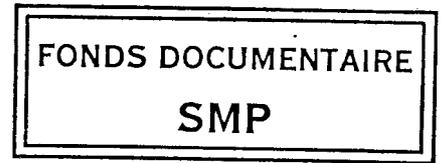


SOLAR

UNITÉS CENTRALES

Test du système d'I.T.



TEST DU SYSTEME D'IT

MANUEL D'UTILISATION

- Δ en haut de page indique le changement complet de la page par rapport à l'IE précédent
- | en marge indique la partie modifiée par rapport à l'IE précédent



SOMMAIRE	Pages
1 -BUT DU TEST	1 -1
1. 1 - MATERIEL TESTE	1 -1
2 - CONCEPTION DU TEST	2-1
3 - UTILISATION DU TEST	3-1
3.1- MOYENS NECESSAIRES A L'UTILISATION DU TEST	3-1
3.2 - CHARGEMENT ET LANCEMENT DU TEST	3-2
3.3 - DESCRIPTION DU CONVERSATIONNEL DU TEST	3-2
Clés disponibles	3-3
Clés Actions	3-3
Clés Recettes	3-3
Clés Outils	3-3
Rappel des instructions flottantes	3-4
Description des clés	3-5
Clé 100 A	3-5
Clé 200 A	3-5
Clé 300 A	3-5
Clé FLD O	3-5
Clé FST O	3-5
Clé NEG O	3-6
Clé ABS O	3-6
Clé CMS O	3-6
Clé FIX O	3-6
Clé POL O	3-7
Clé TIT O	3-9
Clé INS O	3-10
Clé IMQ O	3-11
Clé INT O	3-12
Clé IIP O	3-13
Clé DEP O	3-14
Clé XAR O	3-15
Clé BX2 O	3-16



A - 1 - Annexe 1 Tableau des clés	3-17
A - 2 - Annexe 2 Message d'erreur	3-18
A - 3 - Annexe 3 Liste des numéros d'erreur	3-19



AVERTISSEMENT

Ce manuel ne peut être bien compris par le lecteur que si celui-ci a déjà pris connaissance du contenu du manuel d'utilisation des programmes de test sous noyau SOLAR, dit "NOYAU DE TEST" (réf. : 1 158 000 00 / - 30 XX).



1 -BUT DU TEST

- Tester sur l'ensemble de la gamme SOLAR les instructions nécessaires au traitement d'une interruption.
- Tester les instructions flottantes existantes en standard sur la gamme SOLAR 16.
- Tester les instructions propres au module DSB 65 sur processeurs 16-65 (structure polybus), 16-70 et 16-90.

1.1 - MATERIEL TESTE

- Les processeurs de la gamme SOLAR.





2 - CONCEPTION DU TEST

- Le programme de test "Instructions Spéciales" se déroule en autonome sous noyau de test SOLAR.
- Le programme nécessite la présence : de l'horloge temps réel et du DRPS.
- Les clés actions ne sont que le regroupement des clés outils
 - clé 100 = enchaînement des clés outils
FLD - FST - NEG - ABS - CMZ - CAZ - FIX
 - clé 200 = enchaînement des clés outils
POL - TIT - INS - IMQ - INT - II P
 - clé 300 = enchaînement des clés outils
DEP - XAR - BX2

- Avant de se lancer chaque clé vérifie sa compatibilité avec le processeur testé.



3 - UTILISATION DU PROGRAMME

3.1 - MOYENS NECESSAIRES A L'UTILISATION DU TEST

- Moyens matériels

- Calculateurs SOLAR 16 de capacité mémoire ≥ 16 K
- Organe de dialogue

- Moyens logiciels

- Chargeurs absolu SOLAR
- Moyen de test SOLAR (1.158.000.01)
- Bande de test des Instructions Spéciales (1. 158. 201. 00)

- Documentation

- Présentation du "Système de test du SOLAR 16" (1. 158. 000), dit "NOYAU DE TEST".
- Le présent manuel.



3.2 - CHARGEMENT ET LANCEMENT DU TEST

Se reporter à la présentation du "Système de test SOLAR 16", manuel "NOYAU DE TEST".

3.3 - DESCRIPTION DU CONVERSATIONNEL DU TEST

S'il manque le DRPS

MANQUE L'OPTION DRPS

Si SOLAR 16-65

TESTEZ-VOUS LE MODULE (DSB 65-65) ? Y - N

HTR DEBANALISEE ? Y - N

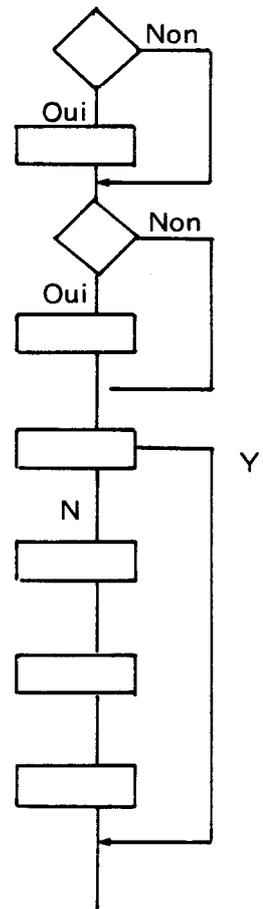
ADRESSE HTR ?

NIVEAU IT ?

S.NIVEAU ?

DONNEZ VOS CLES

01





Exemple de conversationnel

TESTEZ-VOUS LE MODULE (DSB 65 - 65) ? Y
HTR DEBANALISEE ? N
ADRESSE HTR ? ' 17 FA
NIVEAU IT ? 13
S/NIVEAU ? 0
DONNEZ VOS CLES
01

Clés disponibles

Clé 100 : test de l'ensemble des instructions flottantes existant sur les processeurs 16-40 et 16-65.
Clé 200 : test du système d'interruption du SOLAR.
Clé 300 : Test des instructions spécifiques au processeur 16-65 P.

Recettes : REC et RNS

La recette REC comprend toutes les clés action : 100 - 200 - 300.

La recette RNS permet de faire des tests de longue durée et assure le rebouchage à l'infini des clés actions 100 - 200 - 300.

Clés outils

Test des instructions flottantes

FLD - FST - NEG - ABS - CMZ - CAZ - FIX.

Test du système d'interruption

POL - TIT - INS - IMQ - INT - IIP.

Test des instructions spécifiques aux modules DSB 65

DEP - XAR - BX2.



Rappel des instructions flottantes

Une instruction flottante est codée sur deux mots mémoire :

1er mot	'3800	
2e mot	'Indique l'opération	
(AE)	Contenu de l'adresse effective	
FLD	Floating point Load	'3800
		'- 0 - -
	Registres modifiés A,B	<u>A B := (A E)</u>
	Indicateurs $I_1 = I_2 = 0$	
FST	Floating point store	'3800
		'- 8 - -
	Indicateurs	<u>(A E) := A,B</u>
FCAZ	Floating point compare accumulator to zéro	'3800
		'2800
	Indicateurs	I1 I2
		0 0 AB < 0
		1 0 AB = 0
		0 0 AB > 0
FCMZ	Floating point compare memory to zéro	'3800
		'- C - -
	Indicateurs	I1 I2
		0 1 AE < 0
		1 0 AE = 0
		0 0 AE > 0
FIX	Fix	'3800
		'1000
	Registres modifiés A, B	A := Fix (A, B)
		B := 0000
	Indicateurs	$I_1 = 1$ si impossible
		$I_2 = 0$
FNEG	Floating point negate	'3800
		'0000
	Registres modifiés A, B	<u>A, B := - A, B</u>
	Indicateurs	$I_1 = I_2 = 0$



FABS	Floating point absolute value	'3800
		'0800
	Registres modifiés A, B	<u>A, B := A, B</u>
	Indicateurs	$I_1 = I_2 = 0$

Clé 100 A

Format : 100 RC

But : test des instructions flottantes

Contenu : enchaînement des clés outils

FLD - FST - NEG - CMZ - CAZ - FIX - ABS.

Message d'erreur : voir annexe 3
erreurs 01 à 07

Clé 200 A

Format : 200 RC

But : test du système d' I T

Contenu : enchaînement des clés outils

POL - TIT - ~~INS~~- IMQ - INT - I IP

Message d'erreur : voir annexe 3
erreurs 10 à 57

Clé 300 A

Format : 300 RC

But : Test des instructions spécifiques au processeur 16 - 65 P

Contenu : Enchaînement des clés outils

DEP - XAR - BX2

Message d'erreur : voir annexe 3
erreur 60 à 87

Description des clés

Clé FLD O

Format : FLD RC

But : Test de l'instruction flottante FLD

Contenu : Exécute puis simule l'instruction FLD
Vérifie qu'il y a concordance entre les deux résultats

Message d'erreur : voir Annexe 2

Clé FST O

Format : FST RC

But : Test de l'instruction flottante FST

Contenu : Exécute puis simule l'instruction FST
Vérifie qu'il y a concordance entre les deux résultats:

Message d'erreur : voir Annexe 2



Clé NEG O

Format NEG RC

But Test de l'instruction flottante FNEG

Contenu Exécute puis simule l'instruction FNEG
Vérifie qu'il y a concordance entre les deux résultats

Message d'erreur : Voir annexe 2.

Clé ABS O

Format ABS RC

But Test de l'instruction flottante FABS

Contenu Exécute puis simule l'instruction ABS
Vérifie qu'il y a concordance entre les deux résultats.

Message d'erreur : Voir annexe 2.

Clé CMZ O

Format CMZ RC

But Test de l'instruction flottante FCMZ

Contenu Exécute puis simule l'instruction FCMZ
Vérifie qu'il y a concordance entre les deux résultats:

Message d'erreur : Voir annexe 2.

Clé CAZ O

Format CAZ RC

But Test de l'instruction flottante FCAZ

Contenu Exécute puis simule l'instruction FCAZ
Vérifie qu'il y a concordance entre les deux résultats:

Message d'erreur : Voir annexe 2.

Clé FIX O

Format FIX RC

But Test de l'instruction flottante FIX

Contenu Exécute puis simule l'instruction FIX
Vérifie qu'il y a concordance entre les deux résultats:

Message d'erreur : Voir annexe 2.



Clé POL O

Format POL RC

BUT Test du Polling IT HTR

Contenu

La clé vérifie pour le module hardware testé que les interruptions I/O normales ou exceptions arrivent bien sur le niveau correct et le sous-niveau exception correcte et qu'il n'y a pas d'appels parasites:

Avant de faire le polling nous masquons les I/O (bit à IOM du Status).

On vérifie tout d'abord que le niveau d'interruption est correct:

Puis, sur chaque niveau d'interruption de 1 à 15, on fait le polling du bloc normal et des blocs exceptions 0, 1, 2. Pour tous les niveaux d'interruption autre que celui du niveau du module hardware, on attend sur ces pollings.

Rappelons ici

- le niveau testé : dépend du conversationnel
- le sous-niveau exception : O.

Messages d'erreurs

ERR (100/97) HLW

POLLING ATTENDU : 0000 0000 0000 0000

POLLING RECU : 0000 0000 0000 0000

d'erreur sur le bloc normal SLWN

ERR (100/98) I/O NORMAL NIVEAU 'OE

POLLING ATTENDU : 0000 0000 0000 0001

POLLING RECU : 0000 0000 0000 0000

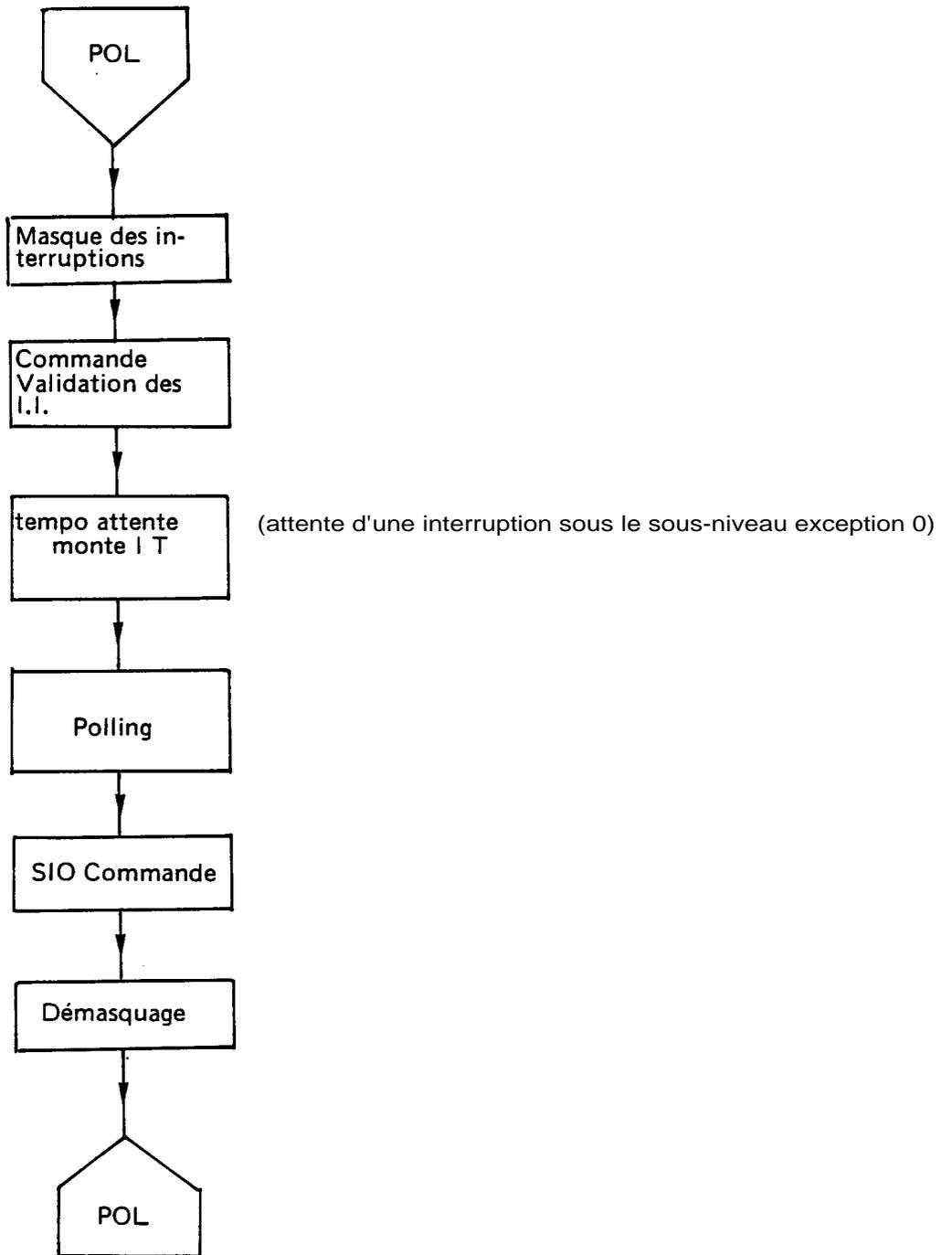
d'erreur sur l'un des blocs exceptions

ERR (100/99) I/O exception bloc 01 niveau 'OE

POLLING ATTENDU : 0000 0000 0000 0001

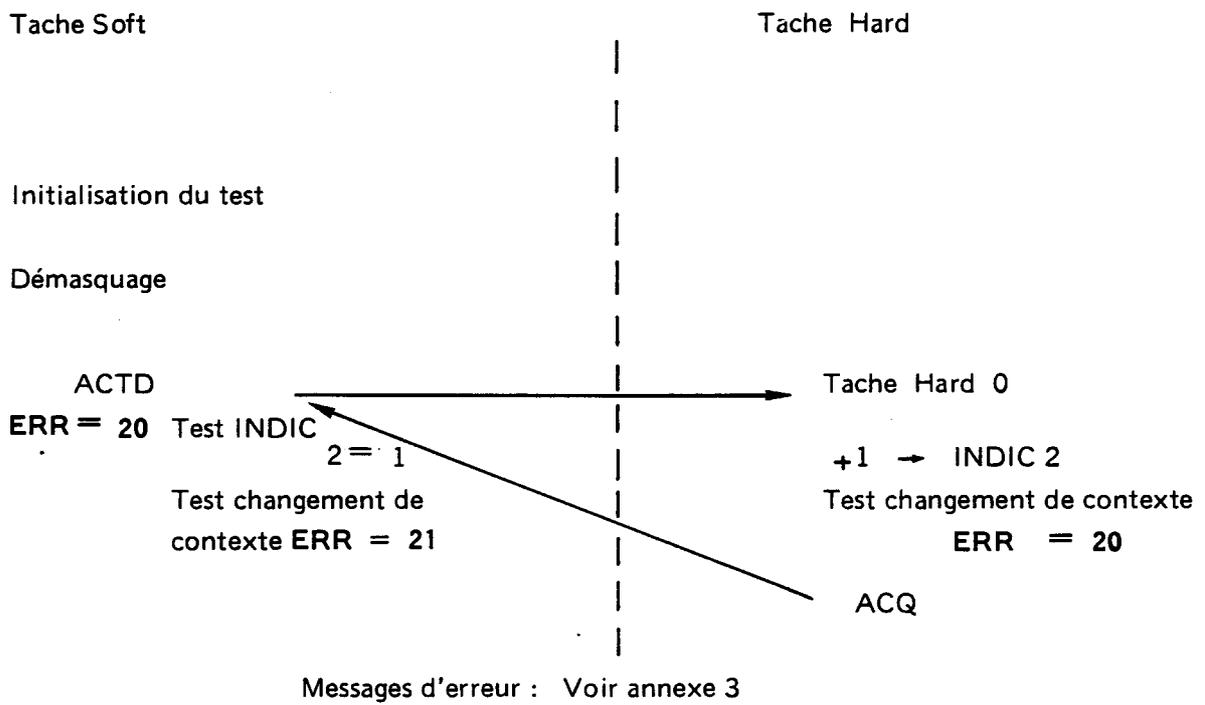
POLLING RECU : 0000 0000 0000 0000.

Organigramme





Clé TIT O
Format TIT RC
But Test de traitement d'une IT
Contenu





Clé INS O

Format

INS RC

But

Tester les instructions de lectures de niveau d'I.T., de sous-niveau, les registres correspondants.

Contenu

Tache Soft

Initialisation du test

Démasquage

Lancement HTR

Tempo

Test A = 0

ERR 33

ACTD

Test A = 0

RDHV

ERR 35

Tache Hard

Tache Hard HTR

Arrêt HTR

RDHV **ERR 30** Test A = Niv.HTR

ACK **ERR 31** Test x = S Niv.HTR
Test C, V

ACK **ERR 32** Test x = 0
Test C, V

ACQ

Tâche Hard 0

RDHV

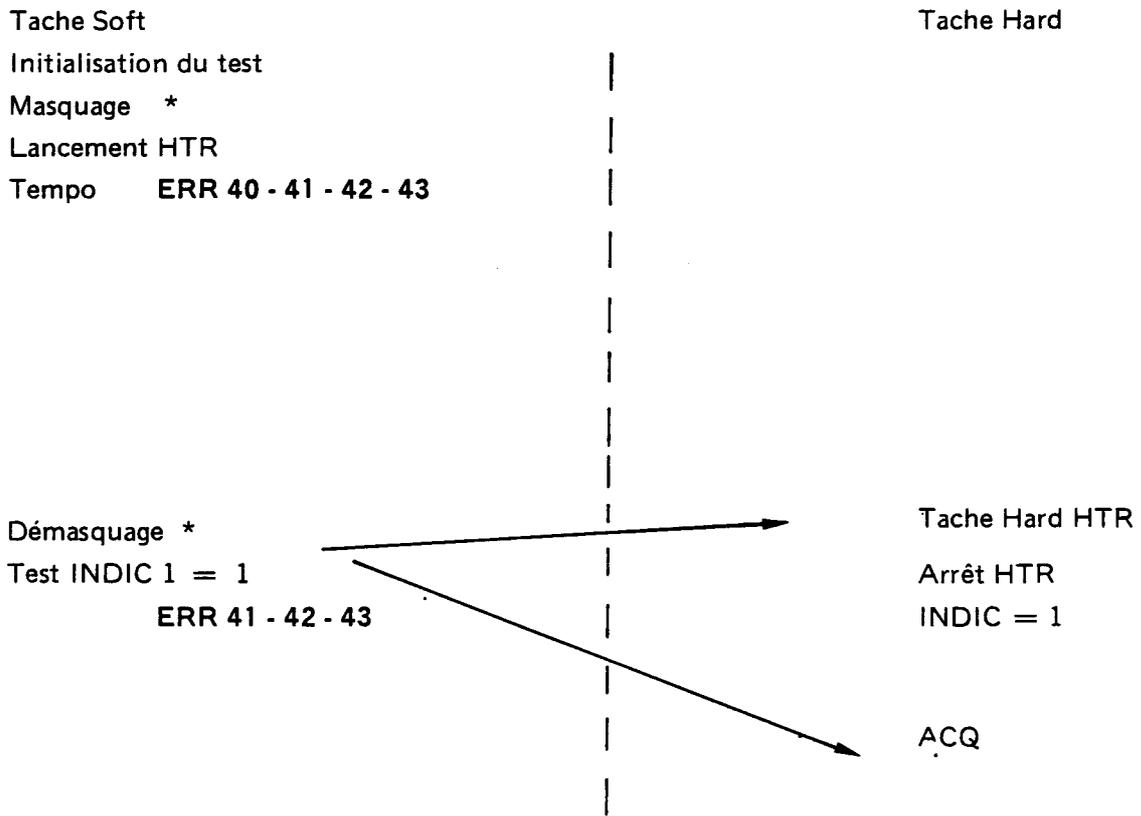
ERR 34 Test A = '8000

Test alarme = 9

Message d'erreur - voir annexe 3



Clé IMQ O
Format IMQ RC
But Test des instructions de masquage et de démasquage
Contenu



* Masquage - démasquage : SST - DIT - XIMR
RST - EIT - XIMR
- ACQ sous tache soft pour créer un polling

Message d'erreur : voir Annexe 3.

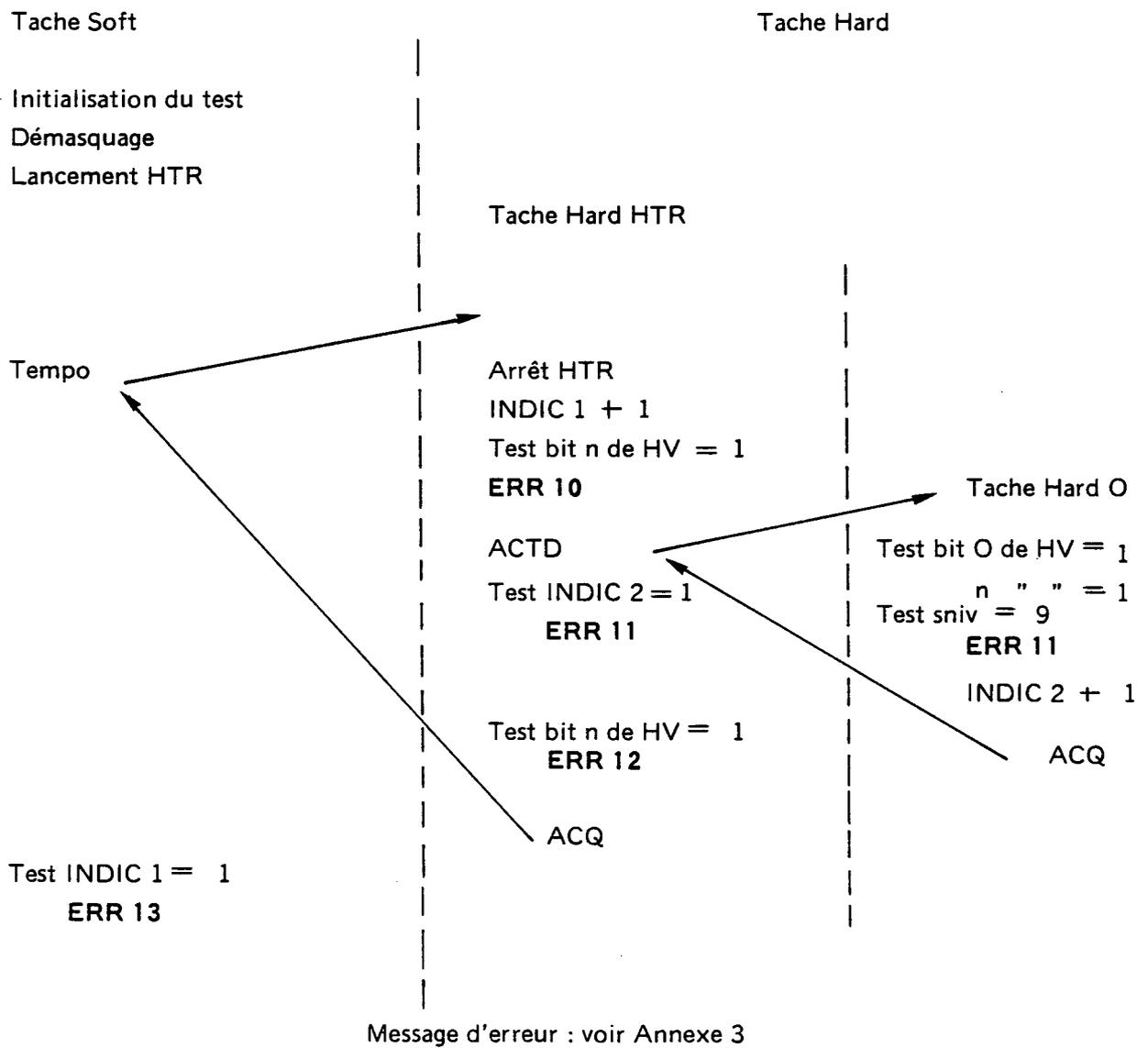


Clé INT O

Format INT RC

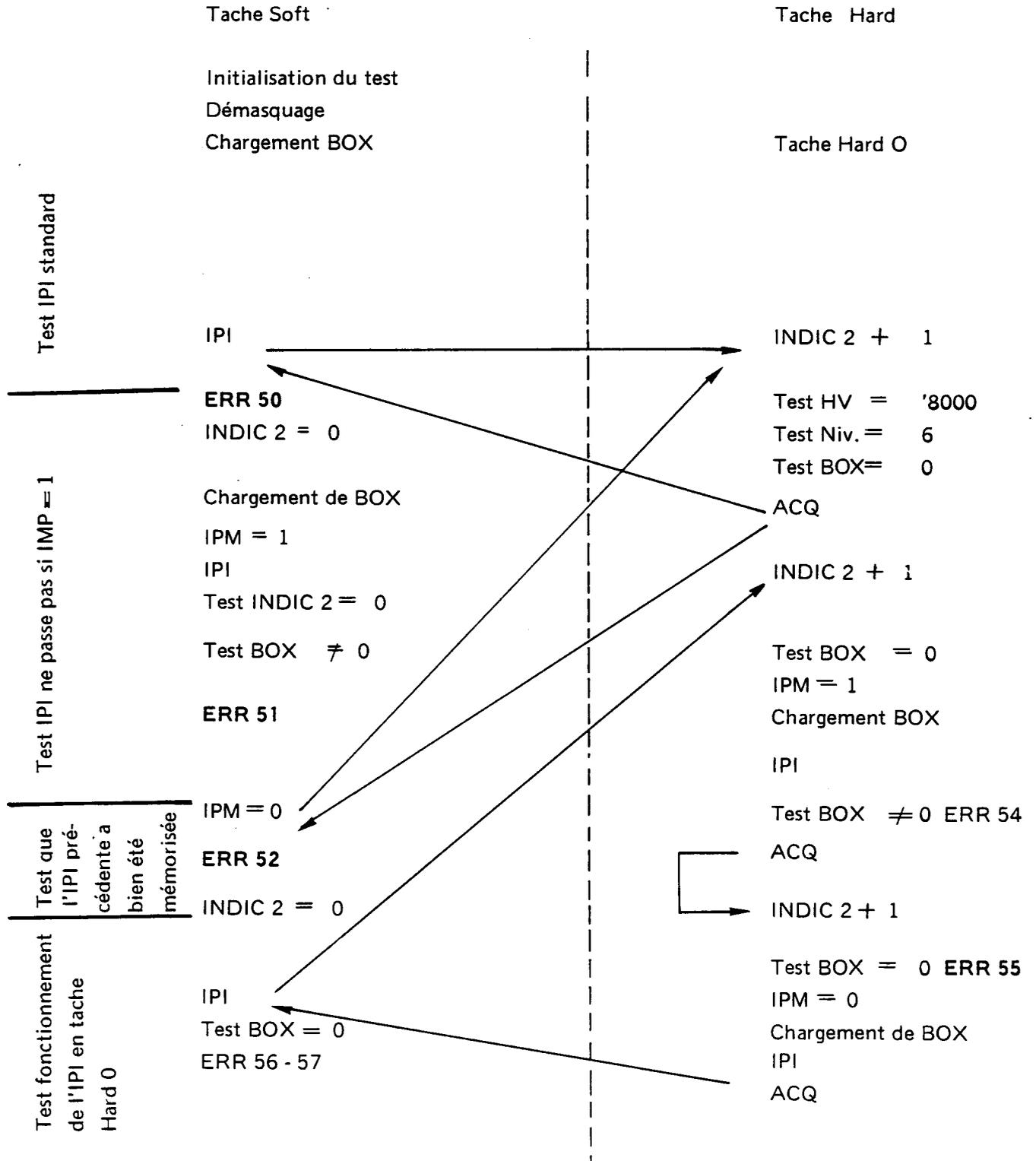
But Vérifier l'interruptibilité d'une tâche Hard par une IT de niveau plus prioritaire:

Contenu



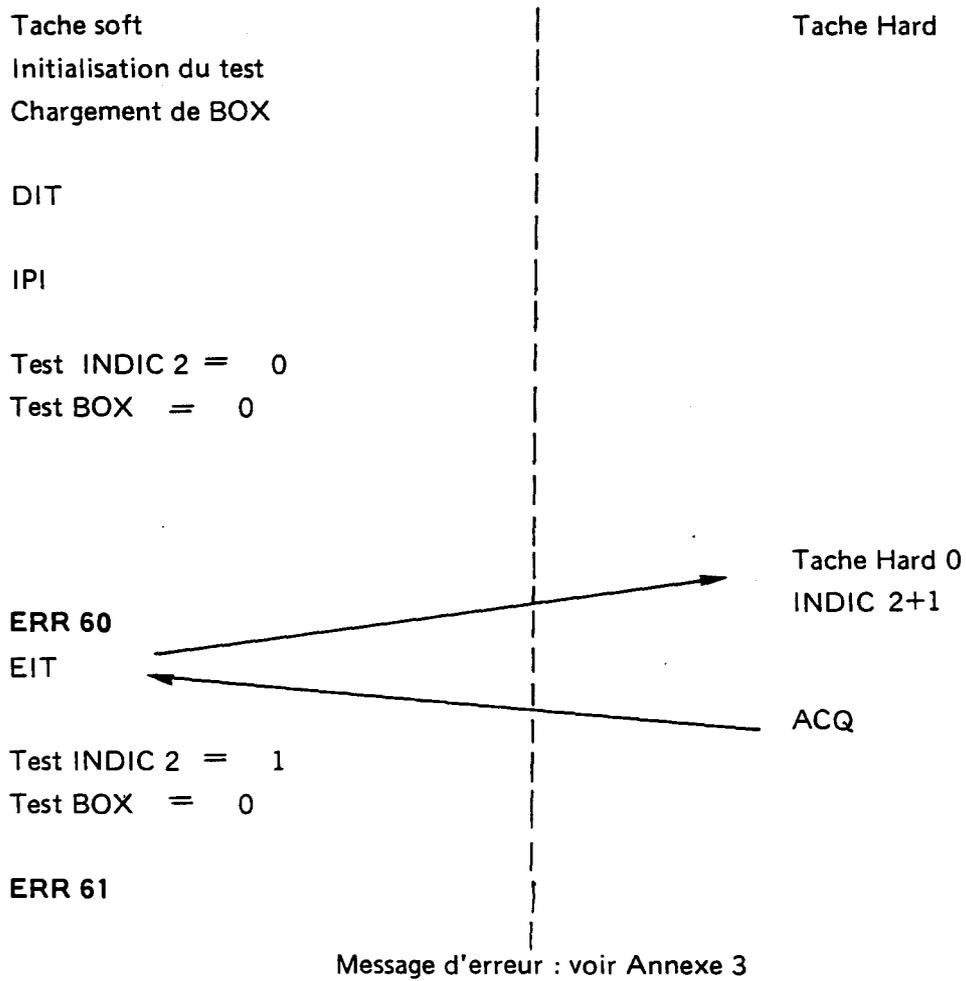


Clé IIP O
Format IIP RC
But Tester les interruptions inter processeurs
Contenu





Clé DEP O
Format DEP RC
But Test des instructions DIT, EIT sur le 65 P
Contenu





Clé XAR O

Format XAR RC

But Test de l'instruction XAR

Contenu

Tache Soft

Initialisation du test

Démasquage

SLO et SLE = fond de mémoire

- à l'adresse fond de mémoire écriture de 'AAAA
avec l'instruction STAR
- échange simultané de l'adresse fond de mémoire et (A = - 1)

A = 'AAAA : ERR 70

- Lecture de fond de mémoire par l'instruction LAR

(A = - 1) ERR 71

XAR à une adresse inexistante

Test si alarme 0

ERR 72

Nota : Les erreurs 73 et 74 entraînent la suppression du test.

Message d'erreur : Voir Annexe 3



Clé BX2 O

Format : BX2 RC

But : Tester les interruptions inter processeurs

Contenu :

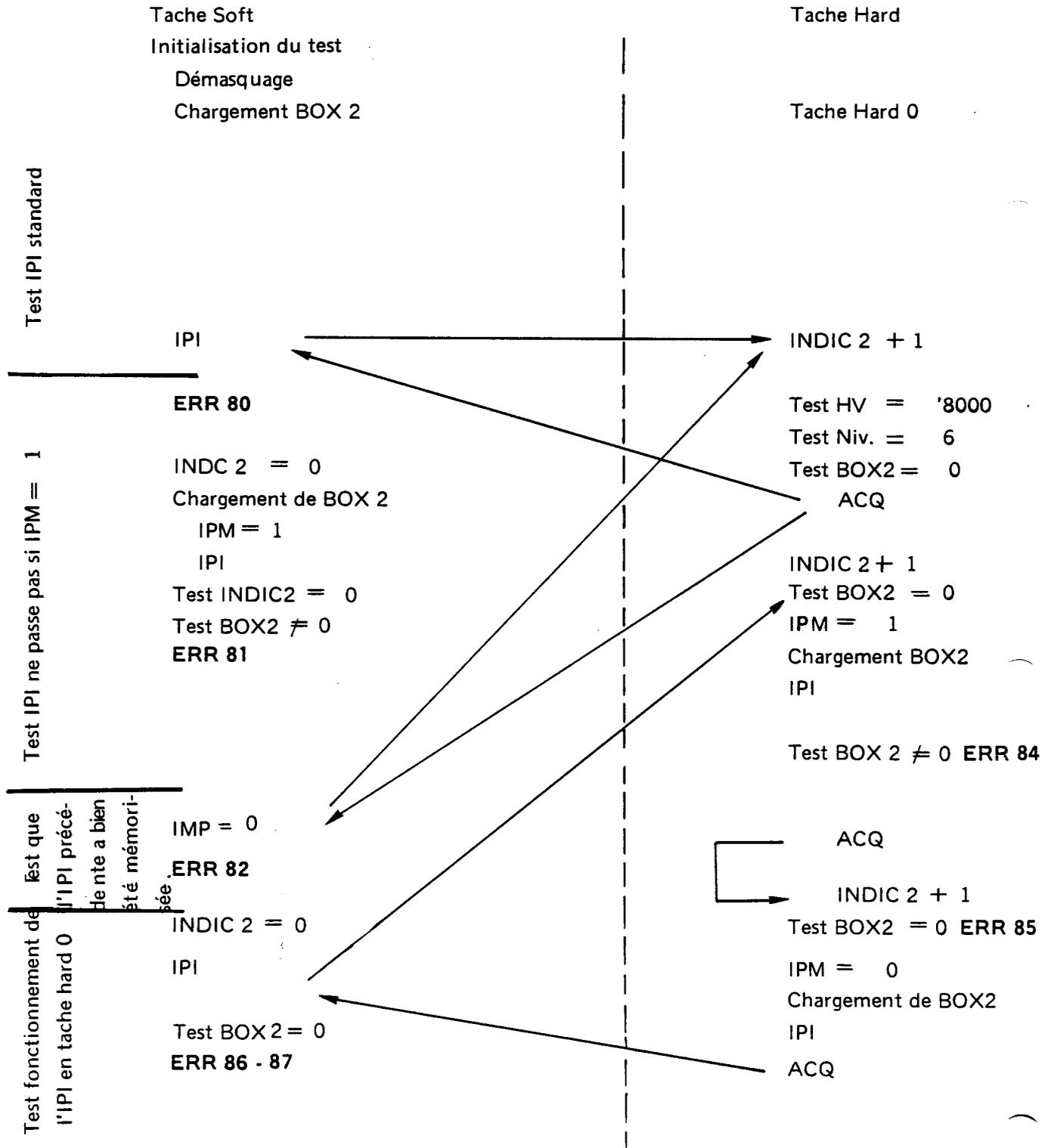


TABLEAU DES CLÉS - ANNEXE 1



Renseignement clés			Composi- tion Recette			Mode fonc- tionement										
											05	30 35 40	70 90 65			
100	A	0		o	o	o						o	o			Test des Instructions flottantes
200	A	0		o	o	o					o	o	o			Test du système d' IT
300	A	0		o	o	o							o			Test des Instructions spéciales du DSB 65
FLD	O	0										o	o			Test de l'instruction FLD
FST	O	0										o	o			Test de l'instruction FST
NEG	O	0										o	o			Test de l'instruction FNEG
ABS	O	0										o	o			Test de l'instruction FABS
CMZ	O	0										o	o			Test de l'instruction FCMZ
CAZ	O	0										o	o			Test de l'instruction FCAZ
FIX	O	0										o	o			Test de l'instruction FIX
POL	O	0									o	o	o			Test de l'IT HTR (Polling)
TIT	O	0									o	o	o			Test du traitement d'une IT (changement contexté)
INS	O	0									o	o	o			Test des instructions de lecture du Niveau et s/niveau d'IT et registres correspondants
IMQ	O	0									o	o	o			Test des instructions de Masquage et démasquage
INT	O	0									o	o	o			Test de l'interruptibilité d'une tâche Hard
IIP	O	0										o	o			Test de l'interruption inter-processeur
DEP	O	0											o			Test du DIT - EIT du DSB 65
XAR	O	0											o			Test de l'instruction XAR du DSB 65
BX2	O	0											o			Test de l'IPI du DSB 65 (BOX2)
REC	S	0									o	o	o			Recette comportant toutes les clés actions
RNS	S	1									o	o	o			Recette de longue durée avec un nombre de message maximum donnée en paramètre.
PSW	S	0									o	o	o			Impos e un état de fonctionnement
BRL	S	0									o	o	o			Permet de boucler un certain nombre de fois à une ligne de Macro.
STS	S	1									o	o	o			Set Top Synchro.
RTS	S	1									o	o	o			Reset Top Synchro.
STO	S	0									o	o	o			Permet de stocker une phase de Test
RST	S	0									o	o	o			Permet de lancer une phase de Test définie par STO
END	S	0									o	o	o			Permet de terminer un Test:



ANNEXE 2 - MESSAGE D'ERREUR

Pour chaque clé, en cas d'erreur il est édité un message d'erreur

ERR (XXX / XX)

↑ ↑
mnémonique de clé no d'erreur

suivi des paramètres ci-après

```
OP1          : XXXXXX  XX
OP2          : XXXXXX  XX
RES HARD     : XXXXXX XX
  V = X              C = X
RES SOFT     : XXXXXX XX
  V = X              C = X
```

à chaque clé est affecté un no de message d'erreur

01	FLD	chargement défectueux
02	FST	rangement défectueux
06	FCAZ	comparaison défectueuse
05	FCMZ	comparaison défectueuse
04	FABS	valeur absolue défectueuse
03	FNEG	négation défectueuse
07	FIX	partie entière A,B défectueuse

LISTE DES NUMEROS D'ERREUR

ANNEXE 3

N°	Signification	
01	Erreur instruction FLD chargement défectueux	
02	" " FST rangement défectueux	
03	" " FCAZ comparaison défectueuse	
04	" " FCMZ " "	
05	" " FABS valeur absolue défectueuse	
06	" " FNEG négation défectueuse	
07	" " FIX partie entière A, B défectueuse	
10	Niveau d'IT HTR non attendu ERR (XXX/10) NIV. LU = oo ATT = oo	
11	ERR (XXX/11) PAS D'ALARME 9 / ACTD	
11	bit n de HV chiffrent de 1(0 = niveau HTR) ERR (XXX/11) HV LU = 0000 0000 0000 0000	Test de l'interrupti- bilité d'une tache
11	Alarme différente de celle attendue ERR (XXX/11) ALARME *	Hard
12	Niveau d' IT HTR non attendu ERR (XXX/12) NIV : LU = oo ATT = oo	INT
13	ERR (XXX/13) PAS D'IT HTR	
20	Erreur dans le changement de contexte en tache Hard 0 ERR (XXX/20) X LU = 0000 0000 0000 0000 ATTENDU = 0000 0000 0000 0000 X = Registre C, K ou P	Test du traitement d'une IT.
21	ERR (XXX/20) PAS D'ALARME 9 / ACTD Erreur dans le changement de contexte en retour de tache Hard ERR (XXX/21) X LU = 0000 0000 0000 0000 ATTENDU = 0000 0000 0000 0000 X = Registre C, K ou P	TIT
30	IT HTR différente de celle attendue ERR(XXX/30) NIV LU = oo ATT = oo	
31	Sous niveau différent de celui attendu ERR(XXX/31) NIV LU = oo ATT = oo	
32	ERR(XXX/31) INDICATEURS V, C = oo Après l'exécution de 2 "ACK" le registre X et les indicateurs sont différents de 0. ERR (XXX/32) NIV.LU = oo ATT = oo	Test des instructions de lecture de niveau et S/niveau d'IT



33	ERR (XXX/32) INDICATEURS V,C = ∞ ERR (XXX/33) PAS D 'IT HTR	
34	Après traitement de l'IT HTR HV ≠ de 0 ERR (XXX/34) NIV. LU = ∞ ATT = ∞ ERR (XXX/ 34) PAS D'ALARME 9 / ACTD Sur alarme 9 HV ≠ de '8000 ERR(XXX/34)HV LU = 0000 0000 0000 0000 Alarme reçue différente de 9 ERR (XXX/34) ALARME *	
35	HV différent de 0 ERR (XXX/35) NIV. LU = ∞ ATT = ∞ L'instruction SST ne marche pas	SST
40	ERR (XXX/40) IOM = 0 ERR (XXX/40) IT HTR/IOM = 1	Test des instructions de masquage et de démasquage
41	Pas d'IT HTR après démasquage de IOM ERR (XXX/41) PAS D'IT HTR Mauvais fonctionnement de l'Instruction XIMR	RST XIMR
42	ERR (XXX/41) IM LU = 0000 0000 0000 0000 IT HTR avec IM = - 1 ERR (XXX/42) IM LU = 1111 1111 1111 1111 Pas d' IT HTR après le démasquage de IM	IMQ
	ERR (XXX/42) PAS D'IT HTR L'instruction DIT ne masque pas IOM	XIMR DIT
43	ERR (XXX/ 42) IOM = 0 ERR (XXX/43) IT HTR/IOM = 1 Pas d' IT HTR après le démasquage de IOM	EIT
50	ERR (XXX/43) PAS D'IT HTR Pas d'alarme réveil processeur sur IPI ERR (XXX/50) PAS DE REVEIL PROCESSEUR Sur alarme réveil processeur HV ≠ '8000 ERR (XXX/50) HV LU = 0000 0000 0000 0000 Alarme ≠ de 6 ERR (XXX/50) ALARME * Pas de remise à 0 de la boîte aux lettres	Test des interruptions inter processeurs
51	ERR (XXX/50) BOX LU = 0000 0000 0000 0000 IT Réveil processeur malgré le masquage d'IPM ERR (XXX/51) REVEIL PROCESSEUR / IPM = 1 Remise à zéro de la boîte aux lettres sans IT	IIP
52	ERR (XXX/51) BOX LU = 0000 0000 0000 0000 Pas d'alarme réveil processeur sur IPI ERR (XXX/52) PAS DE REVEIL PROCESSEUR Sur alarme réveil processeur HV ≠ '8000 ERR (XXX/52) HV LU = 0000 0000 0000 0000	

54	Alarme ≠ de 6 ERR (XXX/52) ALARME *	
	Pas de remise à zéro de la boîte aux lettres ERR (XXX/52) BOX LU = 0000 0000 0000 0000	
55	Remise à zéro sur IPI masque ERR (XXX/51) BOX LU = 0000 0000 0000 0000	
56	Pas de remise à zéro de la boîte aux lettres ERR (XXX/55) BOX LU = 0000 0000 0000 0000	
57	Pas d'alarme réveil processeur sur IPI ERR (XXX/ 56) PAS DE REVEIL PROCESSEUR ERR (XXX/56) PAS D'IT/IPI EN TACHE HARD/IPM = 1 ERR (XXX/56) IT/IPI EN TACHE HARD/IPM = 0	DEP
60	Pas d'alarme réveil processeur sur IPI ERR (XXX/57) PAS DE REVEIL PROCESSEUR Pas de remise à zéro de la boîte aux lettres ERR (XXX/57) BOX LU =	
61	IT réveil processeur DIT ne fonctionne pas ERR (XXX/60) IPM = 0 IT réveil processeur IPM = 1 ERR (XXX/60) REVEIL PROCESSEUR/IPM = 1	Test du DIT - EIP du module DSB 65
70	Remise à zéro de la boîte aux lettres ERR (XXX/60) BOX LU = 0000 0000 0000 0000 Pas de réveil processeur après EIT ERR (XXX/61) PAS DE REVEIL PROCESSEUR Pas de remise à zéro de la boîte aux lettres ERR (XXX/61) BOX LU = 0000 0000 0000 0000	
73	Contenu de A après le XAR ≠ de AAAA ERR (XXX/70) A LU = 0000 0000 0000 0000 ATTENDU = 1010 1010 1010 1010	Test de l'instruction XAR
74	Contenu de l'adresse après le XAR ≠ - 1 ERR (XXX/71) A LU = 0000 0000 0000 0000 ATTENDU = 1111 1111 1111 1111 Pas d'alarme 0 sur XAR dans mémoire inexistante ERR (XXX/72) PAS D'ALARME O/XAR Sur alarme 0 HV ≠ de 8000 ERR (XXX/72) HV LU = 0000 0000 0000 0000	
	Alarme ≠ 0 ERR (XXX/72) ALARME *	
73	Alarme 3 après XAR sur mémoire existante ERR (XXX/73) INSTRUCTION "XAR" NON EXECUTABLE	
74	SVC 6 sur XAR en mémoire existante ERR (XXX/75) INSTRUCTION "XAR" NON EXECUTABLE	

80	<p>Pas d'alarme réveil processeur sur IPI ERR (XXX/80) PAS DE REVEIL PROCESSEUR Sur alarme réveil processeur ≠ 8000 ERR (XXX/80) HV LU = 0000 0000 0000 0000 Alarme ≠ de 6 ERR (XXX/80) ALARME * Pas de remise à 0 de BOX 2 ERR (XXX/80) BOX x LU = 0000 0000 0000 0000</p>	Test de fonctionnement de BOX2 BX2
81	<p>IT réveil processeur malgré le masquage d' IPM ERR(XXX/81) REVEIL PROCESSEUR/IPM = 1 Remise à zéro de la boîte aux lettres sans IT ERR (XXX/81) BOX LU = 0000 0000 0000 0000</p>	
82	<p>Pas d'alarme réveil processeur sur IPI ERR(XXX/82) PAS DE REVEIL PROCESSEUR Sur alarme réveil processeur HV ≠ '8000 ERR (XXX/82) HV LU = 0000 0000 0000 0000 Alarme ≠ de 6 ERR(XXX/82) ALARME * Pas de remise à zéro de BOX 2 ERR(XXX/82) BOX LU = 0000 0000 0000 0000</p>	
84	<p>Remise à zéro sur IPI masque ERR(XXX/84) BOX LU = 0000 0000 0000 0000</p>	
85	<p>Pas de remise à zéro de la boîte aux lettres ERR(XXX/85) BOX LU = 0000 0000 0000 0000</p>	
86	<p>Pas d'alarme réveil processeur sur IPI ERR (XXX/86) PAS DE REVEIL PROCESSEUR ERR(XXX/86) PAS D'II/IPI EN TACHE HARD/IMP/1 ERR(XXX/86) IT/IPI EN TACHE HARD/IPM = 0</p>	
87	<p>Pas d'alarme réveil processeur sur IPI ERR(XXX/87) PAS DE REVEIL PROCESSEUR Pas de remise à zéro de BOX 2 ERR(XXX/87) BOX LU = 0000 0000 0000 0000</p>	