

SOLAR

AID

Aide à la mise au point
en mono utilisation

FONDS DOCUMENTAIRE
SMP

AID

MANUEL DE RÉFÉRENCE

| SOMMAIRE | Pages |
|---|-------|
| 1 - PRESENTATION | 1 |
| 2 - DESCRIPTION | 1 |
| 2.1 - Principes | 1 |
| 2.2 - Fonctions réalisées | 2 |
| 2.3 - Mode maitre, mode esclave | 2 |
| 2.4 - Dialogue opérateur | 3 |
| 3 - VISUALISATION ET CHARGEMENT | 4 |
| 3.1 - Visualisation : clé / | 4 |
| 3.2 - Chargement : clé = | 4 |
| 3.3 - Incrémentation automatique | 4 |
| 4 - MODES D'EXECUTION | 4 |
| 4.1 - Pas à pas | 4 |
| 4.2 - Continu : CN | 5 |
| 5 - POINTS D'ARRET | 5 |
| 5.1 - Commandes individuelles | 5 |
| 5.2 - Commandes globales | 6 |
| 6 - CONSEILS PRATIQUES | 6 |
| 6.1 - Essais en mode esclave | 6 |
| 6.2 - Utilisation du téléimprimeur | 7 |
| 6.3 - interruptions et masque des interruptions | 7 |
| 6.4 - utilisation de l'option schéduleur microprogrammé | 8 |
| 7 - MISE EN OEUVRE | 8 |
| 7.1 - version autonome | 8 |
| 7.2 - version sous système | 9 |
| NOTATIONS | 10 |
| SYNOPTIQUE | 11 |
| MESSAGES D'ERREUR | 12 |

1 - PRESENTATION

AID est un outil d'aide à la mise au point.

Il est utilisable sur toute la gamme SOLAR depuis la configuration minimale (4K mots, téléimprimeur) avec ou sans les options "scheduler microprogrammé" et protection mémoire (DRPS).

Le but qui a été recherché dans ce produit est de reproduire en mode conversationnel au téléimprimeur les fonctions du pupitre.

Il existe donc les mêmes possibilités : visualisation, chargement de registres et de mémoires, exécution en continu ou en pas à pas, pose et effacement de points d'arrêt (arrêt instruction ou arrêt mémoire).

L'utilisation de AID présente par rapport au pupitre l'avantage de conserver une trace écrite des manipulations effectuées et des événements survenus. Le travail est facilité par l'utilisation de l'hexadécimal et les vérifications qui sont effectuées sur les commandes de l'opérateur.

Deux fonctions ont été ajoutées par rapport aux fonctions pupitre :

- Pas à pas multiple.
- Liste des points d'arrêt.

2 - DESCRIPTION

2.1 - Principes

AID s'approprie le niveau hardware 0 (Ho).

Il n'y a pas simulation mais exécution réelle du programme testé. Le programme qui se déroule ne peut passer la main à AID que par une alarme :

- ACTD (active DEBUG) alarme N° 9 prévue à cet effet.
- Passage sur un point d'arrêt instruction ou mémoire. (alarme N°7)
- Toute autre alarme non prévue et provoquée par le programme.

AID relance le programme testé soit en continu (jusqu'au prochain ACTD ou point d'arrêt) soit en pas à pas (Les points d'arrêt ne sont pas vus, l'acquittement de la tâche Ho se fait, dans ce cas, par l'instruction STEP).

2.2 - Fonctions réalisées

Toutes les fonctions demandées par l'opérateur se font en dialogue téléimprimeur sous Ho :

2.2.1. - Fonction visualisation registres : A,B,X,Y,C,L,W,K,P,ST,SLØ,SLE,IM,HV.

Fonction visualisation mémoire : Zone (0,64K) au maximum pour un programme maître, tranche (SLO-SLE) pour un programme esclave.

Fonction chargement registre : (les mêmes sauf HV).

Fonction chargement mémoire : Les mêmes zones qu'en visualisation sont accessibles. AID lui même est protégé.

Notons que le registre ST concerné par ces deux fonctions est le status machine concaténé avec le 4° mot des PST Hardware.

2.2.2. - Déroulement continu : Exécution réelle du programme testé.

Déroulement pas à pas : Le nombre de pas est compris entre 1 et 9.

En cas de pas à pas multiple les instructions sont déroulées successivement en mode STEP.

2.2.3. - Pose, effacement d'un point d'arrêt.

AID permet de poser et d'effacer individuellement les points d'arrêt. Leur nombre n'est pas limité. Ce sont indifféremment des points d'arrêt instruction ou mémoire.

Liste et effacement des points d'arrêt.

Il existe deux commandes globales l'une permettant de lister tous les points d'arrêt présents à cet instant et l'autre de les effacer complètement.

2.3 - Mode maître, Mode esclave

La majorité des fonctions font intervenir une adresse (visualisation, chargement de mémoire, pose, effacement, liste des points d'arrêt).

Cette adresse est absolue si le programme testé est en mode maître. Elle est relative si le programme testé est en mode esclave.

Sous AID on peut modifier SLØ, SLE comme les autres registres. La modification effective intervient lorsque AID exécute la prochaine fonction sur adresse demandée, sinon la modification se fait lors de la relance en pas à pas ou en continu. La modification de SLO, SLE est possible dans les deux cas, maître et esclave.

Pour toute fonction sur adresse AID vérifie que celle-ci est dans la plage permise.

$0 \leq Ad \leq \text{MIN} (64K, \text{MAXMEM})$ pour programme maître.

$SLO \leq Ad \leq SLE$ pour programme esclave.

On vérifie en outre que Ad est en dehors de la zone occupée par AID lui même.

(MAXMEM est une mémoire qui indique la taille de la mémoire).



fonctions LEC (liste des points d'arrêt) et REC (effacement des points d'arrêt) font intervenir le balayage d'une plage mémoire. C'est la plage permise du programme testé, indiquée plus haut.

Par modification de ST, SLO, SLE l'opérateur peut accéder à des zones mémoires normalement interdites, il ne devra pas oublier de restituer les valeurs normales avant de relancer le programme testé.

2.4 - Dialogue opérateur

Le dialogue téléimprimeur se compose :

- des commandes opérateur.
- des éditions de message.


2.4.1. - Les commandes opérateur sont :

- . / : visualisation
- . = : chargement
- . > ou n> : pas à pas (1 pas ou npas $1 \leq n \leq 9$)
 - CN: déroulement continu
- . DB: pose d'un point d'arrêt
- . RB: effacement d'un point d'arrêt
- LEC: liste des points d'arrêt
- REC: effacement des points d'arrêt

Sur les clés visualisation et chargement en mémoire il existe la possibilité d'incrémentatıon automatique.

2.4.2. - Les éditions de message sont :

- ERD \sqcup nn: message d'erreur sur commande inacceptable ou entrée dans AID par ACTD ou alarme non attendue.
- BP P=h : entrée dans AID sur point d'arrêt (instruction ou mémoire).
- P=h : valeur du pointeur éditée à l'entrée dans AID. Si l'on vient de dérouler une ou plusieurs instructions en pas à pas l'édition n'a lieu que s'il y a rupture de séquence.
- TN={^H_S}, h: N° de tâche édité en cas de changement de tâche.

Bull  VISUALISATION ET CHARGEMENT

3.1 - Visualisation : Clé (/)

Syntaxe : $\left[\begin{matrix} \{h\} \\ \{r\} \end{matrix} \right]$ (/)

Exemple : A/FF14 Visualisation du registre A
 '235/OFF2 Visualisation de la mémoire '235

Cette commande permet de visualiser en hexadécimal le contenu du registre r ou de la mémoire d'adresse hexadécimale h.

3.2 - Chargement : Clé (=)

Syntaxe : $\left[\begin{matrix} \{r\} \\ \{h\} \end{matrix} \right]$ (=) h2 (cr)

Exemple : X = '15 Chargement du registre X par '15
 '1200 = '2BC4 Chargement de la mémoire '1200 par '2BC4

Cette commande permet de modifier le contenu du registre r ou de la mémoire d'adresse hexadécimale h1 par la valeur h 2 donnée en hexadécimal. (le registre HV n'est pas chargeable).

3.3 - Incrémentation automatique

Après la visualisation ou le chargement d'une mémoire, la partie adresse d'une nouvelle commande de ce type peut être omise. Dans ce cas, la mémoire concernée reste la même si on change de clé sinon on passe à la mémoire suivante.

Exemple : '239 = '10A4 Chargement de la mémoire '239
 /'10A4 Visualisation de la mémoire '239
 /'FFA0 Visualisation de la mémoire '23A

NOTA : L'incrémentation automatique est inhibée après l'impression d'un message d'erreur, et l'utilisation d'une clé autre que visualisation ou chargement.
Les chiffres hexadécimaux soulignés sont imprimés par AID, ce qui n'est pas souligné étant frappé par l'opérateur.

4 - MODES D'EXECUTION

4.1 - Pas à pas

4.1.1. - Pas à pas simple : Syntaxe (>)

Cette commande entraîne l'exécution d'une seule instruction du programme testé.

- AID reprend ensuite le contrôle en imprimant un message indiquant :
- la nouvelle valeur du pointeur s'il y a eu rupture de séquence.
 - le type et le numéro de la tâche en cours si ceux-ci ont changé.

Exemple :

```

      P = `21C
    >                                     << Exécution de `21C
    >P = `240
    >                                     << Exécution de `240
    >P = `2110   TN=S,0004
  
```

4.1.2. - Pas à Pas multiple = syntaxe : (n) avec $1 \leq n \leq 9$

Cette commande entraîne l'exécution de n instructions du programme testé.

Après chaque instruction il y a un retour dans Ho invisible à l'opérateur. Après les n instructions, il y a retour sous contrôle de AID avec en général édition de P et éventuellement de TN, puis attente de commande.

4.2 - Continu : CN

Syntaxe : CN (cr)

Sur cette commande, AID redonne le contrôle au programme utilisateur en mode continu. AID ne pourra alors reprendre le contrôle que sur une alarme, une activation explicite (ACTD) ou une condition d'arrêt.

Exemple :

| | |
|-----------------|--------------------------------|
| CN | Demande d'exécution en continu |
| ERD 04 | Arrêt sur alarme N°4 |
| <u>P = `234</u> | |

L'exécution de l'instruction située en `234 a provoquée l'alarme N°4 (instruction privilégiée).

5 - POINTS D'ARRÊT

5.1 - Commandes individuelles

DB h (cr) pose d'un point d'arrêt.
 RB h (cr) effacement d'un point d'arrêt.

L'arrêt se produit si le programme tente de dérouler l'instruction sur laquelle a été placé le point d'arrêt (arrêt avant l'instruction et édition de P = h pointant sur l'instruction) ou si le programme tente de dérouler une instruction dont l'opérande comporte un point d'arrêt (arrêt après l'instruction et édition P = h + 1).

L'adresse h sur 16 bits communiquée par l'opérateur est une adresse absolue si le programme appelant est maître, relative si l'appelant est esclave.

L'arrêt se traduit par une entrée sous Ho et l'édition suivante :

BP = h (Cr)

Les points d'arrêt placés sur les buffers d'entrée-sortie des canaux ne provoquent pas d'arrêt.

Le chargement par le programme d'une mémoire détruit un point d'arrêt posé précédemment.

5.2 - Commandes Globales

Syntaxe : LEC (Cr) Liste des points d'arrêt
 REC (Cr) Effacement des points d'arrêt

Ces deux commandes portent sur toute la zone mémoire accessible au programme appelant ($0 \leq Ad \leq \min \left\{ \begin{array}{l} 64K \\ \text{MAXMEM} \end{array} \right\}$ ou $SLO \leq Ad \leq SLE$).

La première édite tous les points d'arrêt de la zone mémoire en partant des adresses les plus élevées.

La seconde se traduit seulement par un "line feed" lorsque la remise à zéro est terminée.

En cas de déplacement par l'opérateur de $SL\emptyset$, SLE, ces fonctions balayent la zone modifiée si le programme testé est esclave. S'il est maître c'est de toute façon la zone 0,64 K qui est balayée.

6 - CONSEILS PRATIQUES

6.1 - Essais en mode esclave

L'exécution du programme sous contrôle n'étant pas simulée, celle-ci peut donc éventuellement détruire des mémoires d'AID.

Pour éviter cet inconvénient, il est conseillé d'essayer ses programmes en mode esclave à chaque fois que ceux-ci n'utilisent pas d'instruction.

Pour cela il faut simplement :

- charger le programme en mode esclave
- sous AID initialiser SLO à la valeur de début de chargement.
- initialiser SLE à une valeur suffisante.
- mettre le programme en mode esclave en initialisant le bit 0 de ST à 0.

La mise au point se poursuit normalement en manipulant des adresses relatives à SLO.

D'une manière générale si on modifie SLO-SLE en testant un programme esclave, toute fonction, visualisation, chargement, DB, RB, LEC, REC utilise les valeurs modifiées.

En mode maître on peut mettre à zéro le bit 0 du status ST pour provoquer artificiellement l'activation du DRPS. Dans ce cas, on peut positionner SLØ au delà de 64K, et effectuer toutes les opérations précédentes dans une zone esclave quelconque.

6.2 - Utilisation du téléimprimeur

AID travaille avec le téléimprimeur de service à l'adresse débanalisée connue par la machine (sélecteur bootstrap) Si l'on dispose de plusieurs téléimprimeurs le cas le plus simple est celui où le téléimprimeur débanalisé sert au dialogue de AID et les entrées-sorties programme se font sur les autres.

Dans le cas où l'on voudrait déplacer le dialogue de AID sur un autre téléimprimeur, il faudrait modifier au pupitre le premier mot du commun de AID, la modification étant prise en compte au moment du lancement.

Dans le cas où les entrées-sorties programme se font sur le même téléimprimeur que AID on doit dérouler en continu les séquences d'entrée-sortie.

Il est souhaitable d'aligner les programmes utilisateur sur ce qui est fait dans le driver téléimprimeur.

Tout échange téléimprimeur est précédé de l'envoi d'un mot de commande.

Le mot de commande positionne le mode écho avant chaque entrée de commande et le supprime avant chaque édition de message.

6.3 - Interruptions et masque des interruptions

6.3.1. - Masque

AID n'intervient pas sur le masque des interruptions IM ni sur les 3 bits de masque du status calculateur ST. C'est à l'opérateur sous AID de les positionner à sa convenance (le bit 0 de IM concerne le pupitre).

6.3.2. - Modification ST dues à AID

De manière analogue au pupitre, AID fait monter le bit "debug" de ST sur un positionnement de point d'arrêt, mais il force en plus à 0 le bit "pop" de ST, ce qui fait que le passage sur un point d'arrêt provoque un lancement de Ho et non un arrêt pupitre. (Si des points d'arrêt pupitre ont été précédemment placés, AID les transforme en points d'arrêt "software" destinés à lancer Ho).

Bull voyant "debug" ne s'allume pas au pupitre malgré le passage en mode "debug" (Il s'allumerait à la première manipulation au pupitre).

Le mode "debug" retombe lors de la fonction REC.

NB. Lorsqu'on relance en continu, on ignore un point d'arrêt instruction éventuel qui serait placé sur la mémoire pointée elle-même.

6.3.3. - Appel superviseur (touche break) :

Cet appel provoque la mise sous tension du téléimprimeur.

En outre, dans l'hypothèse où l'utilisateur travaille sous un superviseur (BØS,...) comprenant le moniteur d'entrées-sorties IØCS ce dernier exécute une instruction ACTD à l'occurrence d'un appel "4coups" (voir BØS,...) Cette action provoque dont un retour sous AID quand celui-ci est actif.

L'utilisateur peut à ce point quitter le niveau hardware sur lequel se trouve le téléimprimeur en exécutant la commande "un pas" (>).

6.4 - Utilisation de l'option scheduler microprogrammé.

Cette utilisation ne pose pas de problème tant que l'on n'utilise que les instructions relatives à cette option : ARM, QUIT, RQST, RLSE, ACT, WAIT.

Des problèmes se posent malgré tout si l'on modifie les files systèmes par d'autre moyens.

Le cas peut se présenter par exemple si une tâche modifie l'état de tâches plus prioritaires.

En effet, d'après le principe même d'AID, en mode pas à pas il s'effectue un passage dans le scheduler et l'on doit donc à chaque pas disposer les files système dans un état correct si l'on ne veut pas provoquer l'élection de tâches non désirées.

Quand on ne peut éviter cette situation, au cours de la mise au point il est alors nécessaire de dérouler les séquences en continu en plaçant des point d'arrêt à la sortie de ces séquences.

7 - MISE EN OEUVRE

7.1 - Version autonome

Avant tout chargement de la version autonome, il est nécessaire d'initialiser les mémoires suivantes :

- APSTH (adresse '2) : relais de la table des PST hardware
- APSTS (adresse '3) : relais de la table des adresses des PST software

De plus AID utilise certaines mémoires débanalisées :

NS (adresse '0) : reprise à zéro en initialisation
INI (adresse '8) : l'adresse de lancement de AID y est forcée
MAXMEM (adresse 'A) : adresse de la dernière mémoire de la configuration.

Il y a élection de la tâche Soft 0 dans les files système au moment du lancement.

Le chargement s'effectue normalement à partir de la bande binaire translatable et le lancement s'effectue automatiquement avec impression du message :

$$\frac{ERD}{P} = \frac{09}{h} \quad \underline{TN = S'0000}$$

et passage en attente de commande.

Pour effectuer son initialisation, AID crée une PST software de niveau 0, que l'utilisateur peut ensuite utiliser, remplacer, ou abandonner à son gré, par contre pour son exécution, AID s'attribue le niveau hardware 0 et l'utilisateur ne doit pas utiliser la PST de Ho ni le relais APSTH.

7.2 - Version sous système (BOS)

Toutes les initialisations de PST sont réalisées par BOS et AID utilise le niveau software de BOS pour son initialisation. AID n'utilise pas la mémoire INI, qui reste donc pointée sur BOS. La mémoire NS n'est pas chargée. Les files système ne sont pas modifiées.

Deux autres mémoires sont utilisées :

ENTBOS : (adresse '7) : adresse de retour sous BOS
SYSMEM : (adresse 'c) : joue le rôle de MAXMEM en autonome.

Cette version se charge comme tout processeur sous BOS (voir notice BOS) et possède la clé d'activation IDBG Cr

De plus sous le contrôle d'AID la clé BOS Cr permet de redonner le contrôle à BOS si la tâche en cours n'est pas sous niveau hardware. La priorité de la tâche en cours (autres que AID) doit être celle de BOS.

NOTATIONS

[.....]

élément syntaxique optionnel

{.....}

éléments syntaxiques à choisir

h

nombre hexadécimal : 1 à 4 chiffres hexadécimaux précédés d'une apostrophe

Exemple : 'FFFF, '2A

r

Nom de registre : A, B, X,.....

Ⓞ cr

Retour chariot

Ⓞ If

Interligne

Règles d'écriture des commandes

Dans AID les espaces ne sont significatifs ni en nombre ni en position, et l'on accepte les caractères de correction :

Ⓞ ←

Pour annuler toute la commande en cours.

Ⓞ ↑

Pour annuler le dernier caractère frappé.
On peut répéter cette correction :

Exemple : BC3 ↑ 4 est équivalent à BC4.

Les interlignes et les lignes vides sont acceptées.

NOTA : Dans les exemples donnés, les caractères et messages émis par AID sont soulignés.

SYNOPTIQUE

Visualisation : Mémoire [h] /
 : Registre r /

Chargement : Mémoire [h] = h (cr)
 : Registre r = h (cr)

Exécution : Pas à Pas > pour 1 pas ; h> pour h pas
 Continu CN (cr)

Retour à BOS : BOS (cr) version sous BOS seulement.

Pose d'un arrêt d'instruction : DB h (cr)

Effacement d'un arrêt instruction : RB h (cr)

Liste des arrêts : LEC (cr)

Effacement des arrêts : REC (cr)

MESSAGES D'ERREUR

Forme : ERD nn

nn

- 00 Alarme "mémoire inexistante"
- 01 Alarme "protection mémoire"
- 02 Erreur de parité hors du mode debug
- 03 Instruction optionelle inexistante
- 04 Alarme "instruction privilégiée"
- 05 Alarme "RQST, QUIT, WAIT" sous niveau hardware
- 06 Appel d'un autre processeur (IPI)
- 09 ACTD (activate DEBUG)
- 10 Erreur de parité en entrée de commande ou commande trop long
- 11 Commande incorrecte
 - commande inconnue
 - registre inconnu
 - syntaxe incorrecte
 - incrémentation automatique inacceptée
- 12 Nombre hexadécimal incorrect
- 13 Adresse inacceptable
- 14 Violation d'AID
- 15 Commande BOS non autorisée sous niveau hardware
- 16 Nombre de pas demandé hors de la plage 1 ≤ ≤ 9

NB : l'alarme 07 (STEP) ne doit jamais apparaitre puisqu'elle est toujours attendue par AID.
l'alarme 08 (Point d'arrêt en mode debug) ne doit jamais apparaitre non plus.