



Traitement, Synthèse, Technologie et Applications

BIARRITZ - Mai 1984 -

LE CALCULATEUR PARALLELE CAPITAN : 600 MIPS POUR L'IMAGERIE TEMPS REEL

THE CAPITAN PARALLEL PROCESSOR : 600 MIPS FOR USE IN REAL TIME IMAGERY

Gérard GAILLAT

MATRA Espace Produits & Technologie - B.P. 1 - 78146 - VELIZY

RESUME

L'architecture du calculateur parallèle CAPITAN est décrite dans cet article, notamment sa structure d'échange entre processeurs et avec les mémoires. Sa capacité d'accueil des circuits présents et futurs de traitement du signal est en particulier mise en valeur. Ses performances sont ensuite analysées dans le cas de l'utilisation du processeur TMS-320 comme élément de base.

Il en ressort que CAPITAN peut soit être couplé à un calculateur hôte et se comporter comme un array processor de très hautes performances, soit constituer, en tant que hardware flexible, le coeur d'une chaîne temps réel de traitement image. L'article se termine en passant en revue un certain nombre de ses applications potentielles. Pour chacune d'entre elles, une configuration est proposée et analysée. La variété des exemples choisis, empruntés aux domaines du traitement d'image, du signal, de la reconnaissance des formes et de la résolution d'équations aux dérivées partielles illustre la souplesse d'adaptation de la machine et démontre son aptitude à concurrencer les systèmes câblés en fonction des critères volume, performances, puissance dissipée.

Mots clés : Calculateur parallèle MIMD, bus en anneau, reconfigurabilité, digital signal processors, structure d'accueil pour CITGV, traitement du signal, traitement des images, équations aux dérivées partielles.

SUMMARY

The architecture of the CAPITAN parallel Processor is described in this paper, in particular its processor memory interconnection network. Its ability to incorporate present and future signal processing chips is emphasized. Its performances are then analyzed assuming TMS-320 as basic processing element.

It comes out that CAPITAN can be either linked to a host computer and act as a very high performances array processor or constitute, as flexible hardware, the heart of a real time image processing device. Finally some potential applications are analyzed. For each one, a configuration is proposed and discussed. The wide range of chosen examples which are taken from image, signal processing, pattern recognition and partial derivative equation fields illustrates machine adaptability and demonstrates that it is a cost and power consumption effective alternative to application dedicated hardware.

Key words : parallel MIMD processor, ring bus, reconfigurability, clusters of digital signal processors, welcome structure for WHSIC's, signal processing, image processing, partial derivative equations.



1 - INTRODUCTION

L'existence aujourd'hui, sur le march , de processeurs int gr s de traitement du signal permet de r aliser des architectures programmables dont les caract ristiques en performances, en volume ou en puissance dissip e, sont tout   fait comparables   celles d'architectures plus sp cialis es. Par contre, leurs capacit s d'adaptation sont incomparables. Le calculateur parall le CAPITAN⁽¹⁾ d velopp  par MATRA⁽²⁾ est une structure d'usage g n ral capable d'accueillir un grand nombre de ces processeurs pour les faire coop rer dans le cadre de traitements parall lisables   tr s haut d bit. Une telle structure permet d'abaisser consid rablement le co t de d veloppement et d'industrialisation d'un syst me temps r el de traitement des images, de traitement du signal, de reconnaissance des formes, de r solution d' quations aux d riv es partielles, de calcul matriciel, ...

L'exemple des applications recens es par l'Agence Spatiale Europ enne¹ est caract ristique des capacit s d'adaptation de CAPITAN : Dans le cadre des missions d'observation de la terre, plusieurs applications n cessitent un traitement plus ou moins  labor  sur un flot de donn es qui se chiffre en millions, voire en dizaines de millions d'octets par seconde. On peut citer principalement la reconstruction d'images   partir du signal  mis par le radar   ouverture synth tique et la compression d'images mono ou multispectrales. Pour chaque application, plusieurs algorithmes et de nombreuses variantes sont possibles. Le choix tient compte des param tres que l'utilisateur veut optimiser. Le but du calculateur CAPITAN est de fournir une architecture unique, suffisamment g n rale mais aussi suffisamment efficace pour s'adapter   diff rentes applications, diff rents algorithmes, diff rents d bits, diff rentes caract ristiques de l'instrument d'acquisition, ...

En fait, la g n ralit  de l'architecture la rend utilisable dans des domaines qui d passent tr s largement le domaine de l'imagerie spatiale embarqu e. La vari t  des exemples choisis dans la suite de cet article en donnera un premier aper u.

Nous allons commencer par d crire bri vement l'architecture du calculateur. Le lecteur int ress  par une description plus approfondie ou par une justification des choix architecturaux pourra se reporter   2 ou   3. Nous  voquerons ensuite la mise en oeuvre de ce calculateur, soit en temps que "hardware flexible" dans une chaine de traitement temps r el, soit en temps qu'"array processor" coupl    un calculateur h te. Puis nous terminerons en passant en revue un certain nombre d'applications et en analysant pour chacune d'entre elles les possibilit s de mise en oeuvre, tant au niveau mat riel qu'au niveau logiciel.

(1) Le sigle CAPITAN signifie : Calculateur Parall le pour l'Imagerie, le Traitement du signal et l'Analyse Num rique.

(2) Le d veloppement a  t  partiellement financ  par l'Agence Spatiale Europ enne sous le contrat 4960/81/NL/HP(SC). Le nom initial du projet  tait HSPP : High Speed Programmable Processor.

2 - DESCRIPTION

2.1 - Principe d'architecture

Le calculateur parall le CAPITAN apparait comme une structure d'accueil, de type MIMD⁴ permettant   un nombre  ventuellement  lev  de processeurs identiques ou diff rents de coop rer dans le cadre d'un travail n cessitant une forte puissance de calcul. Comme l'indique la Figure 1, ces processeurs sont reli s entre eux, avec les m moires globales et avec les p riph riques d'entr e-sortie par une structure de communication. Les abonn s de cette structure peuvent  tre des microprocesseurs d'usage g n ral tels que 8086 ou 68000, des processeurs de traitement du signal tels que TMS-320 ou NEC 7720, des circuits ou des logiques sp cialis es dans telle op ration de traitement du signal (FFT par exemple), des m moires, des p riph riques, voire un calculateur h te universel.   terme, l'ambition de ce calculateur est de constituer une v ritable structure d'accueil pour les circuits int gr s   grande vitesse qui naissent des plans VHSIC am ricain, 5 me g n ration japonais ou CITGV fran ais.

  titre indicatif, les abonn s aujourd'hui disponibles comprennent notamment :

- un abonn  68000 dot  de 256 KO de m moire propre
- un abonn  grappe de processeurs de signal qui regroupe 4 ou 8 TMS-320
- des abonn s m moire (toute carte compatible VME)
- un abonn  m moire d'image assurant le couplage   une cam ra ou un moniteur
- un abonn  coupleur assurant une liaison haut d bit avec les calculateurs Vax ou Sei
- et plus g n ralement toutes les cartes commercialis es sur le bus VME (voir paragraphe 2.4).

2.2 - Structure de communication

La structure de communication se pr sente sous la forme de deux bus en anneau reconfigurables. Chacun des deux bus en anneau parcourt tous les abonn s.

La Figure 2 illustre la notion de bus en anneau. Un tel bus peut  tre consid r  comme un ensemble de cases qui se d placent toutes d'un cran   chaque

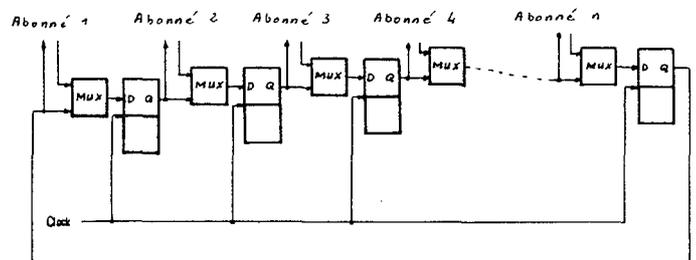


FIGURE 2 : PRINCIPE D'UN BUS EN ANNEAU

coup d'horloge. Si un abonn  veut envoyer un message   un autre abonn , il doit attendre le passage d'une case vise et y d poser le message. Au bout de n coup d'horloge, ce message passera devant le destinataire qui pourra le lire. Le protocole retenu sur ce bus est d crit dans⁵. Il a l'avantage de g rer le flot des  changes de mani re    viter toute collision au niveau des acc s m moires, afin de tirer parti

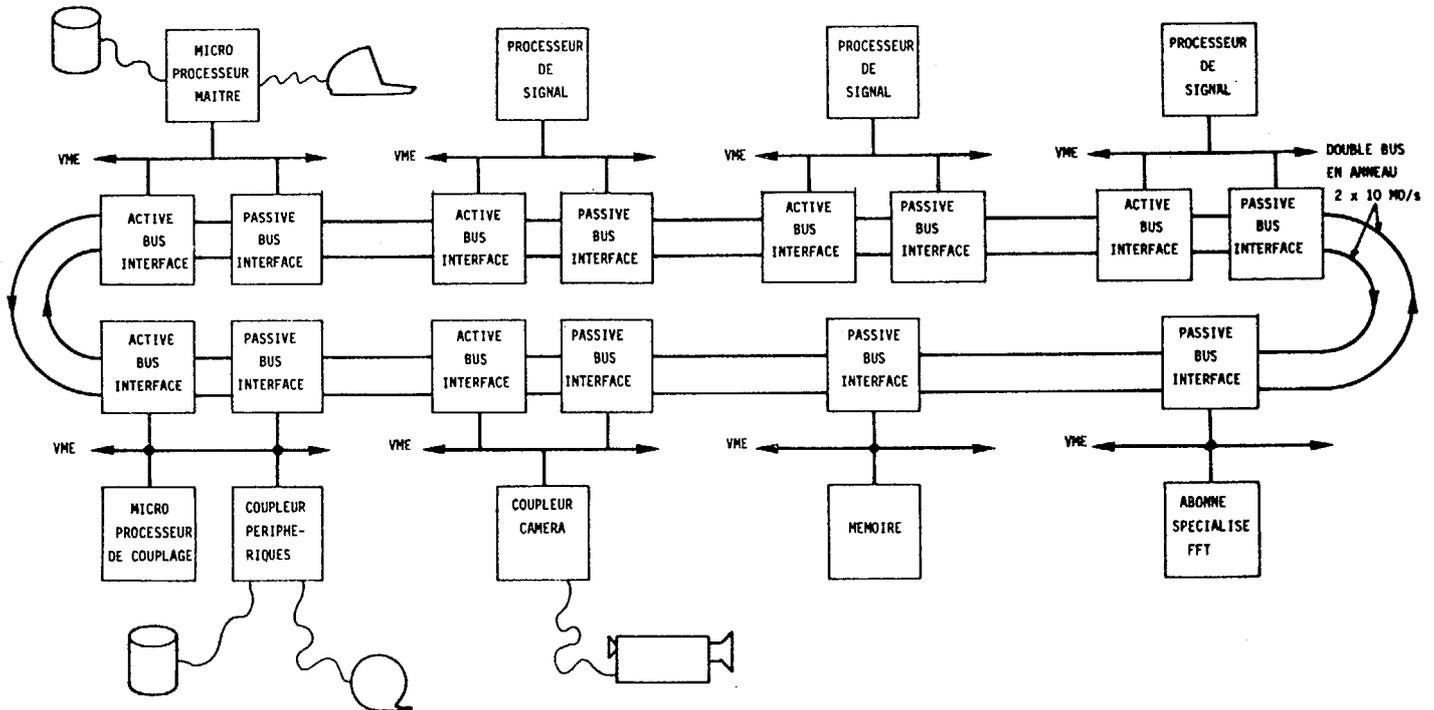


FIGURE 1 - ARCHITECTURE GENERALE DU CALCULATEUR CAPITAN

au maximum de la capacité d'un bus en anneau. De ce fait, le débit effectif du bus au voisinage de la saturation est égal à son débit nominal, soit 10 MO/s par bus dans le cas de CAPITAN.

Muni de ce protocole, le bus en anneau apparaît, vu de l'abonné, très semblable à un bus conventionnel 3 états. Aux performances près, il est capable d'assurer les mêmes services : Ordres de lecture, d'écriture, signal de priorité. Son intérêt essentiel tient à ses performances : Avec le protocole choisi, pour une même fréquence d'horloge, un bus en anneau écoule exactement le même débit qu'un bus 3 états. Par contre, pour des raisons électriques, il est possible dans une même technologie d'utiliser une horloge environ 4 fois plus rapide sur un bus en anneau. A titre d'exemple, l'horloge qui pilote les deux bus de CAPITAN fonctionne à 50 ns pour 16 abonnés en technologie TTL. En comparaison, la réalisation d'un bus 3 états reliant 16 abonnés et fonctionnant à 200 ns nécessite déjà beaucoup de soins et des circuits dissipant sensiblement plus de puissance. En d'autres termes, un bus en anneau permet, soit d'assurer plus de débit sur le même nombre de fils, soit un débit donné sur un moindre nombre de fils. Il se prête en outre bien mieux à une réalisation intégrée.

2.3 - Reconfiguration fonctionnelle et disponibilité

Chacun des deux bus en anneau peut être reconfiguré par logiciel en autant de sous-bus qu'il est nécessaire. Chaque sous-bus est lui-même capable d'écouler un débit de 10 MO/s, c'est-à-dire autant que le bus père.

Au total, si par exemple le premier bus est cassé en 3 et le deuxième en 4, le processeur est capable d'écouler un débit global d'échange de 70 MO/s, alors que sans cassure les deux bus n'auraient pu écouler que 20 MO/s. Cela autorise la création d'une grande variété d'architectures spécialisées. Plusieurs d'entre elles ont été décrites dans⁵ et sont illustrées par les Figures 3a, 3b, 3c. Le lecteur notera que dans chacune d'entre elles, chaque

abonné ne voit que deux bus. Simplement, chaque bus a été coupé en sous bus et ne parcourt de ce fait plus tous les abonnés.

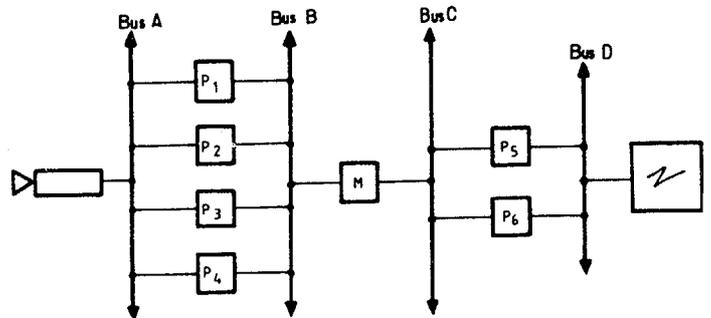


FIGURE 3.a - CONFIGURATION TYPE PIPELINE

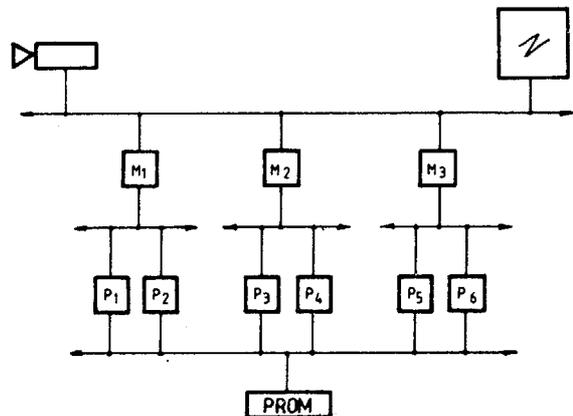


FIGURE 3.b - PROCESSEURS PARTAGEANT UNE PROM

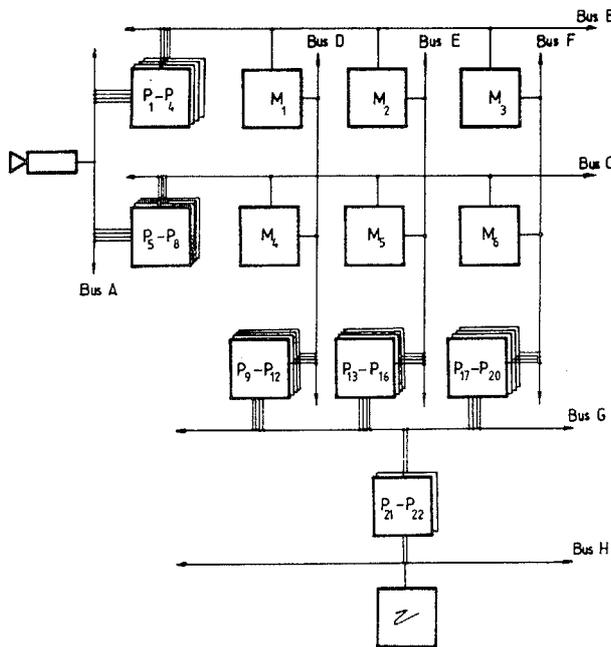


FIGURE 3.c - ARCHITECTURE BIDIMENSIONNELLE (exemple du SAR)

Légende des Figures 3.a, 3.b, 3.c :

- le symbole P représente un processeur avec un ABI et/ou un PBI
- le symbole M représente une mémoire avec un PBI
- tous les bus sont des bus en anneau.

En outre, la structure de communication est conçue de manière à assurer au calculateur une disponibilité élevée : Grâce à la reconfiguration des bus, tout abonné en panne peut être totalement isolé électriquement du reste des abonnés. Le système ne sera pas perturbé, quoiqu'il émette, ou même s'il met des fils du bus en court circuit. La sécurité ainsi offerte est sensiblement plus grande que ce qu'il est possible de réaliser avec un bus 3 états.

2.4 - Description des modules ABI et PBI

La connexion des différents abonnés aux bus en anneau s'effectue grâce à deux modules. L'un appelé Active Bus Interface (ABI) permet à un abonné d'émettre des ordres de lecture ou d'écriture sur l'un ou l'autre des deux bus. L'autre, appelé Passive Bus Interface (PBI) permet à un abonné de recevoir ces ordres, et de retourner une réponse en cas d'ordre de lecture. Typiquement, les processeurs sont reliés au bus par des ABI, tandis que les mémoires sont reliées au bus par des PBI. Néanmoins, dans certains cas, il peut être intéressant de relier les processeurs aux bus par des PBI ou à la fois par des ABI et des PBI. La Figure 1 en est un exemple. Nous en verrons d'autres exemples dans les applications.

De manière plus précise, chaque abonné processeur est organisé autour d'un bus compatible VME⁶ indépendant des autres abonnés. Les cartes ABI et PBI contiennent des mémoires à double accès, l'un étant réservé au bus VME, l'autre aux bus en anneau. Ces mémoires sont destinées à contenir aussi bien les ordres de transferts que les données. Vues du processeur, un module ABI ou PBI apparaît simplement comme un espace mémoire sur le bus VME.

Le rôle d'un ABI consiste à transférer à travers le bus en anneau des données depuis sa propre mémoire vers la mémoire d'un PBI (ordre d'écriture) ou à provoquer le transfert dans sa propre mémoire de données situées sur la mémoire d'un PBI (ordre de lecture). Dans les deux cas, le transfert se fait à l'initiative de l'ABI et selon les ordres de transfert contenus dans sa mémoire. Les transferts se font de préférence par bloc. Un bloc peut être à cheval sur plusieurs PBI. Trois types de transferts de bloc sont offerts par le module ABI :

- des transferts par bloc d'adresses séquentielles
- des transferts par bloc dont les adresses successives sont distantes de 2^n (colonne d'une image)
- des transferts par bloc donc les adresses sont aléatoires et indiquées dans l'ordre de transfert.

Ainsi, le mécanisme ABI-PBI est-il capable d'effectuer non seulement des transferts, mais aussi un certain réagencement des données lors des transferts. Bien entendu, tout ce travail s'effectue en total parallélisme avec les activités des abonnés processeurs. Ceux-ci sont totalement libérés des tâches d'échanges et peuvent se consacrer à 100 % aux tâches de calcul.

3 - MISE EN OEUVRE ET PERFORMANCES

Selon la nature des prestations qui lui sont demandées, le calculateur parallèle CAPITAN peut apparaître soit comme un array processor couplable à un calculateur hôte universel, soit comme un "hardware flexible" intégrable dans une chaîne de traitement temps réel.

3.1 - Configuration standard type array processor

La configuration standard type array processor se caractérise par le fait qu'elle contient seulement deux bus en anneau et, que chacun de ces bus parcourt tous les abonnés. Elle est illustrée par la Figure 1. Un système opératoire est en cours de développement sur cette configuration. Celui-ci suppose en outre que chaque abonné, processeur ou calculateur hôte, est relié aux bus en anneau par un module ABI et un module PBI. De ce fait, ces abonnés peuvent à la fois prendre l'initiative d'envoyer messages ou requêtes et répondre aux initiatives d'autres abonnés. Des abonnés mémoire peuvent également être inclus. Le nombre maximum de modules ABI ou PBI est de 32. Cela limite à 15 le nombre d'abonnés processeurs pouvant être accueillis. Si l'on considère le cas des abonnés grappe de 8 TMS-320 précédemment évoqués, la puissance de calcul s'élève alors à 600 Méga Instructions par seconde (600 MIPS). En effet, chaque TMS-320 exécute une instruction toutes les 200 ns, soit 5 MIPS. Une grappe de 8 fournit 40 MIPS et 15 grappes fournissent 600 MIPS. Bien entendu, cette puissance n'est atteinte que si l'algorithme présente un degré de parallélisme suffisant pour fournir à tout instant à chacun des 120 processeurs une tâche indépendante. C'est heureusement très souvent le cas dans les domaines d'application cités en introduction.

Dans le cas où les 20 MO/s disponibles sur les deux bus en anneau se répartissent équitablement entre données d'entrée et de sortie, les 600 MIPS correspondent à une moyenne de 60 instructions par octet traité. En configuration maximale, la machine est donc adaptée à des calculs de complexité moyenne ou élevée. Pour des calculs de faible complexité, la machine est limitée par ses capacités d'échange si bien que l'adjonction de processeurs supplémentaires n'augmente plus la puissance de calcul. On notera



cependant, si l'on tente une comparaison avec d'autres machines, que les 20 MO/s ne concernent que les échanges entre abonnés. La capacité totale d'échange de la machine est en fait beaucoup plus forte si l'on y ajoute les capacités des bus VME abonnés (5 MO/s chacun) et les capacités de dialogue entre les processeurs de signal et leurs mémoires privées.

3.2 - Configuration type hardware flexible

Si dans le cadre d'une application déterminée, la puissance de calcul ou le volume total d'échange entre abonnés dépassent les limites de la configuration array processor, il reste la solution de construire une configuration spécialisée. On peut alors accroître considérablement ces limites et dépasser largement le Giga instruction par seconde ou les 50 MO/s. En effet, les modules de base qui composent CAPITAN peuvent être agencés de multiples manières par simple modification du fond de panier, sans conception de nouvelles cartes logiques. Par exemple, les abonnés processeurs peuvent n'être couplés au bus que par des modules de type PBI. Ils apparaissent alors comme des mémoires intelligentes. L'utilisateur y dépose un bloc de données à traiter, attend un peu, teste un drapeau de fin de tâche et va récupérer les résultats. Ils peuvent aussi n'être couplés au bus que par des modules de type ABI. Le nombre de bus peut également dépasser 2 et l'on peut construire des structures pipeline ou bidimensionnelles ou mixtes ou ... (voir Figures 3.a, 3.b, 3.c). Un abonné peut même, au besoin, être relié à plus de deux bus. En effet, les ABI et PBI apparaissant comme des mémoires pour le bus VME, rien n'interdit de placer plusieurs ABI et/ou plusieurs PBI sur le bus VME d'un même abonné. Cela lui permet de dialoguer directement avec plus de deux bus en anneau. L'abonné caméra de la Figure 9 en est un exemple.

En outre, il peut être intéressant de construire des abonnés spécifiques d'une application. Par exemple, en traitement SAR, si 75 % des tâches de calcul sont des FFT et si le nombre de processeurs est élevé, il peut être intéressant de construire des abonnés FFT afin de diviser par 4 le nombre de processeurs. Ce point est analysé dans 7.

La démarche que nous proposons pour définir la configuration adaptée à une application consiste à raisonner en deux étapes. On regarde d'abord si les performances de la configuration array processor conviennent à l'application. Ensuite, si la réponse est positive, on peut éventuellement tenir compte des particularités de l'algorithme pour éliminer certains modules et réduire le volume du matériel et sa consommation. Sinon, il faut définir une configuration spécialisée type hardware flexible, en fonction de la nature de l'algorithme.

4 - EXEMPLES D'APPLICATION

Nous allons maintenant passer en revue un certain nombre d'applications qui illustrent la variété des algorithmes traitables par le calculateur CAPITAN. Nous commencerons par des applications qui relèvent de la configuration standard type array processor, puis nous décrirons des applications plus spécialisées.

4.1 - Restauration géométrique d'images

La restauration géométrique consiste à partir d'une image ayant subi des distorsions à recréer par rééchantillonnage une nouvelle image aux propriétés géométriques améliorées. Typiquement, en imagerie satellite, il s'agira de construire une nouvelle image superposable à une carte ou à une autre image.

La rotation d'une image est un cas particulier de restauration géométrique.

Pour obtenir une bonne qualité image, il faut choisir une interpolation bicubique qui consiste à calculer un point de coordonnées réelles en fonction de ses 16 plus proches voisins de coordonnées entières. Le calcul de chaque point peut se faire en deux passes (voir Figure 4) :

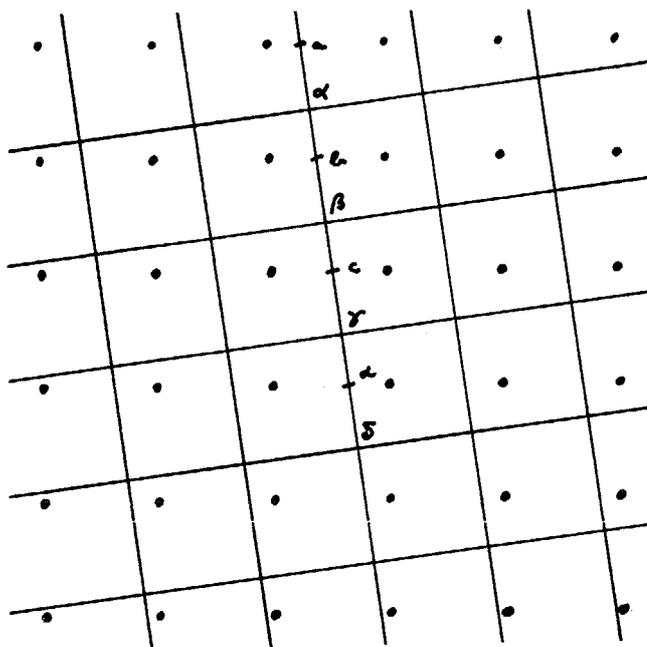


FIGURE 4 - INTERPOLATION GEOMETRIQUE : IL FAUT FORMER LA NOUVELLE GRILLE (, , , ...) A PARTIR DE L'ANCIENNE

Une première passe dans laquelle on rééchantillonne chaque ligne afin d'obtenir les points de type a, b, c, d une deuxième passe dans laquelle on rééchantillonne les points de chaque nouvelle colonne pour obtenir les points de type α , β , γ , δ .

Le traitement des lignes est totalement parallélisable ainsi que celui des colonnes. Le calcul d'un nouveau point nécessite le calcul d'une fonction de distorsion, le choix d'un jeu de 4 coefficients et un produit scalaire par le vecteur de ces 4 coefficients. Cela nécessite 4 μ s sur un TMS. Comme il faut deux passes, le calcul d'un nouveau point représente 8 μ s sur un TMS, soit 0.125 Mégapixel par seconde. Une configuration minimum avec 2 grappes de 8 processeurs, un abonné calculateur hôte et un abonné mémoire est donc capable de produire après rééchantillonnage 2 Méga pixels par seconde. Le plus simple, du point de vue organisation, consiste à effectuer tous les rééchantillonnages ligne en parallèle sur les 16 processeurs, à envoyer les résultats sur une mémoire puis à la relire pour effectuer les rééchantillonnages colonne. Le débit total d'échange formé de 4 flots de 2 MO/s est largement inférieur aux capacités d'échange de la machine. Par contre, une configuration avec 5 grappes de 8 processeurs traiterait 5 méga pixel par seconde et engendrerait un flot d'échange de 20 MO/s, ce qui est la limite de la configuration standard. Il ne servirait à rien pour cette application de mettre une 6ème grappe sur la configuration standard (Voir Figure 5).

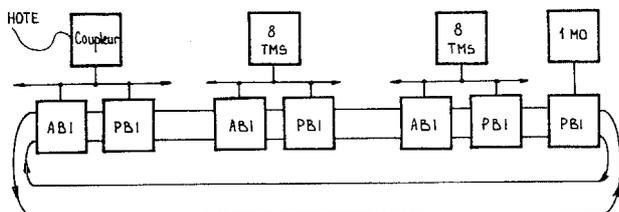


FIGURE 5 - RESTAURATION GEOMETRIQUE D'IMAGES A 2 MO/S

4.2 - Lecture optique de caract res

Prenons l'exemple d'une application de lecture des adresses en tri postal. Il s'agit de lire le code postal, le nom de la ville et de tenter une correction contextuelle en comparant les deux. La zone   balayer pour extraire la derni re ligne de l'adresse est de 6,4 cm en hauteur sur 12 cm en largeur. Avec une r solution de 8 points par millim tre, cela repr sente une grille de 512 x 960 pixels. La cadence typique d'un centre de tri est de 10 enveloppes par seconde, soit un volume   traiter de 4,8 MO/s. En outre, une enveloppe comporte au maximum 15 caract res   reconnaître.

La premi re  tape (pr traitement) consiste   balayer cette zone,   en identifier les points noirs,   faire un filtrage local, et   construire le rectangle circonscrit   chacun des caract res. Un algorithme tr s simple est possible   ce niveau. Le traitement d'un pixel sur un TMS peut  tre  valu    1.6  s en moyenne. Il faut donc 8 processeurs pour assurer ce traitement   la cadence de 4.8 MO/s.

La deuxi me  tape consiste   extraire un jeu de caract ristiques sur chaque caract re. Parmi les nombreuses techniques possibles   ce stade (voir⁸) nous consid rerons la m thode de Komori⁹. Cette m thode pr conis e par l'administration japonaise a l'avantage de donner des r sultats remarquables, y compris sur les caract res manuscrits. Nos  valuations conduisent   un temps de calcul de 4.5 ms pour le traitement d'un caract re sur un TMS. Les dix enveloppes par seconde repr santant 200 caract res par seconde peuvent donc  tre trait es par un seul processeur.

La phase de d cision peut se faire par une m thode de s paration lin aire. En utilisant la m thode pr conis e dans¹⁰, on devra effectuer par caract re environ 50 produits scalaires dans un espace de dimension 100, ce qui repr sente 2.5 ms sur un TMS. Ici encore, les 200 caract res par seconde peuvent  tre trait s par un seul processeur.

La correction contextuelle consiste   comparer le code et le nom de la ville   une liste de 25 000 bureaux distributeurs. S'il n'y a pas d'erreur, une comparaison suffit. S'il y a un caract re rejet , 10 ou 26 comparaisons sont n cessaires. S'il y a un caract re erron , environ 400 comparaisons sont n cessaires. Dans tous les cas, m me s'il y a plusieurs rejets, le temps de calcul est tr s faible, de l'ordre de quelques millisecondes au pire. Le processeur ayant assur  la d cision est parfaitement capable d'absorber ce surplus de travail, d'autant plus qu'il n'y a que 10 enveloppes par seconde   traiter. La liste des 25 000 communes ainsi que la table de d cision par apprentissage tient ais ment dans une m moire de 1 MO qui peut  tre locale   l'abonn  qui les consulte.

Une configuration   10 processeurs (voir Figure 6) est donc suffisante pour traiter cette application. Afin de ne pas saturer le bus VME de l'abonn  pr traitement, nous avons pr f r  r partir les 8 processeurs en 2 abonn s grappe de 4 processeurs au lieu d'une seule grappe de 8. Le dernier abonn  est l'abonn  extraction de caract ristique, d cision et correction contextuelle. Il comprend deux processeurs qui disposent de la m moire de 1 MO pr c demment mentionn e. En outre, un coupleur ligne asynchrone est pr sent sur cet abonn  pour l'envoi des r sultats.

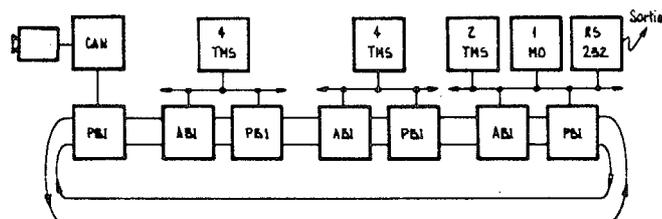


FIGURE 6 - TRI POSTAL A 10 ENVELOPPES PAR SECONDE 4.8 MO/S EN ENTREE

4.3 - R solution d' quations aux d riv es partielles

L'exemple que nous choisirons pour illustrer la capacit  de CAPITAN   r soudre des  quations aux d riv es partielles est emprunt    l' lectricit . Un champ de potentiel f sur un isolant est r gi par l' quation :

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = 0$$

Connaissant les contraintes aux limites, une m thode num rique pour r soudre l' quation consiste   it rer ind finiment en rempla ant   chaque it ration la valeur d'un pixel par la moyenne des valeurs de ses 4 voisins.

L'int r t de l'exemple est double. D'une part, il s'agit d'un type de calcul que l'on rencontre tr s fr quemment en r solution d' quations aux d riv es partielles ou en traitement d'images. Il s'agit de recalculer ind finiment un pixel en fonction de la valeur de ses quatre (ou huit) voisins. Les m thodes de squelettisation¹¹ ou de relaxation probabiliste¹² sont bas es sur ce sch ma. D'autre part, il s'agit d'un type de calcul qui semble poser probl me   CAPITAN. Il y a en effet beaucoup d' changes de donn es entre t ches pour peu de calculs.

L'approche pour les traiter est n anmoins fort simple et m rite que nous nous y attardions un peu. Tout d'abord, il faut consid rer des t ches qui consistent   traiter non pas un pixel, mais un groupe de pixels. Par exemple, un carr  32 x 32. Sur une image 1024 x 1024, le degr  intrins que de parall lisme de l'algorithme est de 10⁶, et m me en le r duisant ainsi d'un facteur 1000, il restera bien sup rieur au nombre de processeurs d'une configuration. Ensuite, il faut traiter ces carr s comme des damiers compos s de cases noires et de cases blanches. Chaque it ration se fera en 2 passes. La premi re calculera les cases noires de tous les carr s   partir des cases blanches, la deuxi me fera l'inverse.

Dans ces conditions, une passe sur un carr  32 x 32 par un TMS co te 1 ms. Une configuration de 16 processeurs partageant la m moire d'image effectu e donc une passe sur une image 1024 x 1024 en 62.5 ms et une it ration compl te en 125 ms, soit huit it rations   la seconde.

Si l'on place la m moire 1024 x 1024 sur un abonn  m moire du bus, le volume d' change processeur m moire s' l ve   16 MO/s.



LE CALCULATEUR PARALLELE CAPITAN : 600 MIPS POUR L'IMAGERIE TEMPS REEL
THE CAPITAN PARALLEL PROCESSOR : 600 MIPS FOR USE IN REAL TIME IMAGERY

C'est acceptable s'il y a 16 processeurs mais cela signifie qu'il ne servirait à rien d'en mettre plus de 20. La capacité mémoire des PBI ne permet pas non plus de distribuer 1 MO sur deux abonnés. La solution que nous proposons (voir Figure 7) consiste à doter chaque abonné d'une mémoire locale de 512 KO sur laquelle nous placerons une demi image par abonné. Seulement une frange d'une ligne à la frontière de chaque demi image sera dupliquée dans le PBI et régulièrement remise à jour par l'abonné, de manière à être lisible par l'autre abonné. Dans ces conditions, le débit d'échange entre abonnés chute à 32 KO/s. La

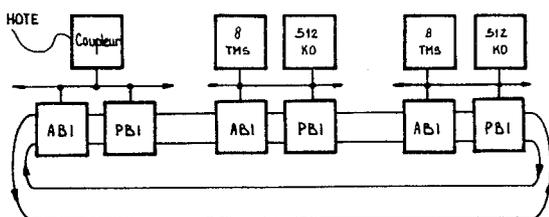


FIGURE 7 - EQUATION AUX DERIVEES PARTIELLES : 8 ITERATIONS PAR SECONDE SUR ZONE 1000'x 1000

configuration de la Figure 7 comprend un troisième abonné qui est un coupleur haute vitesse avec un ordinateur hôte. Les données d'entrée et les résultats en fin de calcul passent par lui.

4.4 - Compression d'image monospectrale

De nombreux algorithmes de compression d'image ont été publiés à ce jour. Cette variété s'explique pour une part par le fait qu'il s'agit d'un domaine en plein essor. Elle s'explique aussi et surtout par le fait que trois critères difficiles à concilier sont à optimiser : le taux de compression, la qualité image et la complexité algorithmique. Suivant ses objectifs, l'utilisateur privilégie l'un ou l'autre et préfère tel algorithme.

L'intérêt de CAPITAN est de proposer une architecture unique qui s'adapte à de nombreux cas. Reconnaissons cependant que dans le cas de codages très simplistes comme les DPCM qui ont été conçus pour réduire la complexité algorithmique, aux dépens des deux autres critères, les réalisations en matériel câblé conduisent à de meilleures performances.

Nous prendrons comme exemple typique celui de l'algorithme de compression monospectral conçu par l'Agence Spatiale Européenne¹³. Cet algorithme permet, pour une complexité raisonnable, d'obtenir un excellent rapport qualité-taux de compression. Il a notamment le mérite, par rapport au DPCM de ne pas déplacer les fronts radiométriques, ce qui est important pour les usagers utilisant la vision stéréo. Très sommairement, il consiste à découper l'image en carrés 8 x 8, à effectuer une transformée cosinus sur chaque carré et à coder chaque composante spectrale sur un nombre variable de bits.

Vu l'indépendance complète des traitements sur chaque carré, la mise en oeuvre sur CAPITAN de cet algorithme ne pose pas de problèmes. Les performances sont par contre intéressantes, notamment dans le cadre d'applications spatiales embarquées, dans la mesure où elles démontrent que CAPITAN permet la mise en oeuvre d'algorithmes de compression efficaces avec des puissances électriques raisonnables. Le calcul de la transformée cosinus bidimensionnelle d'un carré 8 x 8 sur un TMS 320 coûte 415 us. En y ajoutant la partie codage (concaténation des bits) la compression d'un carré 8 x 8 coûte 580 us. Un TMS 320 est donc capable de traiter 0.11 Méga pixels par

seconde.

Pour assumer le débit d'un satellite tel que SPOT, soit 4.6 Méga pixels par seconde, il faut 42 de ces processeurs. Ceux-ci peuvent être répartis en 6 grappes de 7 processeurs (plus éventuellement une 7ème pour assurer la disponibilité - Voir Figure 8).

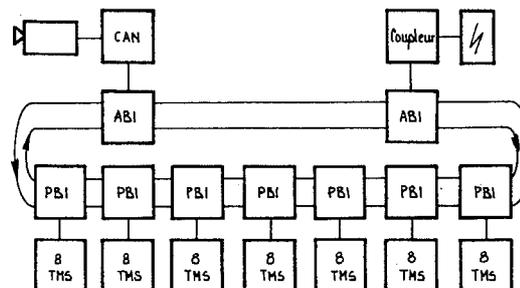


FIGURE 8 - COMPRESSION MONOSPECTRALE - SATELLITE SPOT : 4.6 MO/S

On notera que chaque grappe est seulement dotée d'un PBI : les abonnés processeur apparaissent donc comme une grande mémoire délocalisée écrite par l'organe d'entrée et lue par l'organe de sortie. Lorsque les modules ABI et PBI seront miniaturisés, chaque grappe consommera 9 W (voir⁷). L'ensemble des 6 grappes actives consommera donc 54 W, ce qui est tout à fait envisageable à bord. S'il fallait réduire encore, on pourrait remplacer la transformée cosinus par une transformée de Hadamard, mais au prix d'une détérioration du taux de compression de la qualité image.

4.5 - Compression d'image de télévision numérique couleur

Si l'on utilise pour la compression d'image de télévision numérique couleur un algorithme de compression d'images multispectrales tel que celui décrit en¹⁴, le traitement de chaque bloc de 8 lignes est une tâche indépendante. La mise en oeuvre serait donc très proche du cas précédent si le débit d'échange ne posait un problème nouveau. En effet, si l'on respecte la norme CCIR dite 4.2.2¹⁵, le débit moyen d'entrée est de 20,7 MO/s, réparti à raison de 50 % pour la luminance et 25 % pour chacune des deux valeurs de chrominance. A ceci, il convient d'ajouter, dans le cas d'une compression à 32 Mb/s, un débit de sortie de 4 MO/s. Dans ces conditions, on arrive à un trafic global de 25 MO/s qui dépasse les capacités du double bus en anneau.

La solution que nous proposons (Figure 9) consiste à casser l'un des deux bus en anneau en 3 sous bus et à répartir le flot d'entrée sur chacun de ces 3 bus. Rappelons que la cassure d'un bus peut se faire par logiciel sur la configuration standard, sans nécessiter le dessin d'un nouveau fond de panier. Le deuxième bus est affecté à la sortie. Il n'a pas été cassé puisqu'il ne doit acheminer que 4 MO/s. Pour varier par rapport à l'exemple précédent, nous avons cette fois relié les processeurs aux bus par des ABI et les organes d'entrée-sortie par des PBI. Les processeurs ont donc la charge de lire les données sur la mémoire d'entrée et d'expédier les résultats vers la mémoire de sortie. Nos études du problème conduisent au chiffre de 96 processeurs pour assumer le temps réel. Nous pouvons les répartir en 12 grappes de 8. L'organe d'entrée écrit 8 lignes sur un PBI puis passe au suivant et ainsi de suite. 6 PBI sont nécessaires pour l'entrée. La sortie est assurée au travers d'un



LE CALCULATEUR PARALLELE CAPITAN : 600 MIPS POUR L'IMAGERIE TEMPS REEL.

THE CAPITAN PARALLEL PROCESSOR : 600 MIPS FOR USE IN REAL TIME IMAGERY

Gérard GAILLAT

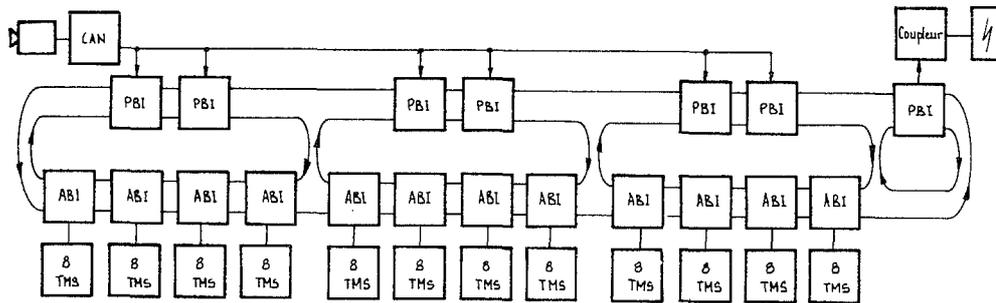


FIGURE 9 - COMPRESSION TELEVISION COULEUR : 20.7 MO/S EN ENTREE - 32 MB/S EN SORTIE

seul PBI, accessible depuis tous les processeurs à travers le bus qui n'a pas été cassé.

Remarquons que l'on peut transposer cet exemple dans le cas d'un compresseur d'image multispectral embarqué à bord de satellite. Il s'agirait d'un compresseur transformant un flot d'entrée de 166 Mb/s en un flot de sortie de 32 Mb/s. La consommation de la configuration de la Figure 9, instrument et antenne non compris peut être estimée à 140 W, si l'on dispose de modules ABI et PBI intégrés. Il s'agit d'un chiffre acceptable dans le cas de missions futures.

4.6 - Traitement Sar en temps réel

L'intérêt d'effectuer à bord la reconstruction d'image à partir du signal émis par le radar à ouverture synthétique tient au fait que la reconstruction est en soi une compression. L'exemple d'une chaîne bord construite à partir de CAPITAN est analysé dans⁷ et illustré par la Figure 10.

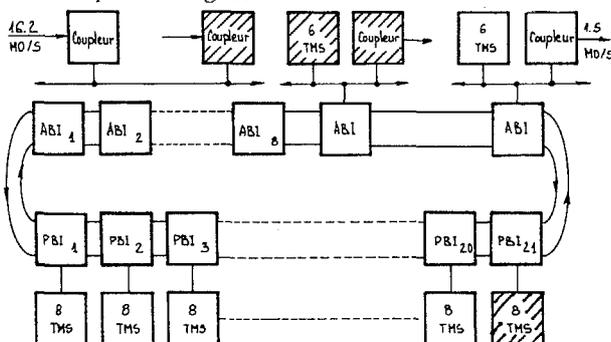


FIGURE 10 - TRAITEMENT SAR SUR SATELLITE ERS-1 EN HACHURE LES ELEMENTS REDONDANTS

Commentons le brièvement : Il suppose que le traitement en range est effectué sur une chaîne analogique. A la sortie de cette chaîne, le débit d'entrée pour CAPITAN est de 16.2 MO/s. En sortie de CAPITAN, le débit est de 1.5 MO/s. Toutefois, comme pratiquement aucun autre flux d'échange interabonnés n'est nécessaire, la structure à deux bus suffit à écouler le débit. 8 modules ABI dont un de réserve sont nécessaires pour écouler le débit d'entrée. 21 grappes de 8 TMS dont une de réserve sont nécessaires pour le traitement en azimuth. Un seul ABI dupliqué pour des raisons de disponibilité est nécessaire pour écouler le débit de sortie. Une grappe de 6 TMS localisée sur le même abonné assure une restauration géométrique de l'image sortante. Les 21 grappes effectuant les traitements en azimuth ne sont équipées que de PBI. En fait, la main corner turning memory (MCTM) et la re corner turning memory (RCTM) sont délocalisées sur ces 21 abonnés. Les ABI d'entrée écrivent la première, l'ABI de sortie lit la deuxième. Le traitement en azimuth consiste pour chaque processeur à lire son fragment de MCTM, à le traiter et à écrire

les résultats dans son fragment de RCTM. Il est intéressant de noter que le mécanisme ABI-PBI élimine totalement les conflits d'accès à des mémoires lors des transferts : La fonction corner turning memory est réalisable avec les éléments standards de CAPITAN^{3,7}. La consommation estimée de cette configuration est de 210 W, avec des possibilités de réduction si l'on dispose de circuits FFT câblés plus performants que le TMS⁷.

4.7 - Elaboration de produits image à bord

L'article¹⁶ indique ce que pourra être dans le futur un système embarqué de traitement d'image et décrit sa réalisation à partir de CAPITAN. La diffusion directe vers les usagers finaux implique en effet l'enchaînement d'un certain nombre de traitements : compensation radiométrique, géométrique, étiquetage des pixels. Une structure comme CAPITAN devient alors particulièrement attractive vu la variété des traitements à effectuer. L'article analyse en outre en détail les mécanismes de synchronisation entre tâches et plus généralement la mise en oeuvre logicielle de l'application sur la configuration proposée.

5 - CONCLUSION

Les différentes configurations et les différentes applications décrites ci-dessus illustrent à la fois la souplesse de CAPITAN et l'étendue de son domaine d'applications. Elles montrent que ce genre de machine est parfaitement capable de s'insérer dans une chaîne temps réel de traitement d'image fonctionnant à cadence vidéo. Si l'on excepte le cas des traitements très élémentaires (nous avons cité le cas du codage DPCM), le volume hardware d'une architecture basée sur CAPITAN est tout à fait comparable à celui d'une architecture câblée. La raison en est la suivante : Certes, l'utilisation de processeurs programmables conduit indiscutablement à une complexité hardware bien plus grande exprimée en nombre de transistors ; mais cette complexité est largement compensée par l'utilisation de circuits commerciaux d'un niveau d'intégration et de performances auquel bien peu d'utilisateurs ont accès.

A l'avenir, l'aspect "structure d'accueil" de la machine garantit l'évolution de ce produit car elle lui permettra de bénéficier pleinement de l'amélioration technologique de ces circuits. Par exemple, l'arrivée prochaine d'équivalents CMOS du TMS-320 augmentera de 66 % sa puissance de calcul et divisera probablement par 2 ou 3 sa consommation. A plus long terme, la machine pourra accueillir des circuits VHSIC, CITGV ou 5ème génération. De plus, l'effort technologique prévu par l'Agence Spatiale Européenne pour la miniaturisation du calculateur en fera un outil encore beaucoup plus attractif. A titre d'exemple, nous avons vu qu'après miniaturisation, l'application Sar embarqué ne consomme que 210 W. 70 % de cette puissance est due aux seuls



LE CALCULATEUR PARALLELE CAPITAN : 600 MIPS POUR L'IMAGERIE TEMPS REEL

THE CAPITAN PARALLEL PROCESSOR : 600 MIPS FOR USE IN REAL TIME IMAGERY
Gérard GAILLAT

TMS 320⁷. Tous les frais généraux incluant les modules ABI, PBI, les mémoires et les coupleurs ne représentent que 30 %. Quel intérêt économique y aurait-il alors à développer une architecture particulière sachant que l'optimisation ne pourra porter que sur une fraction de ces 30 % ?

REFERENCES

- 1 ESTEC : Study of programmable high speed processor for use on board satellite".
ESA call for competitive offer n° A0/1/1378/81/NL/HP (SC), Sept. 81.
- 2 GAILLAT G. : The Design of a parallel processor for image processing on board satellites : An application oriented approach
10 th Annual Symposium on Computer Architecture Stockholm 1983.
- 3 MATRA : HSPP architecture report
MATRA internal report n° EPT/063/339/83
- 4 Flynn M.J. : Some computer organizations and their effectiveness
IEEE transactions on Computers, C-21, Sept. 72 pp. 948-960
- 5 GAILLAT G. : CAPITAN : Un calculateur parallèle pour le traitement d'images embarqué à bord de satellites
4ème congrès Reconnaissance des formes et intelligence artificielle, Paris, 1984
- 6 MOSTEK, MOTOROLA, SIGNETICS, PHILIPS : VME bus specification manual 1981
- 7 OKKES R., GAILLAT G., SCHOTTER R. : On board Sar Processing
ISPRS congress, Rio de Janeiro 1984
- 8 GAILLAT G., BERTHOD M. : Panorama des techniques d'extraction de traits caractéristiques en lecture optique des caractères
2ème congrès Reconnaissance des formes et intelligence artificielle, Toulouse 1979
- 9 NOURORI K., KAWATANI T., ISHII K., IIDA T. : A feature concentration method for character recognition IFIP World Congress Toronto 1977
- 10 GAILLAT G. : Méthodes statistiques de reconnaissance des formes
Cours de l'Ecole Nationale Supérieure des Techniques Avancées, 1983
- 11 FAUGERAS O.D. and BERTHOD M. : Improving consistency and reducing ambiguity in stochastic labelling : An optimization approach"
IEEE Trans PAMI Vol PAMI 3 n° 4 July 81, pp 412-424
- 12 STEFANELLI R., ROSENFELD A. : Some Parallel Thinning Algorithm for Digital Pictures
JACM Vol. 18, n° 2, April 71 - pp. 255-264
- 13 ESTEC : Three image compression algorithms
ESTEC Internal Report THA/DC/1177/AV, 1983
- 14 LOWITZ G.E., CASSAGNE G. : CLADYNE : Un nouveau compresseur pour la télévision digitale
1er colloque image, Biarritz 1984
- 15 CCIR : Codage numérique des signaux de télévision couleur
Rapport CCIR R 629-2, 1982
- 16 DE GAVRE J.C., OKKES R., GAILLAT G. : Study of a programmable high speed processor for use on board satellites
4th symposium on Computers in Aerospace, Hartford 1983.