

**68040**

Pierre Philippe Delacroix

Copyright © ,CopyrightÂ©1996 Editions A.D.F.I., Tous Droits Réservés

---

**COLLABORATORS**

	<i>TITLE :</i> 68040		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Pierre Philippe Delacroix	August 19, 2022	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>68040</b>	<b>1</b>
1.1	Les instructions du processeur 68040 . . . . .	1
1.2	Invalidation des lignes de cache . . . . .	2
1.3	Vidage des lignes de cache . . . . .	3
1.4	Calcul de valeur absolue . . . . .	3
1.5	Addition en virgule flottante . . . . .	3
1.6	Branchement selon les codes de conditions mathématiques . . . . .	4
1.7	Comparaison de deux nombres en virgule flottante . . . . .	4
1.8	Test mathématique, décrémentation et branchement . . . . .	5
1.9	Division en virgule flottante . . . . .	5
1.10	Transfert d'un nombre en virgule flottante . . . . .	5
1.11	Transfert de registres multiples . . . . .	6
1.12	Multiplication en virgule flottante . . . . .	6
1.13	Négation de l'opérande . . . . .	7
1.14	Pas d'opération . . . . .	7
1.15	Restauration de l'état du processeur . . . . .	8
1.16	Sauvegarde de l'état du processeur . . . . .	8
1.17	Positionnement selon les conditions . . . . .	8
1.18	Calculer la racine carrée . . . . .	8
1.19	Soustraction en virgule flottante . . . . .	8
1.20	Exception selon condition . . . . .	8
1.21	Test d'un nombre en virgule flottante . . . . .	8
1.22	Copie d'un bloc . . . . .	8
1.23	Vider le cache d'adresses logiques . . . . .	9
1.24	Tester une adresse logique . . . . .	9
1.25	Transfert d'un registre de contrôle . . . . .	9
1.26	Ceci n'est qu'une démonstration . . . . .	9

# Chapter 1

## 68040

### 1.1 Les instructions du processeur 68040

Le 68040 est à quelques exceptions près compatible avec le 68030, ←  
ici ne  
sont détaillées que les nouvelles instructions ou les instructions ayant  
subi des modifications. Voyez les explications sur le 68000 , 68010 ,  
68020 et 68030 pour une description complète.

Le micro-processeur 68040 contient deux mémoires cache de 4096 octets  
chacune, ce qui accélère ses performances en réduisant le nombre d'accès à  
la mémoire externe. Le bus est ainsi davantage disponible pour d'autres  
processeurs.

Le processeur mathématique intégré au 68040 ne reconnaît pas toutes les  
instructions des 68881 et 68882 . Sur Amiga, la bibliothèque  
68040.library est chargée d'émuler les instructions manquantes de manière  
transparente pour le programmeur, si bien qu'on peut considérer que le  
68040 supporte le jeu d'instructions des 6888x au complet.

Les instructions marquées \* sont privilégiées. Elles déclenchent donc  
l'exception "Violation de privilège" (n\textdegree{}8) si elles sont exécutées en ←  
mode  
utilisateur.

CINV  
\*Invalidation des lignes de cache.

CPUSH  
\*Vidage des lignes de cache.

FABS  
Calcul de la valeur absolue de l'opérande.

FADD  
Addition de deux opérandes.

FBcc  
Branchement selon les codes de conditions mathématiques.

---

FCMP  
Comparaison de deux nombres en virgule flottante.

FDBcc  
Test mathématique, décrémentation et branchement.

FDIV  
Division en virgule flottante.

FMOVE  
Transfert d'un nombre en virgule flottante.

FMOVEM  
Transfert de registres multiples.

FMUL  
Multiplication en virgule flottante.

FNEG  
Négation en virgule flottante.

FNOP  
Pas d'opération.

La suite ...  
Information indisponible dans cette démonstration.

Codes de conditions mathématiques  
L'adressage de la mémoire  
Branchements sur Amiga .

## 1.2 Invalidation des lignes de cache

CINVL  
CINVP  
CINVA

### DESCRIPTION:

Cette instruction marque les lignes de cache indiquées comme invalides.

**ATTENTION:** L'utilisation de cette instruction sous-entend une parfaite connaissance du fonctionnement des caches du 68040. C'est une instruction privilégiée.

### SYNTAXE:

CINVL <cache>, (An)  
CINVP <cache>, (An)  
CINVA <cache>

TAILLE: aucune.

---

### 1.3 Vidage des lignes de cache

CPUSHL  
CPUSHP  
CPUSHA

DESCRIPTION:

Actualise en mémoire les lignes de cache invalides spécifiées, puis marque toutes les lignes de caches indiquées comme étant invalides.

ATTENTION: L'utilisation de cette instruction sous-entend une parfaite connaissance du fonctionnement des caches du 68040. C'est une instruction privilégiée.

SYNTAXE:

CPUSHL <cache>, (An)  
CPUSHP <cache>, (An)  
CPUSHA <cache>

TAILLE: aucune.

### 1.4 Calcul de valeur absolue

FABS  
FSABS  
FDABS

DESCRIPTION:

Cette instruction calcule la valeur absolue de l'opérande source. Si un registre à virgule flottante est indiqué en destination, le résultat est placé dans ce registre.

FABS utilise la précision par défaut pour les arrondis, alors que FSABS utilise le mode simple précision et FDABS le mode double précision.

SYNTAXE:

FABS <EA>, FPn  
FABS FPm, FPn  
FABS FPn

TAILLES:

1ère syntaxe: octet, mot, mot long, simple, double, étendu, compressé.  
2ème syntaxe: étendu.  
3ème syntaxe: étendu.

### 1.5 Addition en virgule flottante

---

FADD  
FSADD  
FDADD

DESCRIPTION:

Cette instruction calcule la somme des deux opérandes signés. FADD utilise la précision par défaut pour les arrondis, alors que FSADD utilise le mode simple précision et FDADD le mode double précision.

SYNTAXE:

FADD <EA>,FPn  
FADD FPm,FPn

TAILLES:

1ère syntaxe: octet, mot, mot long, simple, double, étendu, compressé.  
2nde syntaxe: étendu.

## 1.6 Branchement selon les codes de conditions mathématiques

FBcc

DESCRIPTION:

Cette instruction teste la condition cc, portant sur l'état de l'unité de calcul en virgule flottante. Si elle est vraie, un branchement est réalisé à l'étiquette.

Voir l'avertissement au sujet des branchements relatifs.

SYNTAXE:

FBcc Étiquette

TAILLES: mot, mot long.

## 1.7 Comparaison de deux nombres en virgule flottante

FCMP

DESCRIPTION:

Cette instruction compare les deux opérandes signées.

SYNTAXE:

FCMP <EA>,FPn  
FCMP FPm,FPn

TAILLES:

---



1ère syntaxe: octet, mot, mot long, simple, double, étendu, compressé.  
2nde syntaxe: étendu.

## 1.8 Test mathématique, décrémentation et branchement

FDBcc

DESCRIPTION:

Cette instruction permet de réaliser des boucles :

- La condition cc est testée.
- Si elle est vraie, aucune opération n'est réalisée.
- Si elle est fausse, le mot de poids faible du registre Dn est décrémentée. Si le résultat est différent de -1, il y a branchement à l'étiquette.

Voir l'avertissement au sujet des branchements relatifs.

ATTENTION: Cette instruction n'existe pas sur 68060, elle doit être émulée par un programme approprié.

SYNTAXE:

FDBcc Étiquette

TAILLE: mot.

## 1.9 Division en virgule flottante

FDIV  
FSDIV  
FDDIV

DESCRIPTION:

Ces instructions divisent l'opérande destination par l'opérande source. Le résultat est placé dans le registre de destination. FDIV utilise la précision par défaut pour les arrondis, alors que FSDIV utilise le mode simple précision et FDDIV le mode double précision.

SYNTAXE:

FDIV <EA>,FPn  
FDIV FPM,FPn

TAILLES:

1ère syntaxe: octet, mot, mot long, simple, double, étendu, compressé.  
2nde syntaxe: étendu.

## 1.10 Transfert d'un nombre en virgule flottante

---

FMOVE

DESCRIPTION:

Transfère l'opérande source dans l'opérande destination. Si ce sont des registres en virgule flottante, la précision étendue (X) est utilisée.

SYNTAXE:

```
FMOVE <AE>,FPn
FMOVE FPm,<AE>
FMOVE.P FPm,<AE>{Dn}
FMOVE.P FPm,<AE>{#k}
FSMOVE <AE>,FPn
FDMOVE <AE>,FPn
FMOVE.L <AE>,FPcr
FMOVE.L FPcr,<AE>
```

TAILLES:

1ère, 2ème, 4ème et 5ème syntaxes: octet, mot, mot long, simple, double, étendu, compressé.

Autres: étendu.

## 1.11 Transfert de registres multiples

FMOVEM

DESCRIPTION:

Cette instruction copie les registres indiqués depuis ou vers l'adresse effective. Ces registres peuvent être soit des registres à virgule flottante, soit l'un des trois registres de contrôle (FPCD, FPSR et FPIAR). Si le masque des registres à copier se trouve dans un registre de donnée, seuls les registres à virgule flottante peuvent être spécifiés.

SYNTAXE:

```
FMOVEM <AE>,<liste de registres>
FMOVEM <AE>,Dn
FMOVEM <liste de registres>,<AE>
FMOVEM Dn,<AE>
```

TAILLE: étendu.

## 1.12 Multiplication en virgule flottante

FMUL  
FSMUL  
FDMUL

DESCRIPTION:

---

Cette instruction multiplie les deux opérandes signées et place le résultat dans le registre de destination. FMUL utilise la précision par défaut pour les arrondis, alors que FSMUL utilise le mode simple précision et FDMUL le mode double précision.

SYNTAXE:

```
FMUL <EA>,FPn  
FMUL FPm,FPn
```

TAILLES:

1ère syntaxe: octet, mot, mot long, simple, double, étendu, compressé.  
2nde syntaxe: étendu.

## 1.13 Négation de l'opérande

```
FNEG  
FSNEG  
FDNEG
```

DESCRIPTION:

Cette instruction calcule l'opposée de l'opérande source. Si un registre à virgule flottante est indiqué en destination, le résultat y est placé. FNEG utilise la précision par défaut pour les arrondis, alors que FSNEG utilise le mode simple précision et FDNEG le mode double précision.

SYNTAXE:

```
FNEG <EA>,FPn  
FNEG FPm,FPn  
FNEG FPn
```

TAILLES:

1ère syntaxe: octet, mot, mot long, simple, double, étendu, compressé.  
2ème syntaxe: étendu.  
3ème syntaxe: étendu.

## 1.14 Pas d'opération

```
FNOP
```

DESCRIPTION:

Cette instruction demande au processeur de n'effectuer aucune opération.

SYNTAXE:

```
FNOP
```

TAILLE: aucune.

---

## 1.15 Restauration de l'état du processeur

La suite ...  
Information indisponible dans cette démonstration.

## 1.16 Sauvegarde de l'état du processeur

La suite ...  
Information indisponible dans cette démonstration.

## 1.17 Positionnement selon les conditions

La suite ...  
Information indisponible dans cette démonstration.

## 1.18 Calculer la racine carrée

La suite ...  
Information indisponible dans cette démonstration.

## 1.19 Soustraction en virgule flottante

La suite ...  
Information indisponible dans cette démonstration.

## 1.20 Exception selon condition

La suite ...  
Information indisponible dans cette démonstration.

## 1.21 Test d'un nombre en virgule flottante

La suite ...  
Information indisponible dans cette démonstration.

## 1.22 Copie d'un bloc

La suite ...  
Information indisponible dans cette démonstration.

---

## 1.23 Vider le cache d'adresses logiques

La suite ...

Information indisponible dans cette démonstration.

## 1.24 Tester une adresse logique

La suite ...

Information indisponible dans cette démonstration.

## 1.25 Transfert d'un registre de contrôle

La suite ...

Information indisponible dans cette démonstration.

## 1.26 Ceci n'est qu'une démonstration

Vous trouverez toutes ces informations et bien d'autres choses dans la version française exclusive des Editions A.D.F.I.

Editions A.D.F.I.  
résidence les cottages  
83 rue André Theuriet  
F-63000 Clermont Ferrand

Téléphone : 3304+ 73.93.77.31.

---