327>

LIENS

- [1] Série relative à la documentation des microcontrôleurs : <https://elektormagazine.com/tags/microcontroller-documentation>
- [2] Fiche technique du microcontrôleur Microchip Technologies PIC16F18877 : <https://microchip.com/en-us/product/PIC16F18877>
- [3] Bascules D: [https://fr.wikipedia.org/wiki/Bascule_\(circuit_logique\)](https://fr.wikipedia.org/wiki/Bascule_(circuit_logique))
- [4] Livre sur les microcontrôleurs STM32 pour débutants: <https://elektor.com/programming-with-stm32-nucleo-boards-e-book>
- [5] Microcontrôleurs à base de processeurs ARM : <https://elektor.fr/embedded-in-embedded>
- [6] Projet MSP430 simple : <https://elektormagazine.fr/labs/elektorpost-no-1-led-earring>



Farnell
AN AVNET COMPANY

SOLUTIONS LOGICIELLES DE TEST ET MESURE

Nous livrons maintenant votre solution logicielle numérique adaptée à votre équipement.

Choisissez votre logiciel de test et mesure et recevez-le par e-mail dans les 72 heures suivant l'achat.

