

**MANUEL  
DE REFERENCE  
Outil de mise  
au point : DDT**

TABLE DES MATIERES

1. INTRODUCTION

2. LES COMMANDES DDT

- 2.1 La commande A (d'assemblage)
- 2.2 La commande D (d'affichage)
- 2.3 La commande F (de remplissage)
- 2.4 La commande G (de départ)
- 2.5 La commande I (d'entrée)
- 2.6 La commande L (liste)
- 2.7 La commande M (de déplacement)
- 2.8 La commande R (de lecture)
- 2.9 La commande S (de mise en place)
- 2.10 La commande T
- 2.11 La commande U
- 2.12 La commande X (d'examen)

3. REMARQUES D'IMPLANTATION

### AVERTISSEMENT

Le CPU du SIL'Z étant un Z80, cet utilitaire DDT n'est pas idéal.

Nous vous conseillons de vous procurer le logiciel ZSID de Digital Research, qui est spécialement prévu pour le CPU Z80.

## 1. INTRODUCTION

Le programme DDT permet de tester et de détecter les erreurs de manière dynamique et interactive dans les programmes créés sous CP/M. Ce programme est lancé en tapant les commandes suivantes au niveau de commande console CP/M :

```
DDT
DDT  nomfichier.HEX
DDT  nomfichier.COM
```

où "nomfichier" est le nom du programme qui doit être chargé et testé. Dans tous les cas, le programme DDT est amené en mémoire principale à la place du Processeur de Commande Console (CCP) et réside ainsi directement sous la portion BDOS du CP/M. L'adresse de début du BDOS, qui se trouve dans le champ adresse de l'instruction JMP située à l'emplacement 5H, est modifiée pour refléter la réduction de la zone de programme transitoire (TPA). Les seconde et troisième formes de la commande DDT décrites ci-dessus effectuent les mêmes actions que la première, sauf qu'il y a un changement automatique des fichiers HEX ou COM. L'action est identique à la suite des commandes suivantes :

```
DDT
Inom fichier.HEX ou Inomfichier.COM
R
```

où les commandes I et R mettent en place et lisent le fichier à tester (voir l'explication des commandes I et R ci-dessous). Après le lancement, le DDT imprime un message de début de la forme

```
nnK DDT-s VER m.m
```

où nn est la taille de la mémoire (qui doit correspondre au système CP/M utilisé), s est le système hardware, correspondant aux codes :

```
D      version standard Digital Research
M      version MDS
I      version standard IMSAI
O      systèmes omron
S      version standard Digital Systems
```

et m.m le numéro de la version.

Après le message de début, DDT donne son signal à l'opérateur "-", et attend l'entrée de commandes sur la console. L'opérateur peut taper l'un des caractères de commande, suivi d'un retour chariot pour exécuter la commande. Chaque ligne d'entrée peut être éditée en utilisant les contrôles CP/M standards.

RUBOUT	enlève le dernier caractère tapé
CTRL-U	annule toute la ligne, et permet d'en retaper une autre
CTRL-C	réinitialisation du système

Les commandes peuvent faire jusqu'à 32 caractères de long (un retour chariot automatique est inséré au 33 ième caractère), le premier caractère déterminant le type de la commande.

- A entre en assembleur les mnémoniques et les opérandes
- B affiche la mémoire en hexadécimal et ASCII
- F remplit la mémoire avec une donnée constante
- G commence l'exécution avec des points d'arrêts optionnels
- I met en place un bloc de contrôle de fichier d'entrée standard
- L liste la mémoire en utilisant les mnémoniques d'assembleur
- M déplace un segment de mémoire de la source à la destination
- R lit un programme pour un test ultérieur
- S substitue des valeurs en mémoire
- T donne la trace de l'exécution du programme
- U exécution du programme sans trace
- X examine et éventuellement modifie l'état des registres du CPU.

Le caractère de commande, dans certains cas, est suivi de zéro, une, deux ou trois valeurs hexadécimales qui sont séparées par des virgules ou un simple blanc. Toutes les valeurs données en sortie par le DDT le sont en hexadécimal. Dans tous les cas, les commandes ne sont pas exécutées tant qu'un retour chariot n'a pas été tapé à la fin de la commande.

A n'importe quel moment de l'exécution du programme de détection d'erreur, l'opérateur peut arrêter l'exécution du DDT en utilisant soit un BREAK ou GO (saut à l'emplacement 0000H), et sauvegarder l'image mémoire en utilisant une commande SAVE de la forme :

SAVE n nomfichier.COM

où n est le nombre de pages (blocs de 256 octets) qui doivent être sauvegardées sur disque. Le nombre de blocs peut être déterminé en prenant l'octet le plus significatif de l'adresse la plus élevée de la zone chargée, et en convertissant ce nombre en décimal. Par exemple, si l'adresse la plus grande de la zone de programme transitoire (TPA) est 1234H, le nombre de pages est 12H, soit 18 en décimal. L'opérateur pourrait donc taper un CTRL-C pendant l'exécution du DDT, retournant ainsi au niveau du processeur de console, puis :

SAVE 18 X.COM

L'image mémoire est sauvegardée dans le fichier X.COM sur la disquette, et peut être directement exécutée en tapant simplement le nom X. Si des tests supplémentaires sont nécessaires, l'image mémoire peut être rappelée en tapant :

DDT X.COM

ce qui recharge le programme précédemment sauvegardé à partir de l'emplacement 100H jusqu'à la page 18 (12FFH). L'état de la machine ne fait pas partie du fichier COM, et le programme doit donc être redémarré à partir du début pour le tester correctement.

## 2. LES COMMANDES DDT

Les commandes sont données une par une ci-dessous en détail. Dans tous les cas l'opérateur doit attendre le signal "-" avant d'entrer la commande. Si le contrôle est donné à un programme sous test, et si le programme n'a pas atteint un point d'arrêt, le contrôle peut être renvoyé au DDT en tapant un RST 7 à partir du panneau avant (remarquez qu'il faut taper sur la touche rubout au lieu de faire un RST 7 si le programme est en train d'exécuter une commande T ou U). Dans les explications de chaque commande, la lettre de commande est décrite dans certains cas avec des nombres séparés par des virgules, ces nombres étant représentés par des minuscules. Ces nombres sont toujours supposés être en hexadécimal, et de un à quatre chiffres (les nombres plus grands seront automatiquement tronqués à droite).

De nombreuses commandes travaillent sur un "état du CPU" qui correspond au programme testé. L'état du CPU contient les registres du programme testé, et contient initialement des zéros pour tous les registres et les drapeaux, sauf pour le compteur du programme (P) et le pointeur de pile (S), dont la valeur par défaut est 100H. Le compteur du programme prend ensuite la valeur de l'adresse de début donnée dans le dernier enregistrement d'un fichier HEX si un fichier de cette forme est chargé (voir les commandes I et R).

### 2.1 LA COMMANDE A (D'ASSEMBLAGE)

Le DDT permet d'entrer manuellement des lignes d'assemblage dans la mémoire par la commande A qui a la forme suivante :

As

où s est l'adresse hexadécimale de début de l'assemblage. Le DDT donne sur la console l'adresse de la nouvelle instruction à mettre, et lit la console, en cherchant des mnémoniques de langage d'assemblage (voir l'Intel 8080 Assembly Language Référence Card pour avoir la liste des mnémoriques), suivis des références de registre et des opérandes sous forme hexadécimale absolue. Chaque adresse de chargement est imprimée avant de lire la console. La commande A se termine lorsque la première ligne vide est donnée à la console, lorsque l'on frappe un "." ou en cas d'erreur.

A la fin de l'entrée du langage d'assemblage, l'opérateur peut revoir le segment de mémoire en utilisant le désassembleur du DDT (voir la commande L). Remarquez que la portion du DDT de l'assembleur ou du désassembleur peut être recouverte par le programme transitoire testé, auquel cas le programme répond avec une condition d'erreur lorsque les commandes A ou L sont utilisées (reportez-vous à la section 4).

## 2.2 LA COMMANDE D (D'AFFICHAGE)

La commande D permet à l'opérateur de voir le contenu de la mémoire dans les formats hexadécimaux et ASCII. Les formes sont les suivantes

D  
Ds  
Ds, f

Dans le premier cas, la mémoire est affichée à partir de l'adresse d'affichage courante (initialement 10H), et continue avec 16 lignes d'affichage. Chaque ligne affichée à la forme suivante :

aaaa bb cccccccccccccccc

où aaaa est l'adresse d'affichage en hexadécimal, et bb représente les données présentes en mémoire à partir de aaaa. Les caractères ASCII qui commencent à aaaa sont donnés à droite (ils sont représentés par une suite de c), et les caractères non graphiques sont représentés par des points. Remarquez que les majuscules et les minuscules sont affichées, et qu'elles apparaîtront toutes comme des majuscules sur une console qui n'a que des majuscules. Chaque ligne affichée correspond aux valeurs de 16 octets de données, sauf dans le cas de la première ligne qui est tronquée de façon à ce que les lignes suivantes commencent à une adresse qui soit un multiple de 16.

La seconde forme de la commande D montrée ci-dessus est analogue à la première, sauf que l'adresse d'affichage prend au début la valeur s. La troisième forme fait continuer l'affichage de l'adresse s à l'adresse f. Dans tous les cas, l'adresse d'affichage prend la valeur de l'adresse qui suit la dernière affichée dans cette commande, et on peut ainsi obtenir un affichage en continu en tapant des commandes D les unes après les autres sans donner d'adresse explicite.

Des affichages excessivement longs peuvent être stoppés en tapant sur la touche rubout.

## 2.3 LA COMMANDE F (DE REMPLISSAGE)

La commande F a la forme suivante :

Fs, f, c

où s est l'adresse de début, f l'adresse de fin, et c une constante hexadécimale sur un octet. L'effet est le suivant : le DDT stocke la constante c à l'adresse s, incrémente la valeur de s, et teste à nouveau f. Si s est supérieur à f l'opération s'arrête, sinon elle se répète. La commande de remplissage peut donc être utilisée pour donner à un bloc de mémoire la valeur d'une constante donnée.

## 2.4 LA COMMANDE G (DE DEPART)

L'exécution du programme commence lorsqu'on utilise la commande G, avec éventuellement une ou deux adresses de points d'arrêt. La comande G a une des formes suivantes :

G  
Gs  
Gs,b  
Gs,b,c  
G,b  
G,b,c

La première forme lance l'exécution du programme testé à la valeur courante du compteur de programme dans l'état actuel de la machine, sans points d'arrêt (la seule manière pour reprendre le contrôle dans le DDT est d'exécuter un RST 7). La seconde forme est analogue à la première, sauf que le compteur de programme dans l'état actuel de la machine prend la valeur de l'adresse s avant le début de l'exécution. La troisième forme est la même que la seconde, sauf que l'exécution du programme s'arrête quand l'adresse b est rencontrée (b doit être dans la zone du programme testé). L'instruction située à l'emplacement b n'est pas exécutée. La quatrième forme est identique à la troisième, sauf qu'ici deux points d'arrêt sont spécifiés, l'un à b et l'autre à c. La rencontre de l'un ou l'autre de ces points d'arrêt entraîne l'arrêt de l'exécution, et les deux points d'arrêt sont ensuite enlevés. Les deux dernières formes prennent le compteur de l'état courant de la machine, et mettent respectivement un ou deux points d'arrêt.

L'exécution se fait à partir de l'adresse de début en temps réel jusqu'au premier point d'arrêt rencontré. Autrement dit, il n'y a aucune intervention du DDT entre l'adresse de départ et l'adresse d'arrêt. Ainsi, si le programme testé n'atteint pas de point d'arrêt, le contrôle ne peut être rendu au DDT qu'en exécutant une instruction RST 7. Lorsqu'il rencontre un point d'arrêt, le DDT arrête l'exécution et tape :

\*d

d étant l'adresse d'arrêt. L'état de la machine peut alors être examiné en utilisant la commande X. L'opérateur doit donner des points d'arrêt qui diffèrent de l'adresse du compteur du programme du début de la commande G. Ainsi, si le compteur du programme courant est 1234H, les commandes :

G,1234

et

G400,400

produisent toutes deux un point d'arrêt immédiat, sans exécution d'aucune instruction.

## 2.5 LA COMMANDE I (D'ENTREE)

La commande I permet à l'opérateur d'insérer un nom de fichier dans le bloc de contrôle de fichier par défaut situé à 5CH (le bloc de contrôle de fichier créé par le CP/M pour les programmes transitoires est placé à 5CH. Le bloc de contrôle de fichier par défaut peut être utilisé par le programme testé comme s'il avait été passé par le processeur de console du CP/M. Remarquez que ce nom de fichier est aussi utilisé par le DDT pour lire les fichiers HEX et COM supplémentaires. La forme de la commande I est la suivante :

Inomfichier

ou

Inomfichier.typefichier

Si la seconde forme est utilisée, et si le type (type fichier) est HEX ou COM, des commandes R peuvent être ultérieurement utilisées pour lire le code machine en format hex ou binaire (voir la commande R pour plus de détails).

## 2.6 LA COMMANDE L (LISTE)

La commande L est utilisée pour donner la liste des mnémoniques d'assembleur de la mémoire. Les formes sont les suivantes :

L

Ls

Ls,f

La première commande liste douze lignes de code machine désassemblé à partir de l'adresse courante. La seconde forme donne à l'adresse de liste la valeur s, puis liste douze lignes de code. La dernière forme liste le code désassemblé de l'adresse s à l'adresse f. Dans tous les cas, l'adresse de liste prend la valeur qui suit la dernière adresse listée, en préparation des commandes L suivantes. Lorsqu'un point d'arrêt est rencontré, l'adresse de liste prend la valeur courante du compteur du programme. (voir les commandes G et T) Là aussi, les affichages trop longs peuvent être stoppés en utilisant la touche rubout pendant le processus.

## 2.7 LA COMMANDE M (DE DEPLACEMENT)

La commande M permet de déplacer en bloc des zones de programme ou de données d'un emplacement à un autre en mémoire. La forme est la suivante :

Ms,f,d

où s est l'adresse de début et f celle de fin du bloc à déplacer, et d l'adresse de destination. Les informations sont d'abord déplacées de s à d, et les deux adresses sont incrémentées. Si s est plus grand que f l'opération s'arrête, sinon elle est répétée.

## 2.8 LA COMMANDE R (DE LECTURE)

La commande R utilisée avec la commande I pour titre des fichiers COM et HEX de la disquette dans la zone de programme transitoire, en préparation à la détection d'erreurs. Les formes sont les suivantes :

```
R
Rb
```

où B est une adresse de décalage optionnelle qui est ajoutée à chaque adresse de programme ou donnée lorsqu'elle est chargée. L'opération de chargement ne doit pas écrire par dessus les paramètres du système situés entre 000H et 0FFH (la première page de mémoire). Si b est omis, la valeur b=0000 est prise. La commande R nécessite une commande I antérieure, spécifiant le nom des fichiers HEX et COM. L'adresse de chargement de chaque enregistrement est obtenue à partir de chaque enregistrement HEX, alors qu'une adresse de chargement pour les fichiers COM est supposée égale à 100H. Remarquez que l'on peut faire n'importe quel nombre de commandes R après la commande I pour relire le programme testé, en supposant que le programme testé ne détruit pas la zone de défaut située à 5CH. N'importe quel fichier de type "COM" est supposé contenir un code machine sous forme binaire pure (créé avec une commande LOAD or SAVE), et tous les autres sont supposés contenir un code machine en format hex d'Intel (produit, par exemple, par une commande ASM).

Souvenez-vous que la commande :

```
DDT nomfichier.typefichier
```

qui lance le programme DDT est équivalente à

```
DDT
-Inomfichier.typefichier
-R
```

Chaque fois qu'une commande R est effectuée, le DDT répond soit avec l'indicateur d'erreur "?" (le fichier ne peut être ouvert, ou une erreur de somme de contrôle est apparue dans un fichier HEX), soit avec un message de chargement de la forme suivante :

```
NEXT PC
nnnn pppp
```

où nnnn est l'adresse qui suit le programme chargé, et pppp le compteur de programme supposé (100H pour les fichiers COM, ou pris à partir du dernier enregistrement si c'est un fichier HEX qui est spécifié).

## 2.9 LA COMMANDE S (DE MISE EN PLACE)

La commande S permet d'examiner et de modifier éventuellement des emplacements de mémoire. La forme en est la suivante :

Ss

où s est l'adresse de départ, en hexadécimal, de l'examen et de la modification de la mémoire. Le DDT répond avec un signal numérique qui donne l'emplacement mémoire, et le contenu mémoire de cet emplacement. Si l'opérateur tape un retour de chariot, l'information n'est pas modifiée. Si une valeur d'octet est frappée cette valeur est stockée à l'adresse donnée par le DDT. Dans tous les cas, le DDT continue à donner des adresses successives et les valeurs correspondantes jusqu'à ce qu'un point soit frappé par l'opérateur, ou jusqu'à ce qu'une valeur d'entrée non valide soit détectée.

## 2.10 LA COMMANDE T

La commande T permet d'avoir une trace de l'exécution du programme de 1 à 65535 pas de programme. Les formes en sont :

T

Tn

Dans le premier cas, l'état du CPU est affiché, et le pas de programme suivant est exécuté. Le programme se termine tout de suite, l'adresse de fin étant affichée de la manière suivante :

\*hhhh

où hhhh est la prochaine adresse à exécuter. L'adresse d'affichage (utilisée dans la commande D) prend la valeur de H et L, l'adresse de liste (utilisée dans la commande D) prend la valeur hhhh. L'état du CPU à la fin du programme peut ensuite être examiné en utilisant la commande X.

La seconde forme de la commande est analogue à la première, sauf que la trace de l'exécution est donnée pour n pas (n est une valeur hexadécimale) avant qu'un point d'arrêt n'apparaisse. Un point d'arrêt peut être forcé en mode "trace" en utilisant un "rubout". L'état du CPU est affiché avant que chaque pas de programme ne soit pris en mode "trace". Le format de l'affichage est le même que celui décrit dans la commande X.

Remarquez que la trace du programme a une discontinuité entre l'interface et le CP/M, et qu'elle reprend après son retour du CP/M au programme testé. Ainsi, les fonctions du CP/M qui accèdent aux unités d'E/S, comme le drive de la disquette, tournent en temps réel, évitant les problèmes de temps des E/S. Les programmes qui tournent sous DDT en mode trace s'exécutent environ 500 fois plus lentement que leur temps réel, car le DDT prend le contrôle après que chaque instruction de l'utilisateur ait été exécutée. Les routines qui accomplissent des interruptions peuvent ainsi être tracées, mais il faut remarquer que les commandes qui utilisent les possibilités des points d'arrêt (G, T, et U) accomplissent l'arrêt en exécutant une instruction RST 7, ce qui signifie que le programme testé ne peut utiliser cet emplacement d'interruption. De plus, le mode "trace" exécute toujours le programme testé avec des possibilités d'interruption, ce qui peut poser des problèmes si des interruptions asynchrones sont reçues pendant le traçage.

Remarquez aussi que l'opérateur doit utiliser la touche "rubout" pour rendre le contrôle au DDT pendant le traçage, au lieu d'exécuter un RST 7, pour être sûr que la trace de l'instruction courante soit terminée avant l'interruption.

## 2.11 LA COMMANDE U (SANS TRACE)

La commande U est analogue à la commande T, sauf que les pas de programme intermédiaires ne sont pas affichés. Le mode "sans trace" permet d'exécuter de 1 à 65.535 pas de programmes en mode moniteur, et est utilisé principalement pour retenir sous contrôle un programme en exécution lorsqu'il atteint des conditions d'état régulières. Toutes les conditions de la commande T s'appliquent à la commande U.

## 2.12 LA COMMANDE X (D'EXAMEN)

La commande X permet d'afficher et de modifier l'état courant du CPU pour un programme testé. Ses formes sont les suivantes :

X  
Xr

où r est un des registres du CPU

C	drapeau de retenue	(0/1)
Z	drapeau de zéro	(0/1)
M	drapeau négatif	(0/1)
E	drapeau de parité paire	(0/1)
I	retenue inter-chiffre	(0/1)
A	accumulateur	(0-FF)
B	paire de registres BC	(0-FFFF)
D	paire de registres DE	(0-FFFF)
H	paire de registre HL	(0-FFFF)
S	pointeur de pile	(0-FFFF)
P	compteur du programme	(0-FFFF)

Dans le premier cas, l'état des registres du CPU est affiché sous la forme suivante :

```
CfZfMfEfIf A=bb B=dddd D=dddd H=dddd S=dddd P=dddd inst
```

où f est la valeur d'un drapeau (0 ou 1), bb la valeur d'un octet, et dddd la valeur d'un double octet correspondant à une paire de registres. Le champ "inst" contient l'instruction désassemblée qui apparaît à l'emplacement dont l'adresse est donnée par le compteur de programme d'état du CPU.

La seconde forme permet l'affichage et éventuellement la modification des valeurs d'un registre, r étant un des registres donnés ci-dessus (C,Z,M,E,I,A,B,D,H,S, ou P). Dans tous les cas, la valeur du drapeau ou du registre est d'abord affichée à la console. Le programme DDT accepte ensuite une entrée à la console. Si un retour de chariot est frappé, la valeur du drapeau ou du registre n'est pas modifiée. Si une valeur correcte est frappée, la valeur du drapeau ou du registre est modifiée. Remarquez que EC, DE et HL sont affichés comme des paires de registre. L'opérateur doit donc frapper la paire en entier lorsque B, C ou la paire EC sont modifiés.

### 3. REMARQUES D'IMPLANTATION

L'organisation du DDT permet à certaines portions non essentielles d'être recouvertes pour avoir une zone de programme transitoire plus grande pour mettre au point de grands programmes. Le programme DDT se compose de deux parties : le noyau du DDT et le module d'assemblage et de désassemblage. Le noyau du DDT est chargé par le CCP, et, bien qu'il soit chargé avec le noyau du DDT, le module d'assemblage et de désassemblage peut être recouvert s'il ne doit pas être utilisé.

En particulier, l'adresse du EDOS située à 6H (le champ d'adresse de l'instruction JMP située à 5H) est modifiée par le DDT pour devenir l'adresse de l'emplacement de base du noyau du DDT qui, à son tour, contient une instruction JMP de saut du EDOS. Ainsi, les programmes qui utilisent ces champs d'adresses pour donner une taille à la mémoire, voient la fin logique de la mémoire à la base du noyau du DDT plutôt qu'à la base du EDOS.

Le module d'assemblage/désassemblage réside juste sous le noyau du DDT dans la zone de programme transitoire. Si les commandes A, L, T, ou X sont utilisées pendant le processus de détection des erreurs, le programme DDT modifie là-aussi le champ d'adresse situé à 6H pour y inclure ce module d'assemblage/désassemblage, les commandes A et L sont perdues (elles donnent un "?" en réponse), et les commandes de trace et d'affichage (T et X) donnent le champ "inst" de l'affichage en hexadécimal, au lieu d'une instruction décodée.