



**HAL**  
open science

# Langages de programmation en commande numérique de machines-outils

René M. G. David

► **To cite this version:**

René M. G. David. Langages de programmation en commande numérique de machines-outils. Modélisation et simulation. Université Joseph-Fourier - Grenoble I, 1969. tel-00281601

**HAL Id: tel-00281601**

**<https://theses.hal.science/tel-00281601>**

Submitted on 23 May 2008

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# THÈSES

*présentées*

A LA FACULTÉ DES SCIENCES DE L'UNIVERSITÉ DE GRENOBLE

*pour obtenir*

LE GRADE DE DOCTEUR ÈS SCIENCES PHYSIQUES

PAR

**René DAVID**

Ingénieur A.M. - Ingénieur I.A.G.

---

## DEUXIÈME THÈSE

Langages de programmation en commande numérique  
de machines-outils.

*Soutenues le 24 Avril 1969, devant la Commission d'Examen*

### JURY

MM. L. NÉEL

*Président*

M. FALLOT  
J. KUNTZMANN  
J. LAGASSE  
R. PERRET

} *Examineurs*



# THÈSES

*présentées*

A LA FACULTÉ DES SCIENCES DE L'UNIVERSITÉ DE GRENOBLE

*pour obtenir*

LE GRADE DE DOCTEUR ÈS SCIENCES PHYSIQUES

PAR

**René DAVID**

Ingénieur A.M. - Ingénieur I.A.G.

---

## DEUXIÈME THÈSE

Langages de programmation en commande numérique  
de machines-outils.

*Soutenues le 24 Avril 1969, devant la Commission d'Examen*

### JURY

MM. L. NÉEL

*Président*

M. FALLOT  
J. KUNTZMANN  
J. LAGASSE  
R. PERRET

} *Examineurs*



LANGAGES DE PROGRAMMATION EN  
COMMANDE NUMERIQUE DE MACHINES-OUTILS.

---

|   | Page |
|---|------|
| 1 Généralités sur la commande numérique.  | 1    |
| 2 La programmation en commande numérique. | 4    |
| 3 Le langage APT                          | 6    |
| 3.1 Généralités.                          | 6    |
| 3.2 Structure du langage APT.             | 7    |
| 3.3. Description de la géométrie.         | 10   |
| 3.4 Description des déplacements.         | 13   |
| 3.5 Description de l'usinage.             | 14   |
| 3.6 D'autres instructions.                | 16   |
| 3.7 Exemple de programme.                 | 17   |
| 4 Les autres langages.                    | 18   |
| 5 Evolution et Recherche.                 | 23   |
| 6 Normalisation.                          | 25   |
| 7 Conclusion.                             | 26   |

---



LANGAGES DE PROGRAMMATION EN  
COMMANDE NUMERIQUE DE MACHINES-OUTILS

---

1. GENERALITES SUR LA COMMANDE NUMERIQUE. [12,16,37]

C'est pendant la guerre de 1939-1945 que la commande numérique a été créée, pour résoudre le problème d'usinage particulièrement difficile de cames tri-dimensionnelles de pompes d'injections de moteurs d'avions.

La technique utilisée a été ensuite généralisée à des machines et des problèmes d'usinage plus simples.

Si la fabrication en grande série justifie des machines et des montages spéciaux, on s'est rendu compte, aux Etats Unis d'abord et en Europe plus récemment, de l'intérêt que pouvait présenter la commande numérique pour les petites et moyennes séries.

Les divers éléments de la machine à commande numérique, tels que table de travail et broche porte-outil sont actionnés par des servomécanismes dont les différentes grandeurs de consigne sont généralement données par une bande perforée qui indique aussi les vitesses de coupe et d'avance, la lubrification etc.

La commande numérique présente un certain nombre d'avantages que l'on conçoit bien : un gain de temps dû à la suppression d'opérations manuelles de réglages, de changements d'outils, de contrôle etc. (le rapport temps de coupe/temps d'arrêt étant de l'ordre de 10) ; une diminution des erreurs de réglages que peut faire tout opérateur humain ; une grande souplesse dans les

fabrications (une modification de la bande est plus rapide et moins coûteuse que l'élaboration d'un nouveau montage d'usinage). Mais la commande numérique a bien d'autres influences :

la surface nécessaire à l'entreprise est diminuée du fait du rendement des machines; la suppression des montages d'usinage modifie le travail du bureau des méthodes; la gestion est également affectée notamment par la diminution du nombre de pièces en cours de travail.

Quant au constructeur de machines-outils il doit aussi concevoir en fonction de la commande numérique. Les machines doivent être plus rigides et précises car il n'y a plus de contrôle de dimension en cours d'usinage. On a vu naître des machines universelles qui travaillent suivant 4, 5, 6 "axes" et que l'on appelle centres d'usinages.

La commande numérique constitue donc une évolution dans les travaux d'usinage, évolution favorisée par le développement <sup>considérable</sup> des calculateurs numériques qui préparent le travail de la machine.

En 1967 la France comptait 400 équipements de C.N. l'Allemagne 800, la Grande Bretagne 1200 et les Etats Unis 14000. Aux Etats Unis on évaluait qu'en 10 ans le nombre de pièces usinées par commande numérique passerait de 25% à 75%.

La classification des machines-outils à C.N. peut se faire suivant plusieurs critères. Nous en citerons deux :

#### Mode de fonctionnement de la machine.

Point à point. Le positionnement est discontinu, la trajectoire entre 2 points n'a pas d'importance (perçage).

Contournage. Les vitesses suivant les différents axes commandés sont asservies. On peut engendrer des surfaces compliquées.

Le fonctionnement paraxial est un intermédiaire : fonctionnement en point à point ou déplacement parallèle aux axes.

Nombre d'axes de la machine

Il faut noter la différence entre une machine de contour-nage à 3 axes asservis (3 axes) et une machine qui engendre les surfaces par courbes de niveaux (2 axes 1/2).

La figure 1 représente un centre d'usinage à 6 axes A,B,X,Y,Z,W.

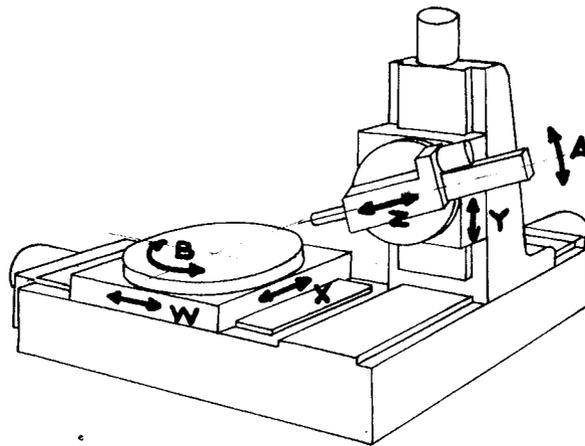


FIGURE 1 . Centre d'usinage à 6 axes

## 2. LA PROGRAMMATION EN COMMANDE NUMERIQUE. [ 35,46 ]

La programmation d'une machine à commande numérique peut être faite de façon entièrement manuelle. On peut afficher les cotes à l'aide de commutateurs, touches, fiches etc.

Cependant dès qu'une pièce doit être fabriquée en série dépassant quelques unités ou que l'usinage comporte un grand nombre d'opérations, même très simples, cette programmation devient fastidieuse. On entre alors les données par bande perforée. 80 à 85% des entrées sur machine à C.N. se font par bande perforée, le reste se répartissant entre entrées manuelle et par bande magnétique (la supériorité de la bande perforée sur la bande magnétique dans ce domaine est due principalement au fait que les précautions que nécessite l'emploi de la bande magnétique sont difficilement compatibles avec les conditions de l'atelier).

Plusieurs types de bandes perforées ont été utilisés mais une normalisation s'imposait pour une interchangeabilité entre les machines de caractéristiques compatibles.

L'Electronic Industries Association a proposé un premier code normalisé, connu sous le nom de code EIA. C'est un code à 6 bits d'information à parité impaire dont le support est la bande de 25,4 mm à 8 bits. Ce code est actuellement encore le plus utilisé.

L'organisation internationale de normalisation qu'est l'ISO a élaboré un code normalisé qui est de plus en plus utilisé. C'est un code à 7 bits d'information à parité paire dont le support est également la bande d'un pouce à 8 bits (C'est un aménagement du code américain ASCII étudié pour sa compatibilité avec les équipements de traitement de l'information).

Les caractères sont donc normalisés, mais aussi le format

qui peut être fixe ou variable.

Le format fixe est applicable aux petites machines. Chaque bloc contient exactement le même nombre de mots qui sont lus simultanément.

Le format variable permet de n'indiquer que les informations du bloc qui ont changé depuis le bloc précédent. Les vitesses d'avance et de coupe par exemple n'ont pas à être répétées. Le système de lecture est plus complexe mais la préparation de la bande est moins longue.

Le contournage nécessite un grand nombre de points pour définir une trajectoire si on la veut suffisamment précise. Pour que l'outil décrive un segment de droite par exemple, il faudra définir des points intermédiaires entre les extrémités connues par leurs coordonnées. Aussi, afin de diminuer le traitement "externe" et la quantité d'informations à introduire, certains équipements de C.N. comportent-ils des interpolateurs linéaires et/ou circulaires (et une correction d'outil).

Malgré cela, quand l'élaboration de la bande perforée devient compliquée, donc fastidieuse et source d'erreurs (définitions géométriques complexes mais aussi calculs trigonométriques pour des travaux de point à point), on éprouve le besoin de la faire préparer par un calculateur numérique auquel on fournira des indications en langage symbolique!

Les instructions élaborées de la bande perforée sont obtenues à partir des instructions symboliques par l'intermédiaire de 2 programmes : d'abord le processeur ou programme de traitement indépendant de la machine-outil, ensuite le post-processeur ou programme d'adaptation.

Le processeur effectue des opérations arithmétiques et géométriques et calcule la trajectoire de l'outil en tenant compte de la forme de l'outil et des tolérances programmées (découpage

de la trajectoire théorique en segments de droites). On sort généralement le fruit de ce traitement sur bande magnétique (Cutter Location tape).

Le post processeur adapte les données précédentes à l'équipement de C.N. utilisé. Il fait le changement de coordonnées pièce-machine, vérifie que les trajectoires et vitesses demandées sont compatibles avec la machine, calcule les accélérations etc. et édite la bande perforée.

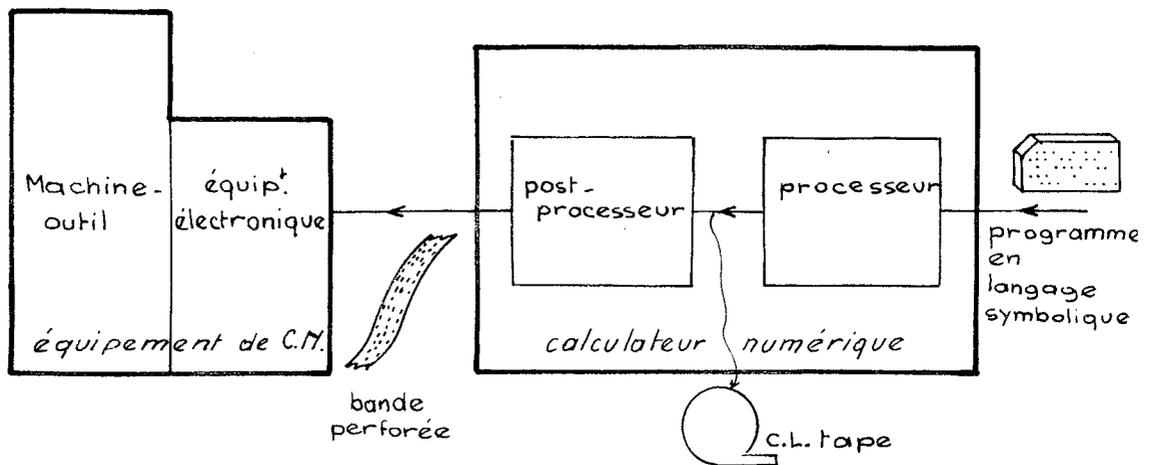


FIGURE 2 - Programmation en commande numérique.

### 3. LE LANGAGE APT. [38,39,41,53,54,55]

#### 3.1 GENERALITES.

Les langages de programmation en C.N. sont nombreux mais il en est un qui mérite un intérêt particulier c'est APT (Automatically programmed tools). Il est utilisé mondialement et a su faire l'unanimité pour certaines de ses qualités ; sa normalisation est proposée.

APT a été créé par le MIT (Massachussets Institute of Technology) en collaboration avec l'industrie aéronautique des Etats-Unis. Il poursuit actuellement son évolution sous l'égide de l'IITRI (Illinois Institute of Technology Research Institute).

De nombreux utilisateurs forment une association et cotisent pour l'amélioration du système APT, qui comprend le langage APT mais aussi un ensemble de programmes pour les ordinateurs qui exploitent le langage.

Afin de mise au point les concepts nouveaux de ce langage restent la propriété de l'IITRI pendant 2 ans.

Nous allons voir quelle est la structure du langage APT, qui est actuellement le plus évolué et qui permet de traiter tous les problèmes d'usinage à 3 dimensions, mais les principes généraux sont les mêmes dans l'ensemble des langages pour commande numérique.

### 3.2 STRUCTURE DU LANGAGE APT.

On dispose d'un certain nombre de caractères pour écrire en APT. Ces caractères comprennent les lettres majuscules de l'alphabet AB...Z, les chiffres décimaux 01...9 et des signes de ponctuation.

Ils servent à écrire les éléments du langage que sont les mots (plusieurs centaines) du dictionnaire APT, les symboles introduits par le programmeur et les nombres.

Ces éléments sont assemblés suivant une syntaxe et constituent des sortes de phrases, les instructions.

Les instructions judicieusement organisées permettent de décrire la géométrie de la pièce, la trajectoire de l'outil et les opérations d'usinage. On a alors un programme de pièce.

Afin que le programme soit facile à comprendre, un mot du langage APT<sup>est un</sup> mot de la langue anglaise, ou une abréviation car le mot APT ne peut pas comporter plus de 6 lettres.

Les instructions sont construites de façon à suggérer des phases dans lesquelles des verbes, articles etc sont implicites ou remplacés par des signes.

Par exemple :

L1 = LINE / PT1, PT2

signifie :

L1 (symbole introduit par le programmeur) est la ligne droite passant par le point PT1 et le point PT2 (précédemment définis).

L'instruction est l'unité de base du programme. Elle se compose essentiellement de 2 sections séparées par une barre oblique.

La section de gauche indique le type d'instruction. Elle contient le mot le plus important de l'instruction, le mot majeur. La section de droite définit ou précise la section gauche. Elle peut comporter des symboles, des nombres, des mots mineurs ou modificateurs ... ou ne pas exister comme dans l'instruction

FINI

qui indique la fin du programme.

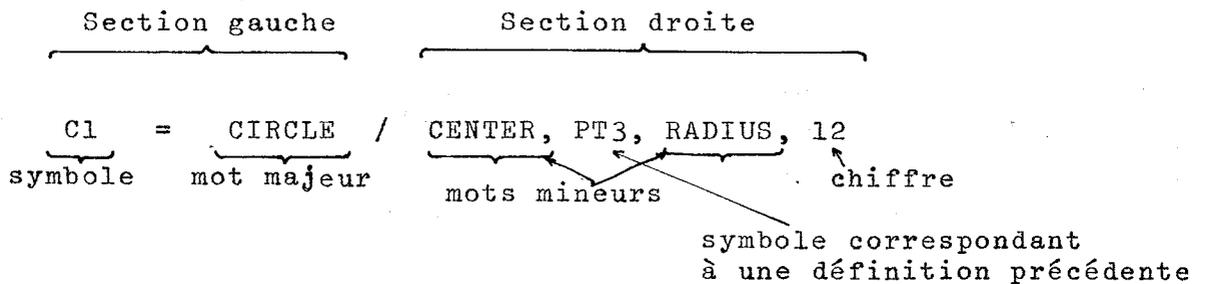


FIGURE 3 . Instruction .

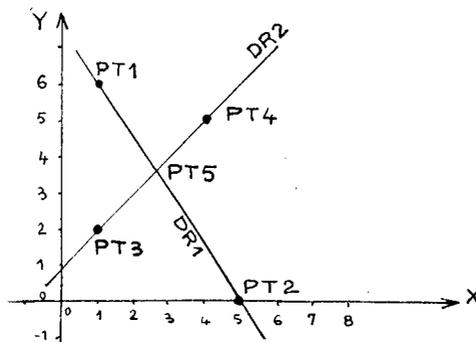
Le mot majeur peut être précédé d'un symbole<sup>choisi</sup> par le programmeur. Le cercle (fig. 3) dont le centre est le point PT3 et dont le rayon est 12 s'appellera désormais C1 et ne sera plus à définir si on veut s'y référer.

Les symboles peuvent également servir à étiqueter une instruction ou une valeur numérique. Ils ne doivent en aucun cas correspondre à un mot du vocabulaire APT.

Les signes de ponctuation ont des usages divers. Signalons le signe dollar qui indique que l'instruction se poursuit sur la ligne/carte suivante et le double signe dollar à droite duquel on peut mettre un commentaire quelconque qui n'aura pas d'action sur le programme (fig. 4).

Le programmeur qui a un grand nombre de fois à écrire certains mots APT peut les remplacer par des mots plus courts à sa convenance par l'instruction SYN (synonyme).

APT est un langage très souple qui laisse une grande "liberté de style" au programmeur. Les instructions imbriquées entre parenthèses en donnent un exemple. Les deux programmes de la figure 4 (et les intermédiaires qu'on peut imaginer) définissent en effet de façon équivalente le point PT5.



Programme 1

```
PT1 = POINT/1,6          $$ COORDONNEES X ET Y
PT2 = POINT/5,0
PT3 = POINT/1,2
PT4 = POINT/4,5
DR1 = LINE/PT1, PT2
DR2 = LINE/PT3, PT4
PT5 = POINT/INTOF, DR1, DR2  $$ INTERSECTION OF
```

Programme 2

```
PT5 = POINT/INTOF, (LINE/(POINT/1,6), (POINT/5,0)), $
(LINE/(POINT/1,2), (POINT/4,5))
```

FIGURE 4 .

Nous allons voir dans les paragraphes qui suivent les différents types d'instructions et essayer de faire transparaître les possibilités et l'adéquation du langage.

### 3.3 DESCRIPTION DE LA GEOMETRIE.

Le système APT est essentiellement à 3 dimensions. Il considère que la géométrie de la pièce et de l'outil sont tridimensionnelles. Ceci permet l'utilisation de machines à 3 axes et plus mais aussi de machines à 2 dimensions. La définition d'une droite ou d'un cercle est interprétée comme un plan ou un cylindre parallèle à l'axe Z.

Le choix pour les définitions géométriques est très grand. Il y a 10 façons de définir un point, 15 pour une droite, 12 pour un cercle etc...

#### Définition du point.

- 1) Par ses coordonnées cartésiennes.
- 2) Par ses coordonnées polaires.
- 3) Par l'intersection de 2 droites.
- 4) Par l'intersection d'une droite et d'un cercle.
- 5) Par l'intersection d'une droite et d'une conique.
- 6) Par l'intersection de 2 cercles.
- 7) Comme centre d'un cercle précédemment défini.
- 8) Par l'intersection d'une courbe quelconque et d'une droite.
- 9) Comme point d'un cercle avec indication de l'angle polaire.
- 10) Par l'intersection de 3 plans.

Nous allons voir dans les paragraphes qui suivent les différents types d'instructions et essayer de faire transparaître les possibilités et l'adéquation du langage.

### 3.3 DESCRIPTION DE LA GEOMETRIE.

Le système APT est essentiellement à 3 dimensions. Il considère que la géométrie de la pièce et de l'outil sont tridimensionnelles. Ceci permet l'utilisation de machines à 3 axes et plus mais aussi de machines à 2 dimensions. La définition d'une droite ou d'un cercle est interprétée comme un plan ou un cylindre parallèle à l'axe Z.

Le choix pour les définitions géométriques est très grand. Il y a 10 façons de définir un point, 15 pour une droite, 12 pour un cercle etc...

#### Définition du point.

- 1) Par ses coordonnées cartésiennes.
- 2) Par ses coordonnées polaires.
- 3) Par l'intersection de 2 droites.
- 4) Par l'intersection d'une droite et d'un cercle.
- 5) Par l'intersection d'une droite et d'une conique.
- 6) Par l'intersection de 2 cercles.
- 7) Comme centre d'un cercle précédemment défini.
- 8) Par l'intersection d'une courbe quelconque et d'une droite.
- 9) Comme point d'un cercle avec indication de l'angle polaire.
- 10) Par l'intersection de 3 plans.

Cette grande variété de définitions permet de traduire immédiatement l'idée du dessinateur sans faire les calculs qui seront le travail de l'ordinateur.

Par exemple si le dessinateur estime que l'angle d'une pièce doit être arrondi il raccordera les 2 côtés par un arc de cercle de rayon approprié sans calculer les coordonnées du centre du cercle. Eh bien le programmeur fera de même. Il définira un cercle de rayon donné et tangent à 2 droites.

Comme dans l'exemple précédent certaines définitions doivent être précisées quand il y a plusieurs solutions. On utilise des modificateurs XSMALL, XLARGE etc... Par exemple le point P6 de la figure 5 peut être défini par

P6 = POINT/YLARGE, INTOF, D4, C1

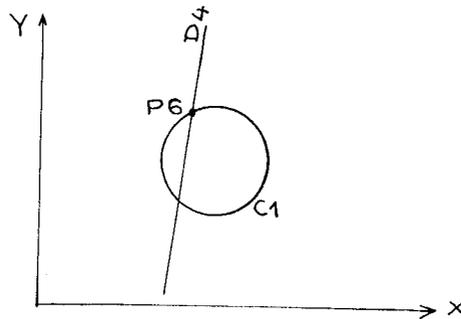


FIGURE 5 .

APT permet aussi la définition de coniques, de courbes expérimentales par l'instruction TABCYL (tabular cylinder) et divers moyens d'interpolation, de plans, de vecteurs, de quadriques, de surfaces de révolution, de surfaces réglées, et de réseaux de points par l'instruction PATERN.

L'instruction PATTERN permet de définir un réseau de points sur un cercle, une droite ou un parallélogramme. Sur la fig. 6, le réseau PAT 1 peut être donné par

$$\text{PAT 1} = \text{PATTERN/LINEAR, P1, P2, 5}$$

Il correspond à 5 points équidistants dont le premier est P1 et le dernier P2.

Supposons PAT 2 défini, l'ensemble des 15 points de PAT 3 est défini par

$$\text{PAT 3} = \text{PATTERN/PARLEL, PAT 2, PAT 1}$$

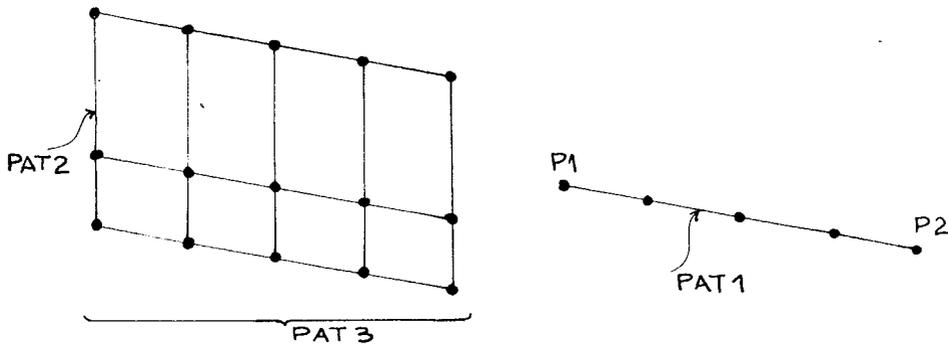


FIGURE 6 . réseaux de points

Si l'usinage ne doit pas se faire sur certains points du réseau, on utilisera le modificateur OMIT.

Dans le domaine de définition de la géométrie, signalons encore qu'on peut réaliser la translation, la rotation, la symétrie, le changement d'échelle, et tout changement de système de coordonnées.

### 3.4 DESCRIPTION DES DEPLACEMENTS.

En usinage point à point l'instruction

GOTO/point ou réseau de points

permet de presque tout faire.

D'une façon générale un mouvement élémentaire de contour-  
nage est défini par 3 surfaces (figure 7).

|                    |               |
|--------------------|---------------|
| Surface de pièce   | PART SURFACE  |
| Surface de guidage | DRIVE SURFACE |
| Surface d'arrêt    | CHECK SURFACE |

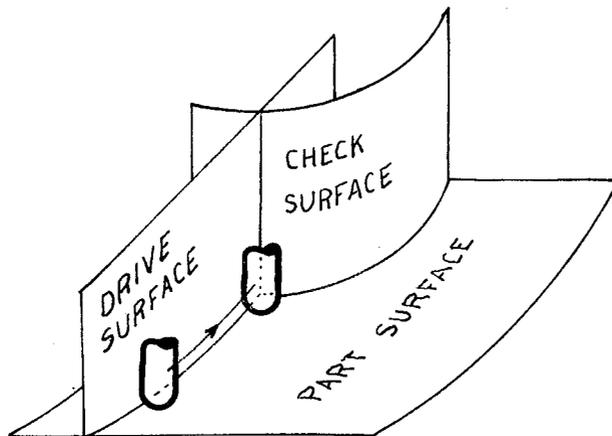


FIGURE 7 . Déplacement élémentaire .

Guidé par les surfaces de pièce et de guidage l'outil se dépla-  
ce jusqu'à la surface d'arrêt. Une autre instruction définit  
alors le mouvement élémentaire suivant. Elle est du type :

GORGT/SUR 1, TO, SUR 2

qui signifie : Aller vers la droite le long de la surface SUR 1  
(qui est la nouvelle surface de guidage et souvent l'ancienne  
surface d'arrêt) jusqu'à la surface SUR 2 (qui est la nouvelle  
surface d'arrêt).

On remarque que la surface de pièce n'est pas indiquée. Elle est précisée par une instruction spéciale (PSIS) mais quand on en change seulement.

La position relative par rapport à la surface d'arrêt est précisée par un modificateur TO, ON, PAST, (représentées à la figure 8) et TANTO (outil tangent à l'intersection des surfaces).

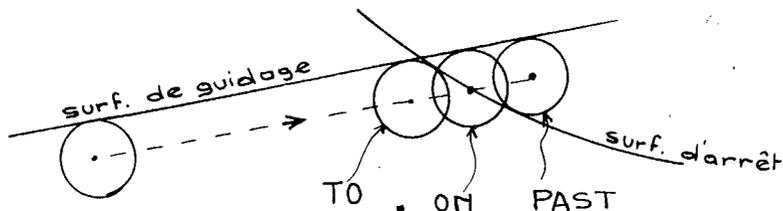


FIGURE 8 . Positions par rapport à la surface d'arrêt.

Des mots majeurs tels que TLRT (outil à droite de la surface de guidage) complètent les instructions GOUP (changement de direction vers le haut) GOFWD (continuer vers l'avant) etc. donnant au programmeur la possibilité de réaliser un programme plus simple.

Une instruction FROM donne le point de départ de l'outil, une instruction GO son premier mouvement (en direction d'un vecteur par exemple et jusqu'à une surface). La suite de sa trajectoire est définie par des instructions dont nous venons de parler (voir exemple au § 3-7).

### 3.5 DESCRIPTION DE L'USINAGE.

Le calculateur déterminera la trajectoire réelle de l'outil à partir des instructions vues aux précédents paragraphes, mais pour cela il lui faut d'autres renseignements.

Il faut que la forme de l'outil soit définie par exemple l'instruction

CUTTER/20, 6

définit les dimensions de l'outil représenté à la figure 9

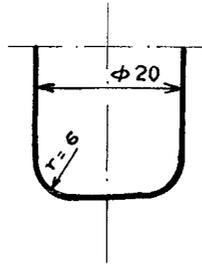


FIGURE 9 . outil

La trajectoire théorique est approchée par une succession de segments de droite. La détermination de ces segments est basée sur une tolérance sur la cote nominale.

OUTOL/a

INTOL/b

définissant la tolérance +a (extérieure) et -b (intérieure) -  
Voir figure 10.

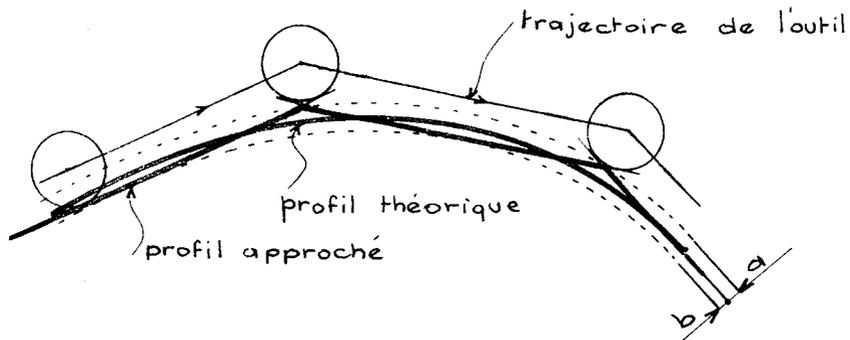


FIGURE 10 . Tolérances

Les conditions d'usinage sont données par des instructions dont, par exemple :

COOLNT/ON

Mettre en route l'arrosage de refroidissement (coolant)

FEDRAT/1000

Vitesse d'avance (feed rate) 1000 mm/mn

SPINDL/2000

Vitesse de broche (spindle speed) 2000 tours/mn

### 3.6 D'AUTRES INSTRUCTIONS

Les calculs algébriques et trigonométriques sont possibles. Dans toute instruction un nombre peut être remplacé par une expression à calculer, placée entre parenthèses, comme un symbole d'élément géométrique peut être remplacé par la définition de cet élément (c. f. page 9).

La répétition d'instructions peut se programmer de différentes façons parmi lesquelles le programmeur pourra choisir la plus astucieuse en fonction du problème.

Comme les langages de calcul scientifique ou de gestion APT permet de réaliser des boucles et des sauts dans le programme.

A l'aide d'une instruction MACRO on peut définir un ensemble d'instructions (une procédure) auquel on fera appel par le symbole qu'on lui aura affecté.

Les instructions COPY et GODLTA (GO delta) commandent aussi la répétition de la même série d'instructions.

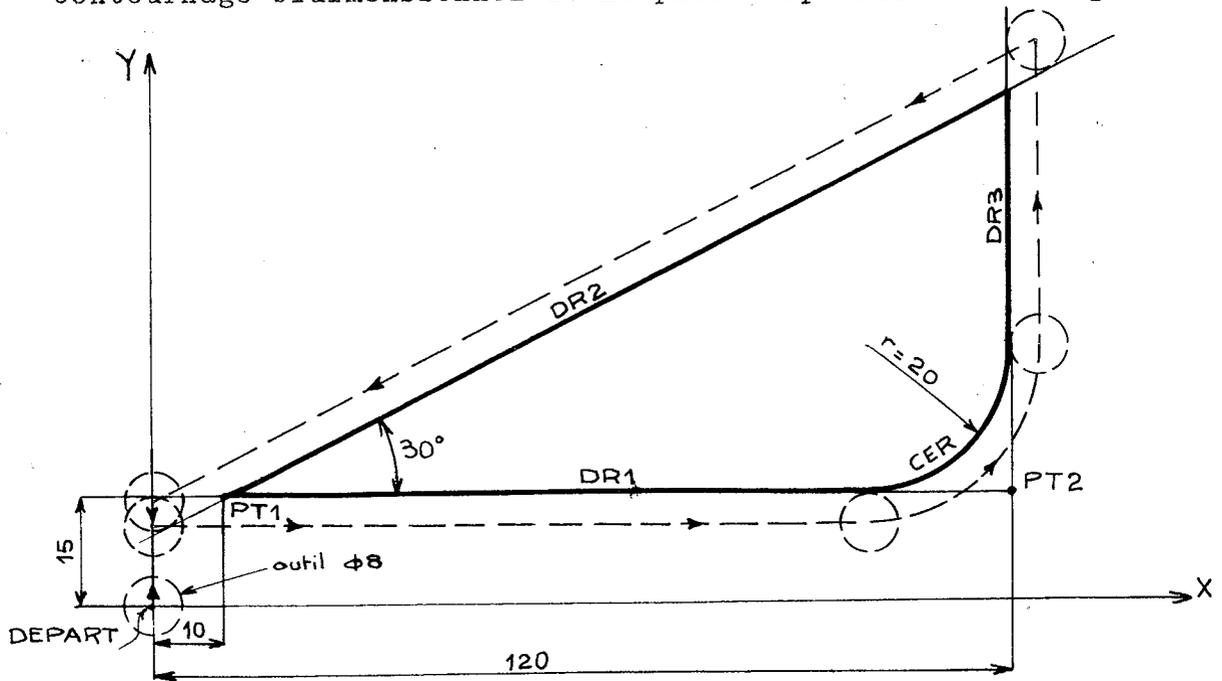
ZSURF définit la surface à partir de laquelle sera mesurée la cote Z. C'est généralement, mais pas obligatoirement, un plan parallèle à XY. Un évidement peut se définir directement par l'instruction POCKET.

Des instructions commandent l'impression, la perforation.

Le programme comprend encore un certain nombre de précisions dont celles qui sont nécessaires au post processeur. Il commence par PARTNO, instruction donnant la référence de la pièce. L'instruction MACHIN indique le post processeur à utiliser etc.

3.7 EXEMPLE DE PROGRAMME.

Contournage bidimensionnel de la pièce représentée à la figure 11.



```

PARTNO EXEMPLE DE PROGRAMME APT (1)
MACHIN/ABC (2)
CUTTER/8 (3)
OUTTOL/.02 (4)
## DEFINITIONS GEOMETRIQUES (5)

DEPART = POINT/0,0,0 (6)
PT1 = POINT/10,15,0 (7)
PT2 = POINT/120,15,0 (8)
DR1 = LINE/PT1, PT2 (9)
DR2 = LINE/PT1, ATANGL,30 (10)
DR3 = LINE/PT2,PERPTO,DR1 (11)
CER = CIRCLE/YLARGE,DR1,XSMALL,DR3,RADIUS,20 (12)
SPINDL/2000,CLW (13)
COOLNT/ON (14)
FEDRAT/800 (15)
FROM/DEPART ## DEBUT TRAJECTOIRE (16)
GO/TO, DR1 (17)
GORG/DR1,TANTO,CER (18)
GOFWD/CER,TANTO,DR3 (19)
GOFWD/DR3,PAST,DR2 (20)
GOLFT/DR2,ON,(LINE/DEPART,ATANGL,90) (21)
GOTO/DEPART (22)
COOLNT/OFF (23)
SPINDL/OFF (24)
FINI (25)
    
```

FIGURE 11 . Programme APT .

La plupart des instructions se comprennent grâce aux explications qui ont précédé et à l'intuition du lecteur qui en devine le sens. Nous allons toutefois en commenter quelques-unes (références indiquées à droite)

- (2) ABC est le nom imaginaire d'un post-processeur.
- (10) DR2 est la droite passant par le point PT1 et faisant un angle de  $+30^\circ$  par rapport à l'axe des X.
- (11) DR3 est la droite passant par PT2 et perpendiculaire à DR1.
- (12) CER est le cercle tangent à DR1, du côté où Y est le plus grand, et tangent à DR3, du côté où X est le plus petit, et de rayon 20.
- (13) Mise en rotation de la broche : 2000 tours/mn dans le sens des aiguilles d'une montre (Clock wise). Arrêt en (24).
- (14) On met en route l'arrosage qui sera arrêté par (23).
- (18) Aller vers la droite le long de DR1 jusqu'au point de contact (tangent to) avec CER.
- (19) Continuer en avant (go forward) le long de CER jusqu'au point de contact avec DR3.
- (20) Continuer en avant le long de DR3 jusqu'au delà de DR2.
- (21) Aller vers la gauche le long de DR2 jusqu'à ce que le centre de l'outil soit sur l'axe des Y.
- (22) Aller au point DEPART.

#### 4. LES AUTRES LANGAGES [24,41,46,57,58,59,60,61,64,68]

Il existe bien d'autres langages que APT. Le tableau ci-dessous donne les principaux et certaines de leurs caractéristiques.

| Langage      | Origine  | mémoire nécessaire, nombre de mots | calculateurs utilisables en France                        | nombre d'axes point à point:P contournage:C |
|--------------|--|------------------------------------|---|---|
| AUTOPROPS    | IBM<br>Pratt et<br>Whitney                     |                                    | IBM 1401  | 2 P   |
| AUTOSPOT     | IBM<br>Kearney et<br>Trecker pour<br>Milwaukee | 32 K<br>+ disques                  | IBM 1620<br>IBM 360/30                                    | 3 P   |
| CAMP I et II | IBM<br>Westinghouse                            |                                    | IBM 7090<br>7094  | 3 et 5 P                                    |
| MILMAP       | ICT<br>pour Milwaukee                          |                                    |   |   |
| PRONTO       | GE<br>pour Milwaukee                           |                                    | IBM 7090<br>GE 225  | 3 P   |
| PROFIL DATA  | Ferranti                                       |                                    | ICT série 1900  |   |
| PAGET        | Olivetti                                       |                                    | GE 425<br>IBM 7040<br>IBM 360/40-64K                      | C   |
| SPLIT        | Sundstrand                                     |                                    | IBM 7090<br>IBM 1620<br>IBM 360/30<br>IBM 650<br>IBM 7094 | 5 P et C                                    |
| AUTOPOL      | IBM  |                                    | IBM 360/30<br>IBM 1130                                    | 2 C<br>tournage                             |
| ROMANCE      | IBM  |                                    | IBM 1130  |   |
| AUTOPROMPT   | IBM  | 256 K<br>+ disques                 | IBM 7090  | 5 P et C                                    |
| AUTOMAP      | IBM  |                                    | IBM 1620  | 3 C   |
| MECA         | BULL G E                                       |                                    | BULL G E  | 3 P   |
| ..           | ..   | ..                                 | ..  | ..  |

|               |                            |                    |   |                         |
|---------------|----------------------------|--------------------|---|-------------------------|
| APT (III)     | MIT<br>IITRI               | 256 K<br>+ disques | IBM 360/40<br>IBM 7090<br>UNIVAC 1707<br>BULL GE 625<br>ICT<br>CDC 3600 | 5 P et C                |
| IFAPT         | France<br>CII<br>LCA       | 64 K               | CII   | P puis C<br>puis P et C |
| EXAPT I<br>II | Allemagne                  | 64 K à 124 K       | ICT<br>UNIVAC 1107  | 3 P<br>tournage         |
| 2 CL          | Grande-<br>Bretagne<br>NEL |                    | ICT   |                         |
| ADAPT         | Etats Unis<br>IBM          | 64 K<br>+ disques  | IBM 360/40<br>UNIVAC<br>BULL GE<br>HONEYWELL<br>CII                     | 3 P et C                |

La plupart de ces langages sont nés de besoins particuliers et d'une façon désordonnée. Certains d'entre eux sont intéressants et ont été très utilisés (AUTOSPOT, en particulier).

Mais le besoin de normalisation est vivement ressenti. Dans une entreprise on ne peut pas envisager de traiter chaque type de problème avec un langage différent. La formation des programmeurs, l'utilisation des calculateurs numériques, les échanges entre utilisateurs fournissent des arguments contre cette diversité.

Par ailleurs, la **richesse** du langage APT III (vocabulaire et syntaxe), qui est complet car prévu pour convenir à tous

les travaux, en fait un langage qui nécessite des équipements de calcul importants. Mais la plupart des travaux d'usinage ne demandent pas un outil aussi puissant et beaucoup de firmes ne peuvent pas disposer couramment des moyens de calculs correspondants.

Ces deux besoins, de normalisation d'une part, et de pouvoir utiliser des calculateurs de taille réduite d'autre part, ont conduit à créer une famille de langages qui sont compatibles avec le langage de référence. Leur capacité de définitions géométriques est réduite.

ADAPT (ADaptation of APT) a d'abord été étudiée pour le contourage à deux dimensions, par IBM sur la demande de l'US Air Force IFAPT en France, 2 CL en Grande Bretagne et EXAPT en Allemagne sont nés depuis. Nous reparlerons de ces langages, qui sont en pleine évolution au paragraphe suivant.

EXAPT (EXtended subset of APT) étudié par différents instituts Allemands (sous la direction des Professeurs OPITZ d'Aix-la-Chapelle et SIMON de Berlin) mérite une mention particulière. Son originalité est l'utilisation d'un processeur technologique après le processeur géométrique. Le matériau à usiner ayant été défini, il suffit de décrire une opération d'usinage principale (par exemple un taraudage) pour que le processeur technologique à l'aide d'un sous-programme sélectionne les opérations d'usinage intermédiaires (surfaçage, perçage, chanfreinage, puis taraudage) et choisisse les outils à utiliser (de nombreux équipements de commande numérique comportent un changeur automatique d'outils), les vitesses de coupe et d'avance adaptées, le liquide d'arrosage à utiliser.

EXAPT fait appel pour cela à 2 fichiers ou cartothèques qui comportent les renseignements nécessaires. Une cartothèque matériau (classés en 10 groupes) et une cartothèque outils.

Bien entendu ces traitements nécessitent des grandes capacités de mémoires (64 à 124K mots pour EXAPT I qui traite les problèmes de point à point).

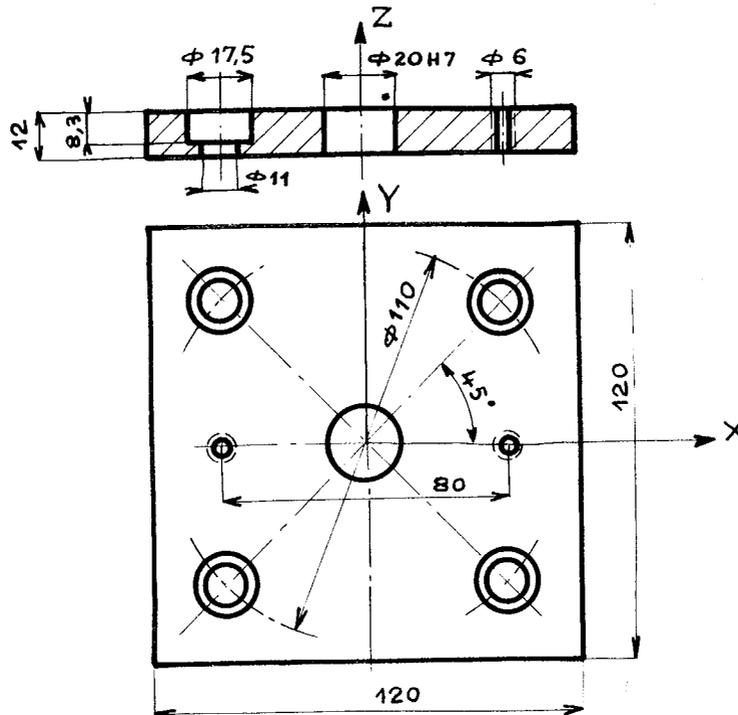


FIGURE 12 .

Nous donnons ci-dessous un programme en EXAPT pour l'usinage de la pièce représentée à la figure 12 .

```

PARTNO/PLAQUE 73 007 013 19 (1)
MACHINE/BOW E3 (2)
TRANS/100,200,0 (3)
PART/MATERL,2 (4)
Z SURF/12 (5)
PTCENT = POINT/0,0,0 (6)
CERCL1 = CIRCLE/CENTER, PTCENT, RADIUS, (110/2) (7)
RESEAU = PATTERN/ARC, CERCL1, 45, CCLW, 4 (8)
CL DIST/0,8 (9)
TARAU1 = TAP/DIAMET, 6, DEPTH, 12 (10)
PERC 1 = DRILL/SO, DIAMET, 11, DEPTH, 12 (11)
LAMAG 1= SINK/SO, DIAMET, 17.5, DEPTH, 8.3, TOOL, 314 (12)
COOLNT/ON (13)
FROM/100,0,100 ## DEPART (14)
WORK/PH,TARAU1 ## APPEL D'USINAGE (15)
GOTO/40,0,0 ## APPEL POSITIONNEMENT (16)
GOTO/-40,0,0 (17)
WORK/(REAM, DIAMET,20, DEPTH, 12) (18)
GOTO/PTCENT (19)
WORK/PERC 1, LAMAG 1 (20)
GOTO/RESEAU (21)
FINI ## FIN PROGRAMME ET ARRET MACHINE (22)

```

- (3) Indique une translation de la pièce lors de sa fixation sur la table.
  - (4) Matériau 2.
  - (8) On appelle RESEAU un réseau circulaire (ARC) de points disposés sur le cercle CERCL1, dont le premier point est à 45° par rapport à l'axe des X, où les positions suivantes sont considérées dans le sens inverse des aiguilles d'une montre (CCLW) et où le nombre total de points est 4.
  - (9) Distance de sécurité (clearance distance) 0,8 mm au dessus de la pièce.
  - (10) Cette instruction fait appel à un sous-programme (surfaçage, perçage, chanfreinage, taraudage)
  - (11) S O signifie single opération - Perçage seul sans surfaçage ni avant-trou.
  - (12) Le programmeur peut imposer l'outil (314)
  - (15) WORK : on appelle une opération d'usinage (ou plusieurs) à faire à chaque trou (P H : per hole)
  - (16)(17) GOTO : positionnement et exécution de l'usinage appelé.
- de même pour (18) et (19) puis (20) et (21).

## 5. EVOLUTION ET RECHERCHE [27,29,33,50,52]

L'évolution de APT doit être assurée dans un programme à long terme. On cherche à satisfaire les besoins qu'ont les utilisateurs (§3.1) pour des applications ou matériels nouveaux ou pour l'aide des ordinateurs à l'étude technique. On recherche par exemple une plus grande efficacité pour la définition des surfaces non mathématiques telles que les carrosseries prototypes d'automobiles.

A Glasgow, le National Engineering Laboratory (NEL) étudie le langage 2 CL conçu pour le contournage 2 axes et demi et

qui comporte le calcul automatique des passes d'évidement.

Pour l'introduction des données technologiques, c'est EXAPT qui est le plus avancé. Après EXAPT I pour le point à point, EXAPT II pour le tournage vient d'être élaboré (on définit les dimensions de la pièce brute et de la pièce finie et le programme détermine les passes à réaliser etc.) - EXAPT III pour le fraisage dans le plan est à l'étude dans les Universités Allemandes. Il pourrait s'inspirer de 2 CL.

En France :

Le langage IFAPT est réalisé par la Compagnie Internationale pour l'Informatique en collaboration avec le laboratoire Central de l'Armement pour n'utiliser que des calculateurs de faible capacité, ou de capacité en rapport avec les pièces à programmer. Son originalité réside dans sa conception modulaire. Il sera fait par étapes : IFAPT.P (point à point), IFAPT.C (contournage), IFAPT.PC (les deux), IFAPT.PCT (avec introduction de données technologiques plus ou moins évoluées par des modules additionnels).

Il utilise la grammaire APT avec certaines restrictions : par exemple il possède 5 définitions de point au lieu de 10.

Son développement est confié à l'ADEPA (Association pour le Développement du système de Programmation Automatique IFAPT). L'ADEPA offre ses services aux constructeurs pour l'écriture de post processeurs IFAPT, P et C.

Réalisé par BULL G.E. le langage MECA (basé sur des appels de procédures FORTRAN) est utilisable par télé-traitement.

La Régie Renault vient de mettre au point en liaison avec la C2I un système à dessiner et à sculpter des formes non mathématiques par le procédé UNISURF.

Plus que de faire de la recherche, la fonction de l'atelier pilote du L.C.A. est d'apprécier les qualités, les possibilités et la rentabilité des nouveautés en matériel et en programmation, et de former des utilisateurs.

Les langages de programmation dans une première phase permettent des définitions géométriques comme nous l'avons vu. Une deuxième phase de leur évolution concerne l'introduction de données technologiques (2 CL et EXAPT) ; c'est alors le programme qui détermine certaines phases d'usinage et le choix des outils. Dans une troisième phase la gestion d'un atelier peut être confiée à l'ordinateur qui choisit les machines-outils et le parcours des pièces en fonction des opérations d'usinage à réaliser. Les firmes Boeing aux Etats-Unis et Molins en Grande-Bretagne ont réalisé de tels systèmes.

Nous sortons un peu du sujet mais il nous faut évoquer : l'aide à la conception et à la programmation (mise au point rapide d'un programme) qu'apporte l'utilisation des écrans cathodiques ; l'utilisation d'un ordinateur qui traite les mesures données par des capteurs sur un prototype en fonctionnement, modifie les dimensions de la pièce en fonction des résultats obtenus et du cahier des charges, puis sort la bande perforée pour réaliser la pièce améliorée ; la commande adaptative, limitée pour l'instant par des problèmes de capteurs, qui modifie à tout instant les conditions d'usinage en fonction de certains critères.

## 6. NORMALISATION [ 5,14,3,63 ]

Après la normalisation de bandes perforées (dimensions, caractères, formats (§2)) et de la nomenclature des axes et mouvements des machines, celle des langages est en cours.

Nous l'avons déjà mentionné c'est APT qui est la base des discussions pour une normalisation. Ce qui ne signifie pas que le langage sera figé car les normes ne sont jamais définitives.

Cette normalisation qui devrait intervenir condamne plus ou moins les langages qui ne sont pas compatibles avec la référence. Quant aux langages dérivés, compatibles avec APT et compatibles entre eux, leurs auteurs espèrent bien qu'ils mériteront d'être inclus dans les normes, mais là il est trop tôt pour le savoir car ces nouveaux langages n'ont pas encore été suffisamment utilisés pour être bien jugés.

IFAPT n'a encore réalisé qu'une partie de son programme assez ambitieux.

L'entrée des données technologiques que propose EXAPT servira-t-elle de base pour une normalisation dans ce domaine ? Son utilisation dans l'industrie est toute récente. Il est encore peu implanté et c'est l'usage qui déterminera sa valeur.

Actuellement, ces deux derniers langages donnent une impression assez favorable.

## 7. CONCLUSION.

Grâce au développement des calculateurs numériques, la commande numérique des machines-outils ouvre une ère nouvelle en matière d'usinage.

Mais pour parler aux ordinateurs il faut un langage symbolique adapté.

Parmi les nombreux élaborés APT III est le plus riche, le plus complet, pour les définitions géométriques et de trajectoires à trois dimensions. Il est la base d'une étude de normalisation.

De nouveaux langages dérivés de APT sont nés dont le but est soit d'utiliser des calculateurs de petite capacité, soit de faire un nouveau pas vers l'automatisation par le calcul automatique des données d'usinage (ADAPT, IFAPT, 2 CL, EXAPT).

B I B L I O G R A P H I E

A- COMMANDE NUMERIQUE DES MACHINES-OUTILS

- [1] ANDERSON E. Introducing numerical control in a large industrial firm  
CIRP Conférence Européenne sur la commande numérique des machines-outils, PARIS Avril 1968, additif
- [2] ANKENNEY H.E, BINGHAM D.H. Jr. Production proved numerical control - three steps from point to point.  
Amer. Mach. 101 (1957) n°23 pp 145-156.
- [3] BEZIER P. Nomenclature des axes et mouvements - Blocs d'instruction et symbolisation des formats.  
Cour. de la Norm. n° 188 III-IV 1966 pp 191-193.
- [4] CHALANCON J. Commande numérique des machines-outils  
Projets de normalisation ISO en cours d'étude.  
Automatisme Tome X, Juillet Août 1965, pp 296-299.
- [5] CHALANCON J. Commande numérique des machines-outils et processus industriels  
Cour. de la Norm. n°188 III-IV 1966 pp 185-186.
- [6] CHAUVEAU J. Description d'un système de commande numérique point à point et paraxial  
Mécanique Electricité - Juin Juillet 1968 n°222-223  
pp 23-26
- [7] DUREAU G. Principes généraux et fonctionnement des systèmes de contournage Alcatel  
Mécanique Electricité, Août Sept. 1968, n°224-225  
pp 33-38
- [8] FARMER P.J. Automatic Machining  
Aircraft production Janv. 1958 p. 28
- [9] FULDNER H. Marriage Triangle : Machine Tools, Numerical control, and computers.  
Report from Cincinnati, Vol 23 (1966) N° 1 pp 12-13
- [10] HOLMES L.L. Adaptive control for Metalworking Applications.  
Automation, August 1967, pp 77-82

- [11] LEBRUSQUE R. Mesures pour l'Automatisation des machines-outils  
Ingénieurs et techniciens, n°225 Nov.1968, pp 71-78.
- [12] LOMBARD J. Aperçu sur la commande numérique des machines-outils  
CIRP Conférence Européenne sur la commande numérique des machines-outils, PARIS Avril 1968, pp 3-10.
- [13] LOMBARD J. Introduction à la commande numérique des machines outils  
Mécanique Electricité, Juin Juillet 1968, n° 222-223 pp 17-22.
- [14] MARCHESSEAU C. Code pour commande numérique des machines-outils.  
Cour. de la Norm. n°188 III-IV 1966 pp 187-190.
- [15] MERCHANT M.E. The manufacturing - System concept in production Engineering Research  
CIRP Annalem Vol 10 (1961-62) pp 77-83.
- [16] MERCHANT E. Progrès et problèmes dans l'application des nouvelles techniques d'optimalisation des fabrications.  
Mécanique Electricité, Août Sept. 1968, n° 224-225 pp 46-54.
- [17] MICHELETTI G.F. Revue des connaissances concernant les servomécanismes et organes de commande des machines à commande numérique  
CIRP Conférence Européenne sur la commande numérique des machines-outils, PARIS Avril 1968, pp 11-25.
- [18] MOTTU A. Les exigences de la précision pour l'étude, la fabrication et l'emploi des machines-outils à commande numérique  
CIRP Conférence Européenne sur la commande numérique des machines-outils, PARIS Avril 1968, pp 33-39
- [19] NYLANDER J.A. N/C centers serve small shops  
Amer. Mach. 107 (1962) n° 3 pp 55-56
- [20] PASTUREL G. Pourquoi, en France, la commande numérique est-elle si peu utilisée dans les machines-outils  
Electronique Actualités, n° 58, 22 Mars 1968 p. 1 et 11
- [21] PEASE V.N. Numerical control  
Aircraft production 14 (1952) n°10 pp 356-358.

- [22] ROSS D.T. Implication of computer-Aided Design for numerically-Controlled Production.  
Proc. of the 1st Annual meeting and Technical conference of the N.C. Society (1964) pp 100-111.
- [23] SACHOT M. Evolution des centres d'usinage  
CIRP Conférence Européenne sur la commande numérique des machines outils, PARIS Avril 1968, pp 27-32.
- [24] SIMON W. Commande numérique des machines-outils (traduit par R. ARONSSOHN)  
Eyrolles - PARIS 1967.
- [25] SORNICLE D. Utilisation des différents modes d'interpolation dans la commande numérique C.O.T.E.C.  
Mécanique-Electricité, Août-Sept. 1968, n° 224-225 pp 29-32.
- [26] STOCKER W.M. The production man's guide to numerical control. Amer. Mach. 101 (1957) n° 14 pp 133-154.
- [27] THILLIEZ J. La commande numérique des machines-outils. Dunod PARIS 1967.
- [28] VERMOT-GAUD J. Commande numérique des machines-outils. 2ème thèse - Faculté des Sciences de GRENOBLE, 1959.
- [29] WILLIAMSON D.T.N. System 24. A new concept of manufacture. CIRP Conférence Européenne sur la commande numérique des machines-outils, PARIS Avril 1968, pp 41-60.
- [30] Bibliographie réunie par P. BEZIER, dans [27] - plus de 800 références.
- [31] Commande numérique directe à distance de plusieurs machines-outils  
Automation n° 107, Janvier 1969, p. 19
- [32] SIA PARIS - Cycle de conférences sur la commande numérique 1965 (avec bibliographie).

B- PROGRAMMATION EN COMMANDE NUMERIQUE.

- [33] BEZIER P. Procédé de définition numérique des courbes et surfaces non mathématiques - Système UNISURF.  
Mécanique-Electricité, Août Sept. 1968, n° 224-225 pp 39-45.  
Automatisme, Tome XIII, n° 5 Mai 1968, pp 189-196.

- [34] FARMER P.J. Programming  
Aircraft Production Sept. 1957 p. 344
- [35] GAMBIN P. Aspects économiques de la programmation  
Mécanique-Electricité, Juin Juillet 1968, n° 222-223,  
pp 91-94.
- [36] GIGNAC J. Exemple de programmation en continu  
Mécanique-Electricité, Juin Juillet 1968, n° 222-223,  
pp 62-78.
- [37] LETAC G. Les différents langages de programmation  
Mécanique-Electricité, Juin Juillet 1968, n° 222-223,  
pp 51-52.
- [38] LETAC G. Définition des courbes et surfaces en langage APT  
et ADAPT.  
Mécanique-Electricité, Août Sept. 1968, n° 224-225, pp 17-21
- [39] MITTMAN B. La commande symbolique. Le système APT de  
programmation des machines-outils à commande  
numérique.  
Automatisme Tome X, n° 2 févr. 1965, pp 76-81.
- [40] MORENS J.C. Aide graphique à la programmation  
Mécanique-Electricité, Juin Juillet 1968, n° 222-223,  
pp 79-82.
- [41] OPITZ H., HIRSCH B. Programmation automatique des machines  
outils à commande numérique  
CIRP Conférence Européenne sur la commande numérique des  
machines-outils, PARIS Avril 1968, pp 61 à 70.
- [42] PONTE J. Le système "Profile-Data"  
Mécanique-Electricité, Août Sept. 1968, n° 224-225,  
pp, 13-16.
- [43] SACHOT M. Exemples pratiques de programmation en  
commande point à point  
Mécanique-Electricité, Juin Juillet 1968, n° 222-223,  
pp 53-61.
- [44] SAUVAIRE C. Intégration des données technologiques dans  
les programmes automatiques  
Mécanique-Electricité, Juin Juillet 1968, n° 222-223,  
pp 83-90.
- [45] SIEGEL A. Automatic Programming of Numerically controlled  
Machine-Tools.  
Control Engineering, Oct 1956, p. 65
- [46] SOUBIES-CAMY H. La commande numérique des machines-outils  
Electronique Indus. Janv. Fév. 1968 pp 3-8, Mars 68  
pp 137-139, Avril 1968 pp. 217-222, Mai 68 pp.365-367.

- [47] SOUBIES-CAMY H. Les langages de programmation en  
commande numérique  
Automatisme. Tome XIII, n° 4 Avril 1968, pp 173-179,  
n° 5 Mai 1968, pp 220-226.
- [48] SOUBIES-CAMY H. Les bases de la programmation en  
commande numérique  
Mécanique-Electricité, Juin Juillet 1968, n° 222-223,  
pp 32-50.
- [49] TAUPIN R. Programmes automatiques spécialisés  
Mécanique-Electricité, Août Sept. 1968, n° 224-225  
pp 23-26.
- [50] WASIUKIEWICZ I. Programmation autom. de machines-outils  
à commande numérique.  
Ingénieurs et techniciens, n° 225 Nov. 1968  
pp 83-88.
- [51] WEILL R. Aperçus sur les méthodes nouvelles de program-  
mation en commande numérique.  
Numéro spécial du GAMI 1967.
- [52] WEILL R. Rôles et activités d'un atelier-pilote et  
d'un centre de programmation de commande  
numérique.  
CIRP Conférence Européenne sur la commande numérique  
des machines-outils, PARIS Avril 1968, pp 71 à 79.
- [53] APT Part Programming  
by the APT Long Range Program Staff, IIT Research  
Institute  
Mac Graw-Hill - New York 1967.
- [54] APT Encyclopedia  
UNIVAC Division of Sperry Rand Corporation.
- [55] N/C 360 APT - Application Description  
IBM Application Program, Form H 20-0181-1
- [56] Procédé UNISURF  
doc. CII Réf. NT 2115, Août 1967
- [57] Système de programmation IFAPT P point à point.  
doc. CII Réf. RT 2217/Fr Janvier 1968.
- [58] Système de programmation IFAPT  
doc. CII Réf. NT 2210 Août 1967.

- [59] EXAPT 1 Sprachbeschreibung  
Universités d'Aix-la-Chapelle et de Berlin (W.SIMON, H.OPTZ)  
Octobre 1966  
Description du langage EXAPT 1 pour commande point à point et linéaire simple. Traduction Française (assurée par la Société Alsthom).
- [60] N/C AUTOSPOT - Application Description  
Programme d'application IBM n° H 20-0179-1.
- [61] MECA - Notice Bull G.E. 0030 204 F
- [62] Le langage MECA - Les séminaires de programmation de l'IMAG 1967-68.
- [63] Rapports du sous-comité ISO/TC 97/SC5. Programming languages for the numerical control of machines.
- [64] Langages en commande numérique (IFAPT)  
Document Télémechanique Electrique 86655 TRE/AH 24/10/67
- [65] Intérêt des langages de programmation en commande numérique  
Document Télémechanique Electrique 86658 TRE/AH 14/11/67
- [66] Compte-rendu cours post processor EXAPT  
Document Télémechanique Electrique 86675 TRE/AH/NW  
23/1/68
- [67] Utilisation des langages en commande numérique  
Document Télémechanique Electrique 86716 TRE/A Hamet  
14/6/68
- [68] Les langages de commande numérique  
Document Télémechanique Electrique 86755 TRE/A Hamet  
10/1/69