Zeste de savoir

# L'avènement des chiplets

6 juillet 2022

# Table des matières



## Introduction

Suite à une discussion en JZDS et sur le Discord, je me suis dit qu'écrire un article sur les chiplets serait bénéfique au plus grand nombre. Et permettrait de garder une trace écrite contrairement à ce qui peut se passer aux JZDS 🍊
Plutôt que d'écrire un très long billet, j'ai préféré le format de l'article pour rentrer un peu plus dans les détails. J'espère ainsi pouvoir vous apprendre ce que sont les chiplets, pourquoi cette technologie a été créée et pourquoi elle va se développer dans les années à venir.

## 1. Préambule

Cet article parle de notions informatiques, électroniques et d'architecture des ordinateurs qui peuvent être assez avancées pour certains lecteurs. Je vous propose dans ce préambule un peu de vulgarisation permettant de comprendre un peu mieux ce dont on parle.

> ! Pour les puristes, des raccourcis seront faits, il se peut que cette vulgarisation puisse contenir des informations volontairement imprécises pour faciliter la compréhension.

## 2. Chiplet, quésaco ?

Commençons par le plus difficile, définir ce qu'est un chiplet! 🍊
En effet le terme chiplet est apparu dans les années 70 mais son utilisation a surtout décollé ces dernières années, pour ceux qui s'intéressent aux processeurs ou aux puces électroniques complexes comme les FPGA (des puces dont on peut reprogrammer les portes logiques internes).

Pour les autres, au fond de la salle, vous n'avez peut-être jamais entendu parler de ce terme, nous allons y remédier! 🍊

Revenons d'abord aux bases de ce qu'est une puce électronique: un morceau de silicium gravé (les fameux transistors) qui est encapsulé dans un boîtier. Avec les composants traversants, de minuscules fils d'or ou d'argent relient les pattes du composant au morceau de silicium. Au début les puces sont composées de transistors gravés à des résolutions assez grossières (comparées à aujourd'hui) et les fonctions étaient assez basiques: des portes logiques, des amplificateurs opérationnels, etc. Néanmoins c'était déjà un progrès énorme en terme de miniaturisation!

À cette époque les composants ont des pattes traversantes et il faut relier la puce en silicium à ces pattes. C'est réalisé avec de minces fils d'argent ou d'or qui sont soudés entre la puce et les pattes à l'intérieur du boîtier.

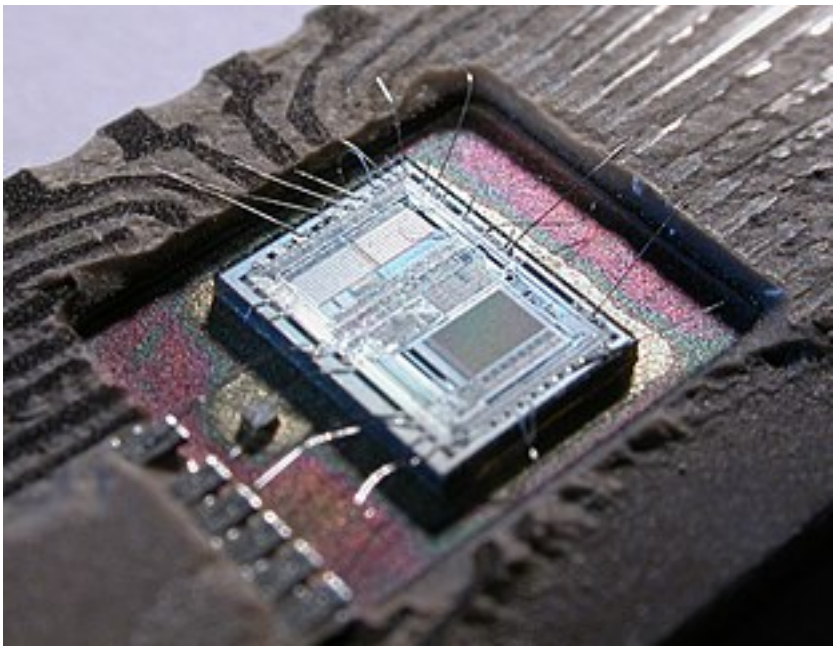

FIGURE 2.1. – Processeur Intel 8742 - Fils de bonding visibles (Ioan Sameli )

Quelques années plus tard, sont apparus les premiers processeurs avec notamment l'Intel 4004, relativement simples aujourd'hui. Puis les processeurs se sont complexifiés.

À partir des années 70, IBM développe des composants MCM (*Multi-chip Module*) comprenant plusieurs puces de silicium dans un seul boîtier. Mais cette technologie va surtout se développer à la fin des années 90. On peut noter le Pentium Pro d'Intel sorti en 1995. Ce processeur comprenait deux puces en silicium: une pour le processeur à proprement parler et une autre pour la mémoire cache L2 (une mémoire tampon entre le processeur et la RAM, beaucoup plus rapide mais bien plus chère car gravée avec le processeur).

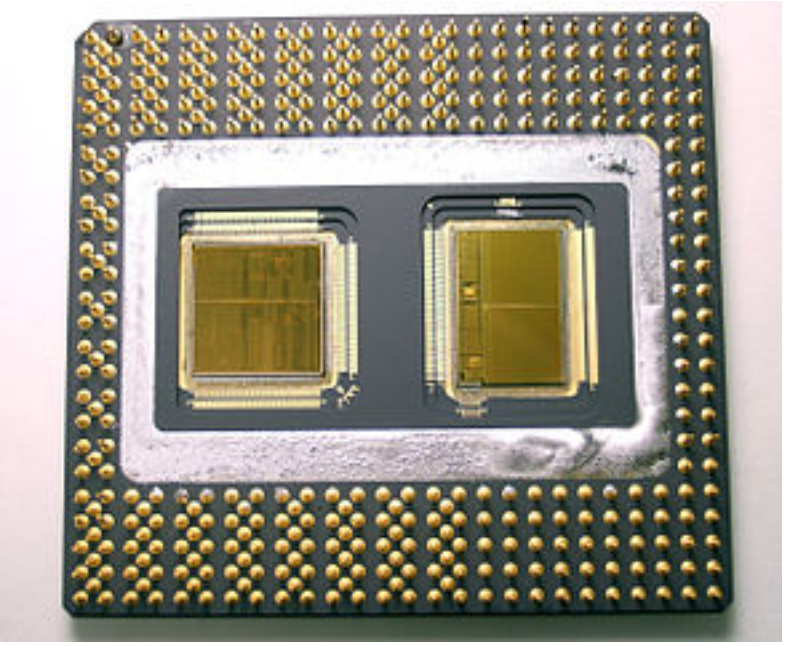

FIGURE 2.2. – Intel Pentium Pro 256KB (Moshen )

Comme on peut le voir sur la photo les deux puces ont à peu près la même taille et Intel proposait plusieurs tailles de mémoire cache L2. L'avantage de séparer le processeur de la mémoire cache était de pouvoir faire des économies d'échelle sur la puce processeur tout en proposant différentes tailles de mémoire cache en mettant une puce de taille différente dans le boîtier.

Ce type de composants reste relativement peu développé, même si IBM a continué à développer des composants MCM. Notons le POWER5 d'IBM sorti en 2004 qui voit carrément quatre processeurs avec chacun une puce de mémoire cache L3. L'interconnexion des puces se fait à l'intérieur du boîtier.

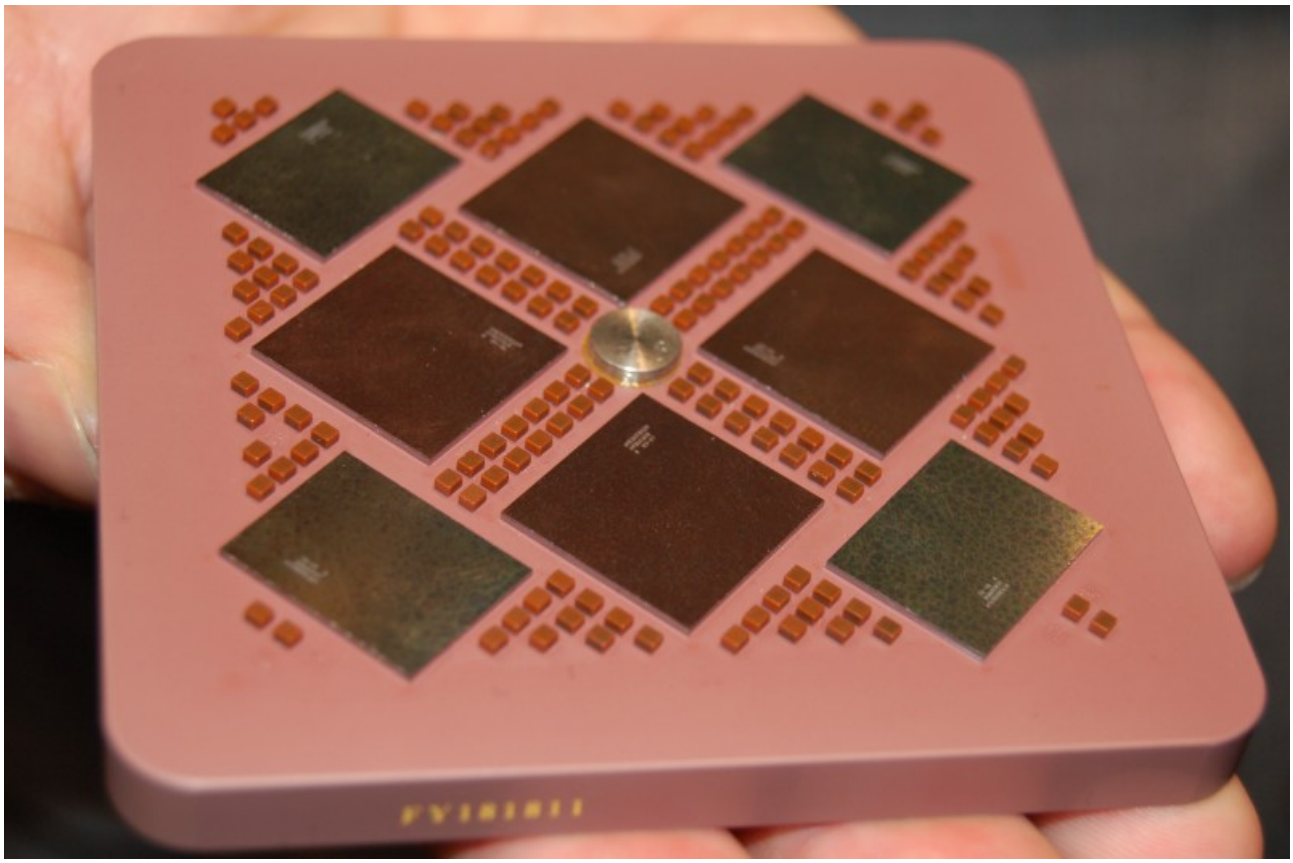

FIGURE 2.3. – IBM POWER5 (Carsten Schulz )

Aujourd'hui la technologie à évoluée et les puces MCM sont présentes dans les produits grand public avec les processeurs AMD. Ici nous pouvons voir un processeur EPYC 7702 (sorti en août 2019) composé de 9 puces en silicium interconnectées: 8 puces contenant des cœurs et de la mémoire cache et une puce centrale reliant les 8 autres et qui gère la DDR ainsi que les signaux d'entrée/sortie (SATA, PCI express, USB, etc.).

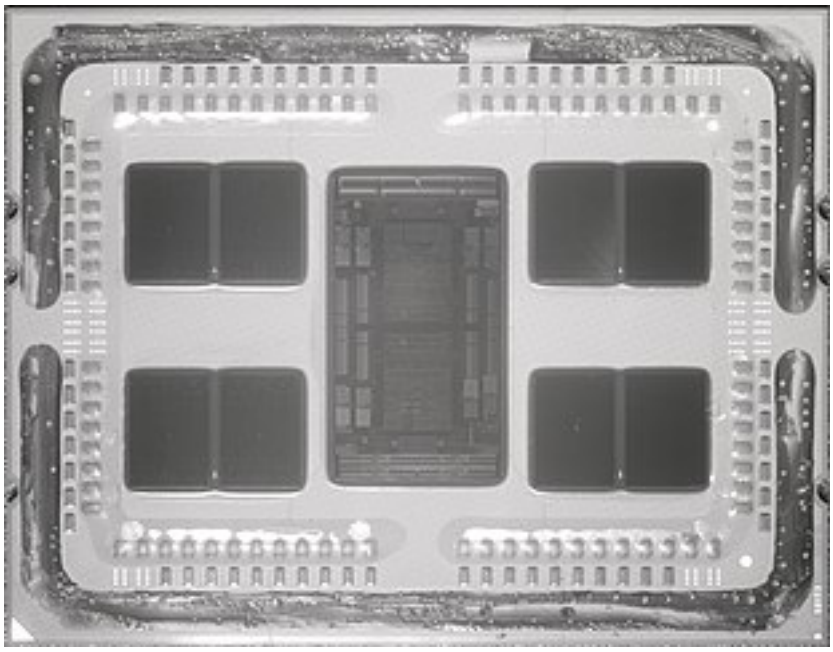

FIGURE 2.4. – AMD EPYC 7702 (Fritzchens Fritz )

? Mais dis-moi Jamy, c'est quoi un chiplet?

Ah oui, j'ai un peu dérivé
En fait un chiplet c'est une des puces en silicium présente dans un MCM. Un chiplet est fait pour être interconnecté à d'autres chiplets. Oui c'est relativement simple mais il fallait bien montrer quelques photos sympathiques pour comprendre

Néanmoins pour être un peu plus précis sur le sens des chiplets, l'idée n'est pas forcément de mettre plusieurs puces différentes reliées entre elles. Il y a aussi une notion de puce générique qui puisse être réutilisée et non pas dédiée à une référence de processeur en particulier.

## 3. Aspects économiques des chiplets

Après cette introduction tout en image, cherchons maintenant à comprendre pourquoi les chiplets vont se développer à l'avenir. Pour cela il faut revenir sur le processus de fabrication des puces électroniques.

i Installez-vous confortablement dans un fauteuil car le voyage depuis la plage de sable fin sera long

Non, attendez!
On va passer toute une partie de la fabrication du silicium. Ce qui va nous intéresser c'est la répartition des puces (*die*) sur la galette de silicium (*wafer*) et notamment l'évolution du rendement avec l'augmentation de la finesse de gravure.

Mais avant cet aspect du rendement, il faut parler de la taille physique maximum d'un die. En effet, sur une galette de silicium le même design d'une puce est répété plusieurs fois (dizaine voire centaine de fois). L'impression de ce design se fait de manière optique via de la lumière ultraviolette. Or il y a tout un jeu de lentilles et mécanismes optiques qui empêche de graver un unique die sur l'entièreté de la galette de silicium.
Plus on complexifie les puces et plus on souhaite mettre de transistors, il faut donc soit augmenter la taille de la puce, soit augmenter la finesse de gravure pour caser plus de transistors dans une même surface. Mais là aussi d'autres contraintes et limites se font sentir.

C'est pour cela que le principe de chiplet est intéressant pour contourner ces limites: utiliser plusieurs petites puces en silicum connectées entre elles pour réaliser une puce plus complexe mais impossible à graver de manière monolithique.

Revenons maintenant sur le rendement (*yield* en anglais). Premièrement, les wafers sont de forme ronde et on souhaite graver dessus des puces de forme rectangulaire. L'entièreté du silicium n'est pas donc pas utilisée. Mais plus les dies sont petits au niveau des bords et plus on pourra avoir de dies entiers. C'est le même principe que l'aliasing dans un jeu vidéo: plus les pixels utilisés pour former une forme ronde sont petits et moins on se rend compte du crénelage.

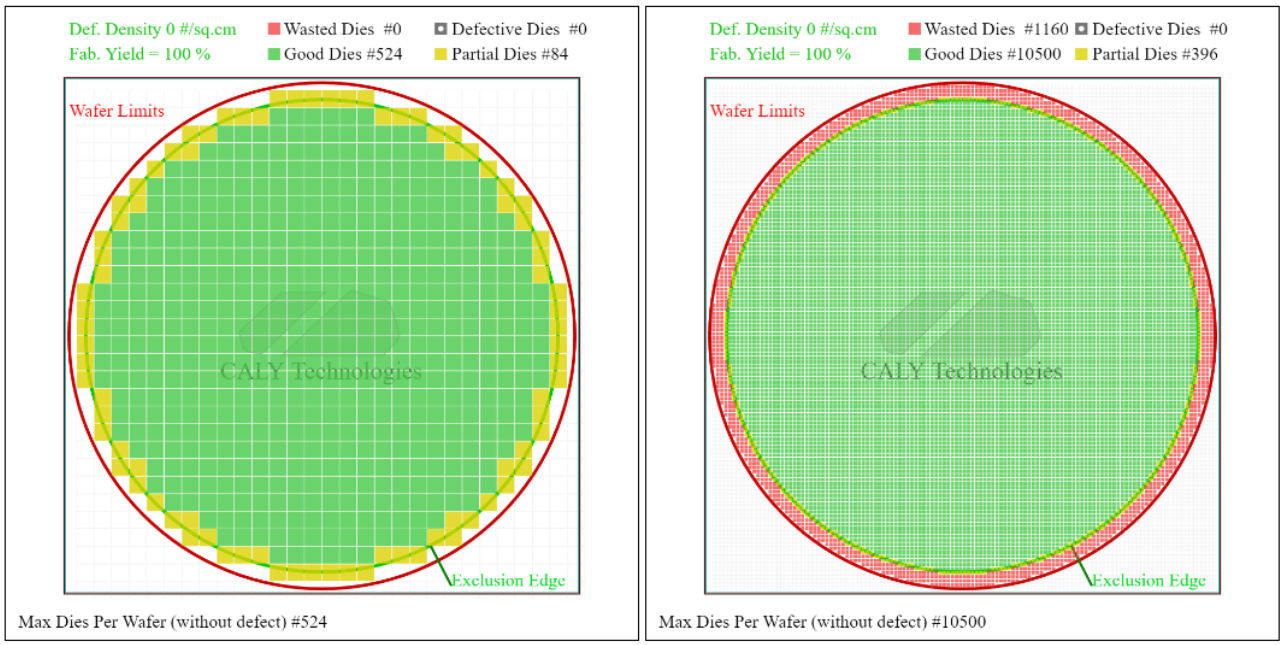


FIGURE 3.5. – À gauche: die de 5×5 mm - À droite: die de 1×1 mm

Dans l'exemple ci-dessus, si l'on fait le rapport de dies partiaux sur le nombre total de dies (bon et partiaux) on obtient un ratio de 13.8% dans le cas de dies de 5×5 mm et de 3.6% dans le cas de dies de 1×1 mm. Plus le die est petit et plus on peut avoir de dies valides au niveau des bords, ce qui augmente le rendement.
On peut aussi faire un mélange de die de grande taille au centre du wafer et utiliser des dies de plus petite taille au niveau des bords pour optimiser le rendement dû à l'aliasing.

?

Dis Jamy, pourquoi utilise-t-on des wafers ronds pour faire des puces rectangulaires? 🍊
Eh bien c'est à cause de la méthode de fabrication du silicium appelé procédé de Czochralski qui donne du silicium sous forme de cylindres, découpés en tranches très fines pour donner des *wafers*.

Deuxièmement, le rendement est affecté par les défauts qui peuvent apparaître sur le wafer. Vous pouvez penser à des grains de poussière qui tombent sur le wafer.

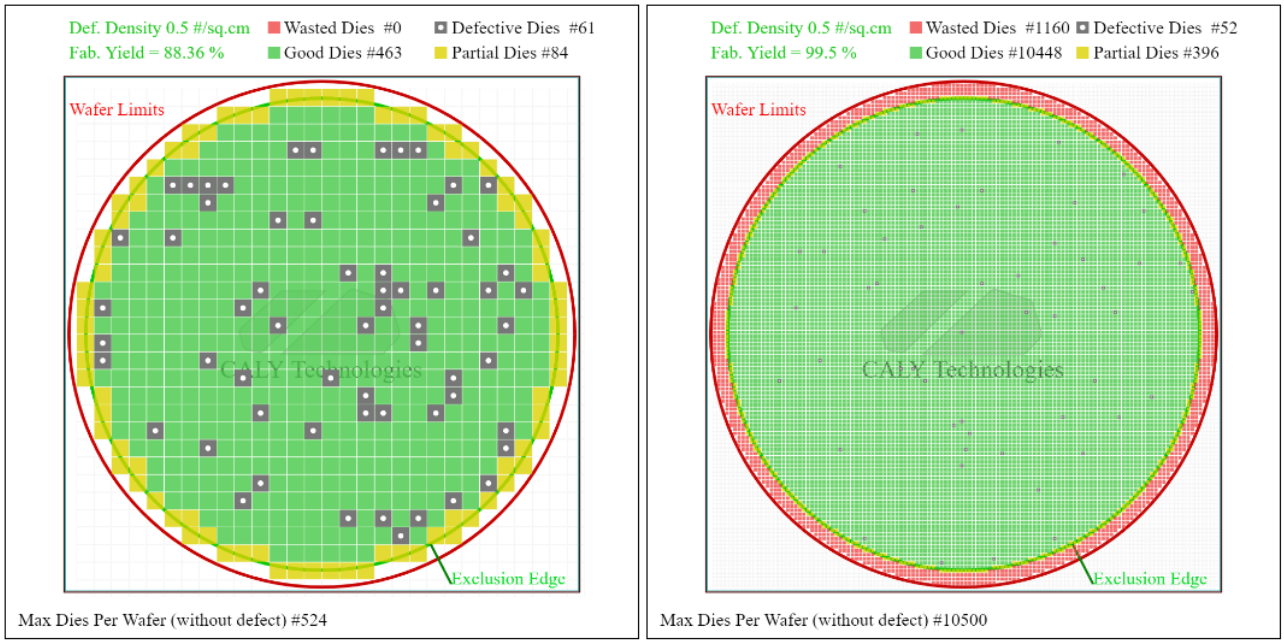


FIGURE 3.6. – À gauche: die de 5×5 mm - À droite: die de 1×1 mm

J'ai repris l'exemple précédent en ajoutant une densité de défaut de 0.5 par $cm^2$. Comparons maintenant le *fabrication yield* qui correspond au ratio entre le nombre de dies fonctionnels et le nombre total de dies produits. Dans le cas d'un die de 5×5 mm, le yield est de 88.4% tandis qu'avec des dies de 1×1 mm, le yield est de 99.5 %.

Il est donc doublement intéressant d'avoir des dies de petite taille pour optimiser la production de puces électroniques. Néanmoins le fait de découper une puce complexe en plusieurs puces de taille plus petite nécessite de faire communiquer ces différentes puces entre elles, il faut donc rajouter des éléments de communication qui viennent prendre de la place supplémentaire et utiliser de l'énergie supplémentaire.

De plus, l'utilisation de chiplets peut permettre d'utiliser des dies de différentes finesses de gravure selon les fonctions permettant de moduler les coûts de la puce finale avec les performances.

Enfin, un autre aspect économique à voir est la complexité de développement de nouvelles fonctionnalités. Cela tend à avoir des sociétés spécialisées (ou en tout cas des start-ups au début) proposant des blocs de propriété intellectuelle (des fonctions) prêts à l'emploi. Par exemple un fabricant de processeur pourra se focaliser sur le développement du processeur en lui-même tout en achetant des dies pour des fonctions comme le PCI express, l'USB ou les contrôleurs DDR.

Pour faciliter l'interopérabilité de chiplets venant de fabricants différents, des acteurs majeurs comme Intel, AMD, ARM, Qualcomm, Samsung ou TSMC ont créé une norme de communication entre chiplets, l'UCIe (*Universal Chiplet Interconnect express*).

# 4. Deux exemples : AMD et Intel (Altera)

## 4.1. AMD EPYC

Aujourd'hui de plus en plus de processeurs utilisent cette technique des chiplets. AMD utilise les chiplets depuis la première génération des processeurs EPYC, où les différents cœurs sont reliés entre eux par l'*Infinity Fabric*.

La première génération de processeurs EPYC voyait un ensemble de dies que l'on pourrait assimiler à des processeurs complets étant reliés entre eux par l'*Infinity Fabric* pour former le processeur final. Les chiplets étaient donc une sorte de petit processeur autonome: chaque die gérait ses entrées/sorties et avait son contrôleur DDR.
Ces dies, ou plutôt chiplets, comportent deux *Core Compute Complex* (CCX, un ensemble de quatre cœurs avec de la mémoire cache) ainsi qu'un contrôleur DDR, gère des entrées/sorties (PCI Express par exemple) et dispose de modules de communication pour l'*Infinity Fabric*.

Petite subtilité, il y a toujours quatre chiplets sur un EPYC de première génération. Pour faire varier le nombre de cœurs, AMD désactive des cœurs à l'intérieur des CCX. Par exemple pour avoir 24 cœurs, les CCX n'ont que 3 cœurs actifs

Cette première génération utilisait donc le principe des chiplets comme une sorte de copié/collé de dies au lieu de développer un die monolithique de grande taille.

Pour la seconde génération, AMD pousse le concept un peu plus loin. En effet, les CCX sont maintenant indépendants, regroupés par paires au sein d'un *Core Compute Die* (CCD) relié par *Infinity Fabric* à un die gérant la DDR et les entrées/sorties appelé *I/O Die* (IOD).
AMD exploite pleinement cette séparation accrue des fonctions. En effet le CCD est gravé en 7 nm tandis que l'IOD est gravé en 14 nm.

Ci-dessous une présentation d'AMD résumant le passage en chiplets des processeurs EPYC.

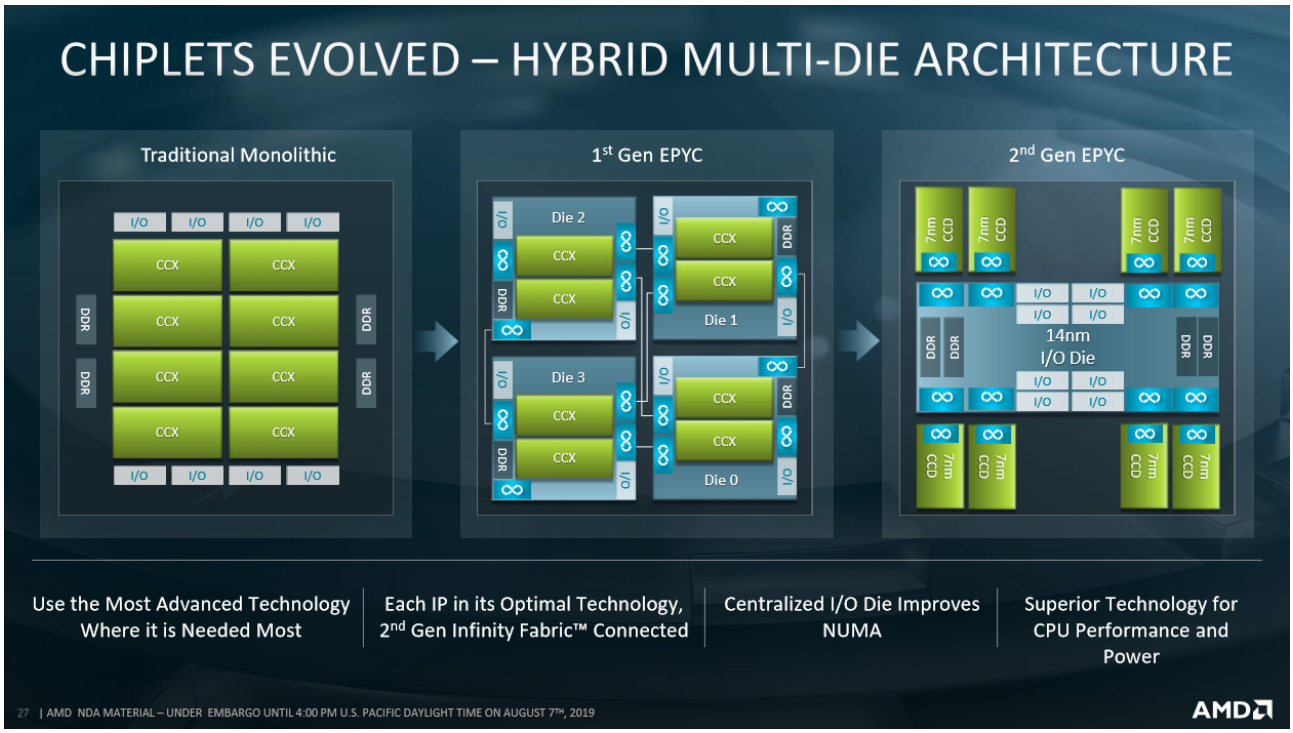


FIGURE 4.7. – Evolution de l'architecture des processeurs AMD (Source : AMD)

## 4.2. Intel FPGA (Altera)

Les processeurs Intel sont toujours des puces monolithiques sauf quelques exceptions comme nous avons pu le voir au début de cet article. Néanmoins dans le secteur des FPGA (des puces reconfigurables) Intel utilise les chiplets pour la dernière génération, les Agilex.
Ces chiplets concernent essentiellement le type de transceivers utilisés (les liens rapides) et sont appelées *Tiles*. Si Intel propose des gammes prédéfinies à partir de ces tiles, il doit être possible d'avoir des puces customisées pour ses propres besoins.
Les tiles sont divisées par vitesse maximum des transceivers et les protocoles supportés (Ethernet, PCI Express, etc.): 16G pour les P, 28G pour les H, 32G pour les R, etc.
Intel évoque aussi pour le futur la possibilité de connecter des chiplets customisés qui apporteraient des fonctions supplémentaires. Actuellement des sociétés ont sorti un chiplet ADC/DAC (Jariett Technologies) ainsi qu'un autre de connexion optique (Ayar Labs).

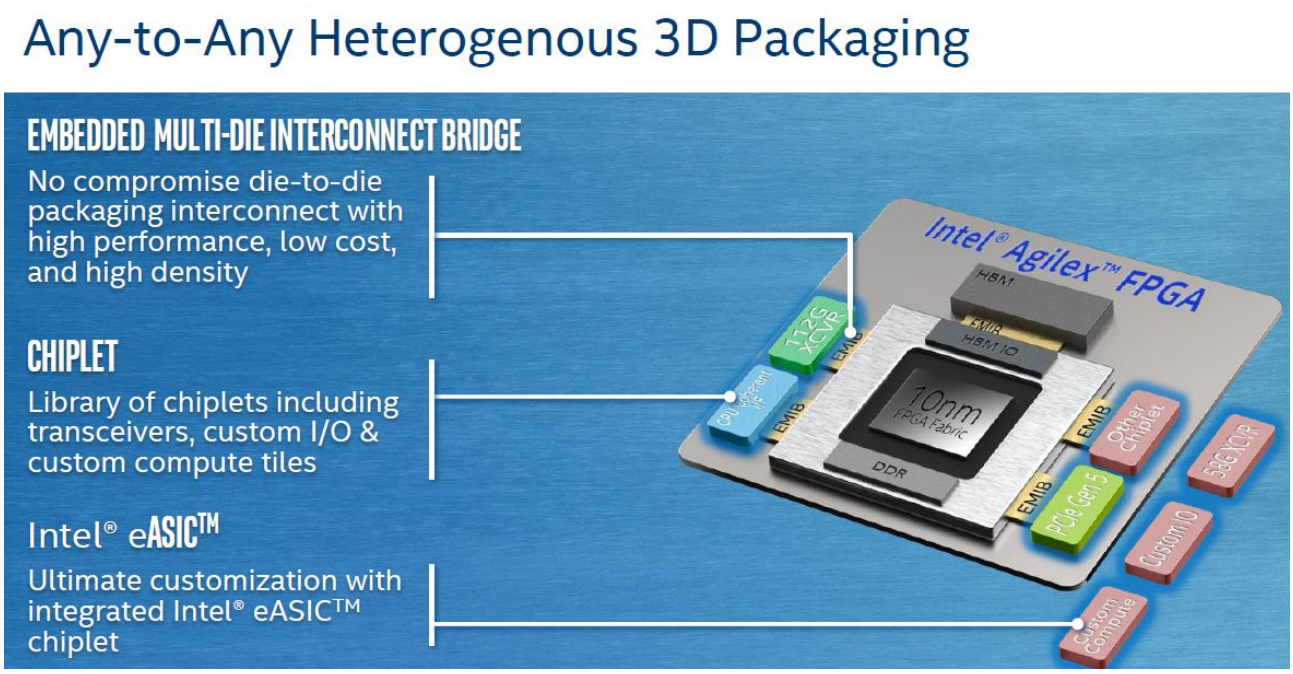


FIGURE 4.8. – Intel Agilex architecture (Source: Intel)

## Conclusion

Enfin il ne faut pas croire non plus que les puces *monolithiques* sont mortes. Elles ont toujours des avantages, notamment en terme de consommation et de latence de communication interne, ce qui peut s'avérer critique pour certaines applications nécessitant des puces de grande taille. C'est le cas de Broadcom et de ses puces de switch 400G dont le choix est expliqué par le concepteur dans cette vidéo: https://www.youtube.com/watch?v=B-COGMbaUg4

J'espère que cet article vous a plus et vous a permis d'en savoir un peu plus sur la fabrication des puces actuelles. J'ai tenté de vulgariser un sujet complexe, j'espère là aussi avoir pu réussir à vous garder après le premier paragraphe 🍊
N'hésitez pas à laisser un commentaire si certains points restent cryptiques pour vous, je tenterai d'apporter des précisions.