

Chapitre 2 : Jeu d'instructions du 6809

On peut classer les instructions du 6809 en six catégories principales :

- Instructions de transfert de données
- Instructions de traitement de données
- Instructions de pointeurs de données
- Instructions de tests et branchements
- Instructions d'entrées-sorties
- Instructions de commandes

1. - Instructions de transfert de données

Les instructions de transfert de données transmettent des données entre :

- les registres internes du 6809;
- les registres internes et la mémoire;
- les pointeurs.

1.1. Instructions de transfert sur les registres internes

Les instructions **EXG** et **TFR** permettent le transfert de données entre les registres internes.

Le transfert ne peut se faire que sur des registres de même taille.

EXG R1,R2 Les contenus des registres R1 et R2 sont échangés. Exemple : EXG A,DP

TFR R1,R2 Le contenu du registre R1 est transféré dans le registre R2.

Exemple : TFR D,X

1.2 Instructions de transfert sur les registres internes et la mémoire

Les instructions **LD (LOAD : charger)** et **ST (Store : stocker)** permettent le transfert de données entre les registres internes et la mémoire.

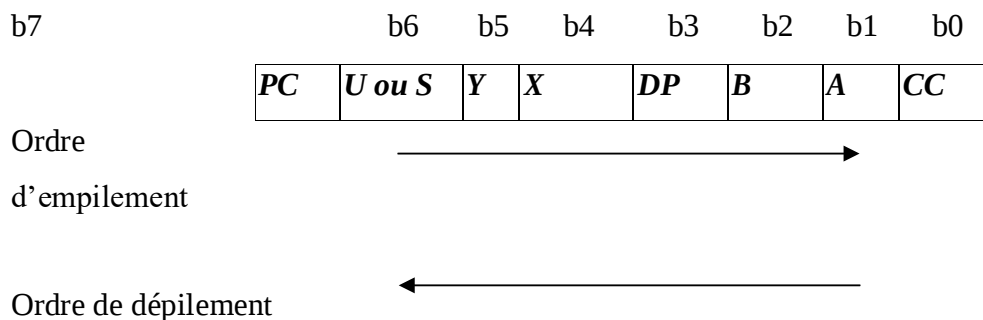
Instruction	Fonction
LD	la donnée en mémoire est chargée dans le registre spécifiée
ST	Le contenu du registre interne spécifié dans l'instruction est transféré dans l'emplacement mémoire indiqué.

Exemple :

LDA \$1000 charger l'accumulateur A avec le contenu de \$1000
 STA \$2000 Stocker le contenu de l'Acc. A à l'adresse mémoire \$2000

1.3 Instructions de transfert sur les pointeurs

Les instructions **PUSH (empiler)** et **PULL (dépiler)** permettent le transfert des registres internes dans la pile en mémoire. Les registres à **empiler/dépiller** sont indiqués dans l'octet (post octet) qui suit immédiatement le code opération de l'instruction **PUSH/ PULL**. Chaque bit du post octet indique un registre interne



Quand un de ces bits vaut 1, le registre correspondant est empilé/dépilé. Le pointeur de pile spécifié dans l'instruction ne peut être empilé/dépilé. Chaque fois qu'un octet est mis dans une pile, le pointeur de pile est décrémenté de 1. Pour un registre de 16 bits, on empile d'abord l'octet de poids faible dans la pile. Le dépilement est identique excepté qu'il incrémente le pointeur de pile.

Instruction	Fonction
PUL	Dépilement de(s) registre(s) de la pile.
PSH	Empilement de(s) registre(s) dans la pile.

Exemple :

PSHS #\$FF

<i>ssss - C ssss - B ssss -A ssss -9</i>	<i>CC</i>
<i>ssss - 8</i>	<i>A</i>
<i>ssss - 7</i>	<i>B</i>
<i>ssss - 6</i>	<i>DP</i>
	<i>X-haut</i>
<i>ssss - 5</i>	<i>X-bas</i>
<i>ssss - 4</i>	<i>Y-haut</i>
	<i>Y-bas</i>
<i>ssss - 3</i>	<i>U-haut</i>
<i>ssss - 2</i>	<i>U-bas</i>
<i>ssss - 1</i>	<i>PC-haut</i>
<i>ssss</i>	<i>PC-bas</i>

2. - Instructions de traitement de données

Les instructions de traitement de données peuvent être classées en quatre catégories :

- les instructions arithmétiques;
- les instructions logiques;
- les instructions de déplacements (rotation et de décalage);
- Les instructions d'incrément/décément, mise à zéro, complémentation

2.1. Instructions arithmétiques

Les instructions arithmétiques sont listées dans le tableau suivant :

Instruction	Fonction
ADD	Addition du contenu mémoire à un accumulateur
ADC	Addition du contenu mémoire à un accumulateur avec retenue
ABX	Addition de l'accumulateur B à X
DAA	Ajustement décimal de l'acc. A
MUL	Multiplication de A par B (non signée)
SUB	Soustraction du contenu mémoire à un accumulateur
SBC	Soustraction du contenu mémoire à un accumulateur avec retenue
SEX	Extension de signe de l'accu. B à l'acc. A

2.2. Instructions logiques

Les instructions logiques sont listées dans le tableau suivant :

Instruction	Fonction
AND	'ET logique' entre mémoire et registre interne
EOR	'XOR' entre mémoire et registre interne
OR	'OU logique' entre mémoire et registre interne

L'instruction AND est utile pour mettre à 0 ou masquer un ou plusieurs bits dans un mot. L'instruction XOR est utile pour des comparaisons. On peut l'utiliser aussi pour complémenter un mot (EORA #\$FF).

2.3. Instructions de déplacements (rotation et de décalage)

Il faut tout d'abord distinguer entre une opération de décalage et une opération de rotation.

Dans une opération de décalage, tous les bits sont décalés d'une position vers la droite ou vers la gauche. Le bit qui sort du registre va dans le bit de retenue C ; le bit qui entre est un zéro.

Dans une opération de rotation, le bit entrant dans le registre est celui qui provient de la retenue C.

Instruction	Fonction
ASR	Décalage arithmétique à droite. Les bits sont décalés vers la droite. b_0 est transféré vers C et b_7 reste inchangé
LSL ou ASL	Décalage logique ou arithmétique à gauche. Les bits sont décalés vers la gauche. b_7 est transféré vers C et b_0 est mis à 0.
LSR	Décalage logique à droite. Les bits sont décalés vers la droite. b_0 est transféré vers C et b_7 est mis à 0.
ROL	Rotation à gauche Les bits subissent une rotation vers la gauche. b_7 est transféré vers C et la valeur d'origine de celui-ci est transférée vers b_0 .
ROR	Rotation à droite ² Les bits subissent une rotation vers la droite. b_0 est transféré vers C et la valeur d'origine de celui-ci est transférée vers b_7 .

Exemple :

RORA ; ROLB ; LSRA ; LSRB

2.4 Instructions d'incrément/décément, Mise à zéro, Complémentation

Ces instructions sont listées dans le tableau suivant :

Instruction	Fonction
CLR	Remise à 0 du contenu mémoire ou de l'accumulateur
DEC	Décémentation du contenu mémoire ou de l'accumulateur
INC	Incrément du contenu mémoire ou de l'accumulateur
COM	Complément à 1 du contenu mémoire ou de l'accumulateur
NEG	Complément à 2 du contenu mémoire ou de l'accumulateur
NOP	Pas d'opération. Incrément du compteur programme

3. Instructions de tests et branchements

Les instructions de tests et branchements peuvent être classées en trois catégories :

- les instructions de tests et de comparaison;
- les instructions de tests et branchements;
- les instructions de test et branchement

3.1 Instructions de tests et de comparaison

Ces instructions sont utilisées pour réaliser des tests de bits d'état et des comparaisons afin de pouvoir prendre des décisions au cours du déroulement d'un programme dépendant de la valeur de ces indicateurs. Seul le registre d'état est modifié ; ni le registre spécifié ni l'opérande en mémoire ne sont changés. Aucun branchement n'est effectué.

Ces instructions sont listées dans le tableau suivant :

Instruction	Fonction
BIT	Test de bits BITA M ; BITB M L'acc. (A ou B) spécifié et l'opérande en mémoire font l'objet d'un ET logique.
CMP	Comparaison d'un contenu mémoire avec un accumulateur CMPA ; CMPB ACCX – M
TST	Test du contenu mémoire ou d'un accumulateur TSTA ; TSTB ; TST M ACCX – 0 ; M – 0

3.2 Instructions de test et branchement

Ces instructions effectuent des tests sur 4 indicateurs du registre d'état (C, N, Z, V) afin de réaliser des branchements au cours du déroulement d'un programme dépendant de la valeur de ces indicateurs. Deux types de branchement existent :

- Un branchement court : déplacement entre –128 et +127
- Un branchement long : déplacement entre –32768 et +32767.

Les instructions utilisant un branchement long ont une lettre 'L' qui précède leurs mnémoniques.

Ces instructions sont listées dans le tableau suivant :

Instruction	Fonction
BCC ou BHS	Branchement si pas de retenue BCC N ; LBCC NN Si C=0, alors : PC ← PC + N (ou) PC ← PC + NN
BCS ou BLO	Branchement si retenue BCS N ; LBCS NN
BEQ	Branchement si égal à zéro
BNE	Branchement si différent de zéro
BGE	Branchement si supérieur ou égal à zéro (signé) Si (N⊕V)=0 alors : PC ← PC + N (ou) PC ← PC + NN
BLT	Branchement si inférieur (signé)
BGT	Branchement si supérieur (signé)

3 BLE	Branchement si inférieur ou égal (signé)
BHI	Branchement si supérieur (non signé)
BLS	Branchement si inférieur ou égal (non signé)
BMI	Branchement si négatif
BPL	Branchement si positif
BVC	Branchement si pas de débordement
BVS	Branchement si débordement

4. instructions de traitement des interruptions

Le tableau suivant illustre ces instructions :

Instruction	Fonction
CWAI	Mise à zéro d'indicateurs d'état et attente d'interruption CWAI #N CC = CC and N. Ceci peut mettre à 0 certains bits donc démasquer les interruptions.
SYNC	Synchronisation avec une ligne d'interruption SYNC
RTI	Retour de sous-programme d'interruption RTI
SWI/SWI2/SWI3	Interruptions logicielles L'état complet est sauvegardé dans la pile système.

