

## Modes de fonctionnement et utilisation du clavier de programmation PCA2.P10

### Sélecteur des modes de fonctionnement

**RUN**  
**BREAK**  
**STEP**  
**MAN**  
**LCM**  
**PROG**

Les entrées et sorties sont actives

PCA2 seulement

### Touches du clavier de programmation PCA2.P10

7	8	9	A
4	5	6	+
1	2	3	-
O	L	E	C

Adresse

Adresse +1

Adresse -1

Clear (effacer)

Enter (introduire)

LCM (charger la mémoire de recopie)

### RUN

#### Déroulement normal du programme

Si le clavier de programmation est débranché le PCA2 reste toujours dans le mode RUN, quel que soit la position du sélecteur. (Contrairement au PCA1)

### BREAK

#### Interruption du déroulement du programme suivi d'une progression pas à pas

- ☐ la dernière ligne d'instruction traitée est affichée
- ☐ ☐ ... les lignes d'instructions suivantes sont effectuées en pas à pas

Le déroulement du programme en mode de fonctionnement BREAK est à tous points de vue analogue à celui de mode de fonctionnement RUN, c'est-à-dire que les programmes parallèles sont effectués parallèlement.

#### Positionnement d'un «Breakpoint»

- ☐ 820 ☐ le programme se déroule automatiquement (environ dix fois plus lentement que dans le mode RUN) jusqu'à l'adresse de pas 820 et puis s'arrête
- ☐ ☐ ... les lignes d'instructions suivantes sont effectuées en pas à pas

Toute la ligne d'instruction et l'état de l'accumulateur est affiché.

En remplaçant le sélecteur des modes de fonctionnement sur RUN, le programme continue à se dérouler là où il est arrêté.

### STEP

#### Saut à une adresse présélectionnée

- ☐ la dernière ligne d'instruction traitée est affichée
- ☐ 139 ☐ saut direct sur l'adresse de pas 139
- ☐ ☐ ... les lignes d'instructions suivantes sont effectuées en pas à pas

Toute la ligne d'instruction et l'état de l'accumulateur est affiché.

En effectuant un saut sur l'adresse de début d'un programme parallèle il est possible de faire que seul ce programme parallèle se déroule.

En remplaçant le sélecteur des modes de fonctionnement sur RUN, seul ce programme parallèle continue à se dérouler là où il est arrêté.

### MAN

#### Scrutation manuelle et positionnement / repositionnement de l'état logique d'un élément

(éléments = entrées, sorties, indicateurs volatils et non-volatils, compteurs et temporisateurs)

scrutation:  
☐ 32 scruter l'élément 32. Dans le dernier chiffre de l'opérand, 1 ou 0 est indiqué (High ou Low)

positionner / repositionner:

- ☐ libère l'introduction de l'état logique
- ☐ / ☐ positionne l'élément sur High ou Low

### LCM

#### Charger la mémoire de recopie (Cette fonction n'est possible qu'avec la série PCA2.)

- STEP**  
☐ XXXX introduction de l'adresse de début du programme à transférer
- CODE OPERAND**  
☐ 00 XXXX introduction de l'adresse finale du progr.
- ☐ lancement du processus de transfert

Transférer c'est-à-dire recopier un programme de RAM sur EPROM, EPROM sur EPROM, EPROM sur RAM, RAM sur RAM (mémoire de recopie enfichée en position inférieure).

Le transfert du programme s'effectue à une vitesse de 1K/100s et peut être suivi sur l'affichage du clavier.

### PROG

#### Programmer

- STEP**  
☐ XXXX introduction de l'adresse de début. La touche ☐ positionne le registre d'adresse sur 0.
- CODE OPERAND**  
☐ XX XXXX la touche ☐ positionne le registre «code» et le registre «opérand» à 0. Un contenu de la mémoire sera effacé.

Introduction d'une instruction en code numérique ainsi que de l'opérand

☐ effacement d'une introduction fausse.

☐ ☐ resp. ☐ ☐ contrôle d'un programme.

L'introduction ou la correction d'un programme ne sont possible que sur une mémoire RAM (mémoire utilisateur).



# Apperçu des instructions Niveau 1

Adresse de pas  
**STEP**

Instruction en code  
numérique  
**CODE**

Eléments ou adresses de saut  
**OPERAND**

Mémoire pour  
le résultat  
de combinaisons  
**ACCU = 1**



Affichage sur le  
clavier de  
programmation  
PCA2.P10

0 1 3 9

0 1

0 3 5 6

## Adresses des éléments

Entrées et sorties 0...255  
Temporisateurs et compteurs 256...287  
Indicateurs, volatil 288...764  
Indicateurs, non-volatil 765...999

Domaine des  
adresses:  
AE = adresse des  
éléments  
AP = adresse de pas  
PP = numéro du  
programme  
parallèle  
OP = opérand  
quelconque  
I = valeur de  
l'indexe  
(i) = indexable

## Effet concernant l'ACCU

seulement exécuté  
si l'ACCU est  
instruction positionne  
l'ACCU sur

### Instructions de combinaison logique



### Instructions de commutation



### Instructions de temporisation et de comptage

### Instructions de saut



### Instruct. d'attente

### Instructions auxili- aires



Code numé- rique	Code mnémo- nique 2e ligne	Instruction en anglais	Description				
01 02	STH STL	Start High Start Low	{ Début d'une combinaison avec scrutation de l'état de l'élément }	High Low	AE = 0...999 (i)		1/0
03 04	ANH ANL	And High And Low	{ Combinaison ET: ACCU avec l'état de l'élément }	High Low	AE = 0...999 (i)		1/0
05 06	ORH ORL	Or High Or Low	{ Combinaison OU: ACCU avec l'état de l'élément }	High Low	AE = 0...999 (i)		1/0
07	XOR	Exclusive Or	Combinaison OU exclusif: comparaison de l'ACCU avec l'état de l'élément adressé		AE = 0...999 (i)		1/0
08	NEG	Negate Accu	Inversion du contenu de l'ACCU		0		0/1
09	DYN	Dynamic Control	Déclenchement par le front d'un signal ou dynamisation d'une combinaison		AE = 288...999 (i)		1/0
10	OUT	Set Output with Status of Accu	Positionnement d'une sortie ou d'un indicateur selon l'état de l'ACCU		AE = 0...255, 288...999 (i)	1/0	
11 12	SEO REO	Set Output Reset Output	Positionnement } d'une sortie ou d'un Repositionnement } indicateur avec mémorisation		AE = 0...255, 288...999 (i)	1	
13	COO	Complement Output	Scrutation de l'état d'une sortie ou d'un indicateur et positionnement inverse		AE = 0...255, 288...999 (i)	1	
14	STR (00')	Set Timer -	Positionnement du temporisateur avec la valeur présélectionnée et départ de la temporisation Valeur de la temporisation en 1/10 s (resp. 1/100 s)		AE = 256...287 (i) 0...2047	1	
15	SCR (00')	Set Counter -	Positionnement du compteur avec la valeur présélectionnée. Valeur de comptage 1) pour des valeurs > 2047 voir le manuel de base C 7.3.		AE = 256...287 (i) 0...2047	1	
	16 17 18	- - -	Introduction externe de valeurs BCD: Code pour une valeur 2 x 4 Bit BCD x1 Code pour une valeur 2 x 4 Bit BCD x10 Code pour une valeur 2 x 4 Bit BCD x100		AE = 7...255, 295...999		
17 18	INC DEC	Increment Counter Decrement Counter	Augmente } la contenu du compteur de 1 Diminue }		AE = 256...287 (i)	1	
20	JMP	Unconditional Jump	Saut inconditionnel à une adresse de pas		AP = 0...2047		1
21 22	JIO JIZ	Jump if Accu is One Jump if Accu is Zero	{ Saut à une adresse de pas si } ACCU = 1 ACCU = 0		AP = 0...2047	1 0	1 1
23 24	JMS RET	Jump to Subroutine Return from Subroutine	Saut dans un sous-programme Retour d'un sous-programme		AP = 0...2047 0		1 1
25 26	WIH WIL	Wait if High Wait if Low	{ Attend aussi longtemps que } High l'état de l'élément est } Low		AE = 0...999 (i)		1 1
00	NOP	No Operation	Pas d'opération		0		
19	SEA	Set Accumulator = 1	Positionne l'ACCU = 1		0		1
16	SEI	Set Index	Positionne le registre d'index sur la valeur choisie (I0)		I0 = 0...255		1
27 28	INI DEI	Increment Index Decrement Index	Augmente } le registre d'index de 1 Diminue } jusqu'à la valeur finale (I)		I > I0 I < I0		1/0 1/0
29	PAS (00)	Program Assignment -	Assignment de programmes parallèles Adresse de pas de début du programme parallèle		PP = 0...15 AP = 0...2047		
30 31	DOP DTC	Display Operand Display Timer/Counter	Affichage du contenu d'un opérand choisi Affichage du contenu d'un temporisat. ou d'un compt.		OP = 0...2047 AE = 256...287 (i)	0 1	



# Automates programmables à mémoires NIVEAU 2

## Introduction de textes dans la mémoire de textes

1

### Programmation de l'interface comme éditeur

(avec le clavier de programmation P10, mode de fonctionnement PROG)

```

4001 PAS (29) 100
4002      00 898      paramètre (selon le périphérique)
4003      01 63       01 = activer l'éditeur pour l'introduction des textes
4004      00 254      un élément qui reste en permanence sur «LOW»
4005      00 254
4006      00 0
4007      00 0
4008      00 0
4009      00 0
4010      00 0
4011 PAS (29) 23
4012      00 TXn      TXn = numéro du texte
4013 JMP (20) 0
4014      00 4011     sortie de texte en mode de fonctionnement STEP
  
```

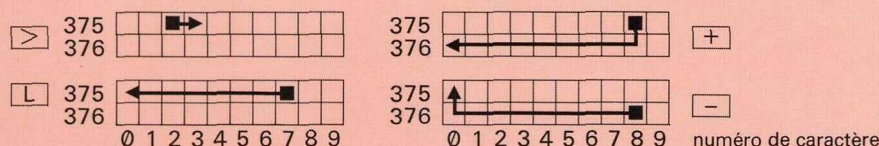
3

### Mode affichage de texte

(Affichage du contenu de la mémoire sur l'écran)

TXn [CR] la ligne de texte ayant le numéro de texte TXn sera affichée

Commande du curseur



2

### Activation de l'éditeur

En mode de fonctionnement STEP, à l'aide du clavier de programmation P10, sauter à l'adresse de pas 4001 et travailler jusqu'à 4011

4

### Mode introduction de texte

(avec appareil périphérique)

[CTRL] [T] Commutation au mode introduction d'un texte, à partir du numéro de caractère préselectionné avec le curseur en mode affichage. Introduction du texte suivant ainsi que des signes de fonctions.

[CTRL] [ @ ] \* Fin du texte  
 [ ^ ] [ CR ] Est compris comme «CR» (Carriage Return) et est mémorisé dans la mémoire de textes

[ CR ] Retour au mode affichage de textes

\* Le caractère ASCII NUL est dépendant de l'appareil périphérique

5

### Sortie du texte pour contrôle

En mode STEP, avec le clavier de programmation P10, actionner 2 fois la touche [ + ]. A chaque fois, le texte sera affiché pour contrôle.

## Signes de fonctions à introduire dans la mémoire de textes

Introduc- tion alpha- numérique	Introduction avec le P10	Description
\$C Cn* \$c Cn*	36, 67, Cn 36, 99, Cn	Sortie du contenu des registres temporisateur/compteur Cn 5 chiffres décimaux <u>sans</u> suppression des zéros du début 5 chiffres décimaux <u>avec</u> remplacement des zéros du début par des espaces
\$E En* \$e En*	36, 69, En 36, 101, En	Sortie de l'état de 8 éléments En...En-7 (8 caractères 1 ou 0) Sortie du caractère ASCII, formé sur les 8 éléments En...En-7
\$T \$H \$D	36, 84 36, 72 36, 68	<b>Sortie du contenu de l'horloge:</b> 14 : 04 / 18 : 50 : 42 semaine jour heure minute seconde (14.) (jeudi) 84 : 04 : 05 / 18 : 50 : 42 année mois jour heure minute seconde 84 : 04 : 05 année mois jour
\$ _ xxx*	36, _ , xxx	Sortie répétée du caractère suivant immédiatement le signe \$, aussi souvent que xxx l'indique: xxx ≤ 127: Nombre de répétitions correspondant à xxx xxx = 256...319: Nombre de répétitions correspondant au contenu du compteur adressé (max. 127) Tous les caractères ASCII (NUL également) peuvent être utilisés, excepté les suivants, qui sont réservés pour le déclenchement de fonctions: \$, A, C, D, E, H, L, R, T, U, a, c, e, r
\$L xxx*	36, 76, xxx	Saute au texte No xxx avec mémorisation de l'adresse de départ (numéro du caractère) Retour à l'adresse de départ +1
\$U	36, 85	
\$ \$	36, 36	\$ (Sortie du caractère Dollar)

\* Pour Cn, En resp. xxx, 3 caractères sont nécessaires. Par exemple 027.

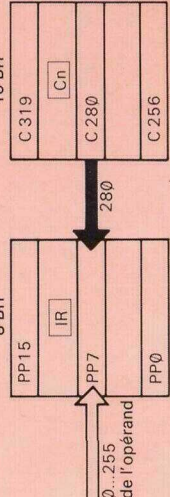
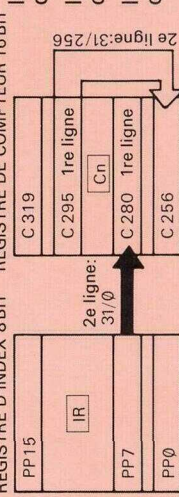
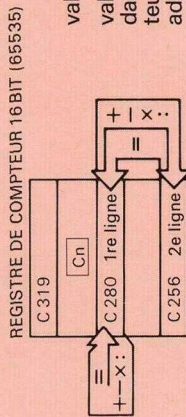
## ASCII CODE CHART

B7 BITS	B6 00	B5 00	00	01	10	11	10	11	11	11
	00	01	01	01	10	10	11	11	11	11
B4B3B2B1	CONTROL		SYMBOLS		UPPERCASE		LOWERCASE			
0000	NUL	DLE	SP	48	@	P	96	p	112	
0001	SOH	DC1	!	49	A	Q	97	q	113	
0010	STX	DC2	"	50	B	R	98	r	114	
0011	ETX	DC3	#	51	C	S	99	s	115	
0100	EOT	DC4	\$	52	D	T	100	t	116	
0101	ENQ	NAK	%	53	E	U	101	u	117	
0110	ACK	SYN	&	54	F	V	102	v	118	
0111	BEL	ETB	7	55	G	W	103	w	119	
1000	BS	CAN	(	56	H	X	104	x	120	
1001	HT	EM	9	57	I	Y	105	y	121	
1010	LF	SUB	*	58	J	Z	106	z	122	
1011	VT	ESC	+	59	K	[	107	k	123	
1100	FF	FS	,	60	L	\	108	l	124	
1101	CR	GS	-	61	M	]	109	m	125	
1110	SO	RS	.	62	N	^	110	n	126	
1111	SI	US	/	63	O	_	111	o	127	DEL

\* Dans la plupart des appareils périphériques, les caractères des cases «CONTROL» peuvent être obtenus par pression simultanée sur la touche [CTRL] et celle du caractère correspondant, situé dans «UPPERCASE».  
 Par exemple «FF» (Form Feed, No 12) est obtenu en actionnant simultanément [CTRL] et [L] (No 76).



Code chiffre	Code mnémon. 2e ligne	Description	Opérand (i) = indexable	Code chiffre	Code mnémon. 2e ligne	Opérand	Description
14	STR 00...31 <sup>1)</sup>	<b>INSTRUCTIONS DE TEMPORISATION ET DE COMPTAGE</b> Positionner le temporisateur adressé sur la valeur préselect. et départ de la temporisation Valeur du temps en 1/10 s (resp. 1/100 s)	256...287 (i)	29	PAS	16	<b>INSTRUCTIONS PAS</b> Exécution exclusive d'un programme parallèle (avec blocage du déroulement des temporisa- teurs, de l'horodateur et de la ligne série) 2e ligne toujours 0
15	SCR 00...31 <sup>1)</sup>	Positionner le compteur adressé sur la valeur préselectionnée Valeur du compteur	256...319 (i)	29	00 PAS 00	17 0	Fin de l'exécution exclusive 2e ligne toujours 0
	00 : 15	<sup>1)</sup> Valeurs 0...32 767 00 (≡ 0) 01 (≡ 2048) 02 (≡ 4096) 03 (≡ 6144) 04 (≡ 8192) 05 (≡ 10240) 06 (≡ 12288) 07 (≡ 14336) 08 (≡ 16384) 09 (≡ 18432) 10 (≡ 20480) 11 (≡ 22528) 12 (≡ 24576) 13 (≡ 26624) 14 (≡ 28672) 15 (≡ 30720)	+ valeur dans l'opérand 0...2047	29	PAS	18	Limitation du nombre de programmes parallèles travaillés 2e lig. indique le nombre de prog. parallèles actifs
	16 17 18 19 20 21 22 23 24 25 26	<sup>1)</sup> Ordre de lecture pour 2 fois 4 bits BCD x1 pour 2 fois 4 bits BCD x10 pour 2 fois 4 bits BCD x100 pour 5 fois 4 bits BCD x1 pour 8 bits binaires pour 12 bits binaires pour 16 bits binaires	7...999 (i) l'adresse la plus haute du groupe d'éléments (LSB = Least Significant Bit)	29	PAS 00	23 0...818	Instruction pour la transmission d'un texte 2e ligne indique le numéro du début du texte, la transmission s'effectuera jusqu'au signe «NUL» dans le texte
	27 28 29 30	<sup>1)</sup> Opérations arithmétiques addition soustraction multiplication division	1...255 256...319 (i)	29 29	PAS 00 PAS xx	30 0 31...38	Contrôle du programme système par «check-sum» Contrôle du programme utilisateurs par «check-sum» (31...38 ≡ 1er K...8eK) 2e ligne contient la somme de référence
	31	<sup>1)</sup> Echange de données entre les registres de compteur et le registre d'index	0 256...287 (i) 288...319 (i)	29	PAS xy	50 Cn	Echanges de données entre horodateur et registre de compteurs xy indique la fonction à effectuer: x: 0 ≡ écrire dans le registre de l'horodateur 1 ≡ lire le contenu du registre de l'horodateur y: 0 ≡ numéro de la semaine (valeur: 1...53) 1 ≡ jour de la semaine (valeur: 1...7) 2 ≡ année (valeur: 0...99) 3 ≡ mois (valeur: 1...12) 4 ≡ jour du mois (valeur: 1...31) 5 ≡ heures (valeur: 0...23) 6 ≡ minutes (valeur: 0...59) 7 ≡ secondes (valeur: 0...59) Cn indique le temporisateur/compteur contenant la valeur à inscrire dans le registre de l'horoda- teur resp. dans lequel la valeur lue dans le registre de l'horodateur doit être déposée
20/21 22/23	JMP/JIO JIZ/JMS 00 <sup>2)</sup>	<b>INSTRUCTIONS DE SAUT</b> <sup>2)</sup> les adresses de pas ≤ 2047 peuvent être introduites en une ou deux lignes les adresses de pas > 2047 doivent être introduites en deux lignes	0 0 0...819 <sup>2)</sup>	29	PAS	100	Instruction à 10 lignes définissant les paramètres de transmission de la ligne série et le mode T, E, C, N ou P
16	SEI	<b>INSTRUCTIONS AUXILIAIRES</b> le registre d'index est positionné à la valeur initiale se trouvant dans l'opérande (constante) le registre d'index est positionné à la valeur initiale contenue dans le temporisateur/compteur adressé	0...255 256...319	29	PAS 00	190 19 0	Instruction à 10 lignes pour gestion d'interrupt (réaction très rapide due au travail exclusif en tant que programmes parallèles 15) fin de gestion d'interrupt 2e ligne est toujours 0
27 28	INI DEI	la valeur finale pour le registre d'index se trouve dans l'opérand (constante) la valeur finale pour le registre d'index se trouve dans le temporisateur/ compteur adressé	0...255 256...319	29	PAS PAS	200 201 202 250 251	Instruction à 10 lignes contenant les paramètres pour effectuer jusqu'à 32 boucles de régulation PID Instruction à 10 lignes pour la définition et la gestion d'un registre à décalage ou rotation Instruction à 10 lignes pour la définition et la gestion d'un registre FIFO

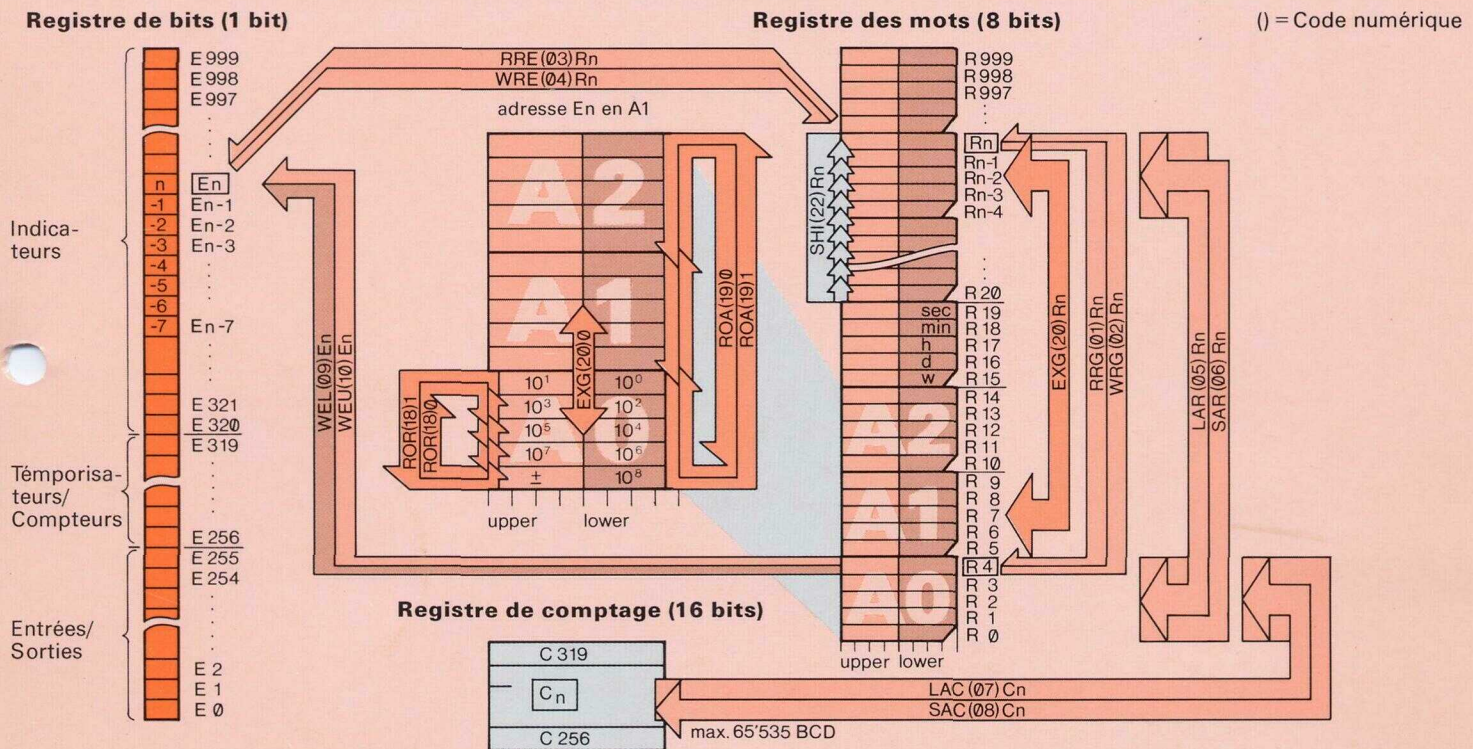




# Automates programmables à mémoires

NIVEAU ③

## Jeu d'instructions du processeur de mots



Code numérique	Code mnémotique	Operand	Instruction en anglais	Description	Format des données	
Instruction de transfert de données						
01	RRG	Rn	Read Register	Lire le mot en Rn et le mémoriser dans R4	8 bits	i
02	WRG	Rn	Write into Register	Ecrire en Rn le mot R4	8 bits	i
03	RRE	Rn	Read Register and write in Elements	Lire le mot en Rn et le mémoriser dans les 8 éléments En...En-7, adressés par A1	8 bits	i
04	WRE	Rn	Write Register with Elements	Ecrire en Rn le contenu des 8 éléments En...En-7 adressés par A1	8 bits	i
05	LAR	Rn	Load A0 with Registers	Chargement de A0 avec le bloc de registres Rn	5 x 8 bits	i
06	SAR	Rn	Store A0 into Registers	Mémorisation de A0 dans le bloc de registres Rn	5 x 8 bits	i
07	LAC	Cn	Load A0 with Counter	Chargement de A0 avec le compteur/temporisateur Cn	BCD	i
08	SAC	Cn	Store A0 into Counter	Mémorisation de A0 dans le compteur Cn	BCD	i
09	WEL	En	Write Elements with { Lower } Digit	Ecrire dans les éléments { R4 (10 <sup>0</sup> ) } En...En-3 le digit de { R4 (10 <sup>1</sup> ) }	4 bits	i
10	WEU	En				
Traitement des registres arithmétiques						
18	ROR	0/1	Rotate Register	Rotation R4...R3...R2...R1...R0 ou R0...R1...R2...R3...R4		
19	ROA	0/1	Rotate A	Rotation A0...A1...A2 ou A2...A1...A0		
20	EXG	0	Exchange A1 with A0	Echange A1 avec A0	5 x 8 bits	
		Rn	Exchange A1 with Rn...Rn-4	Echange A1 avec Rn...Rn-4	5 x 8 bits	i

i = indexable

i = indexable

### Signification des signes spéciaux (complémentaire au Niveau ②)

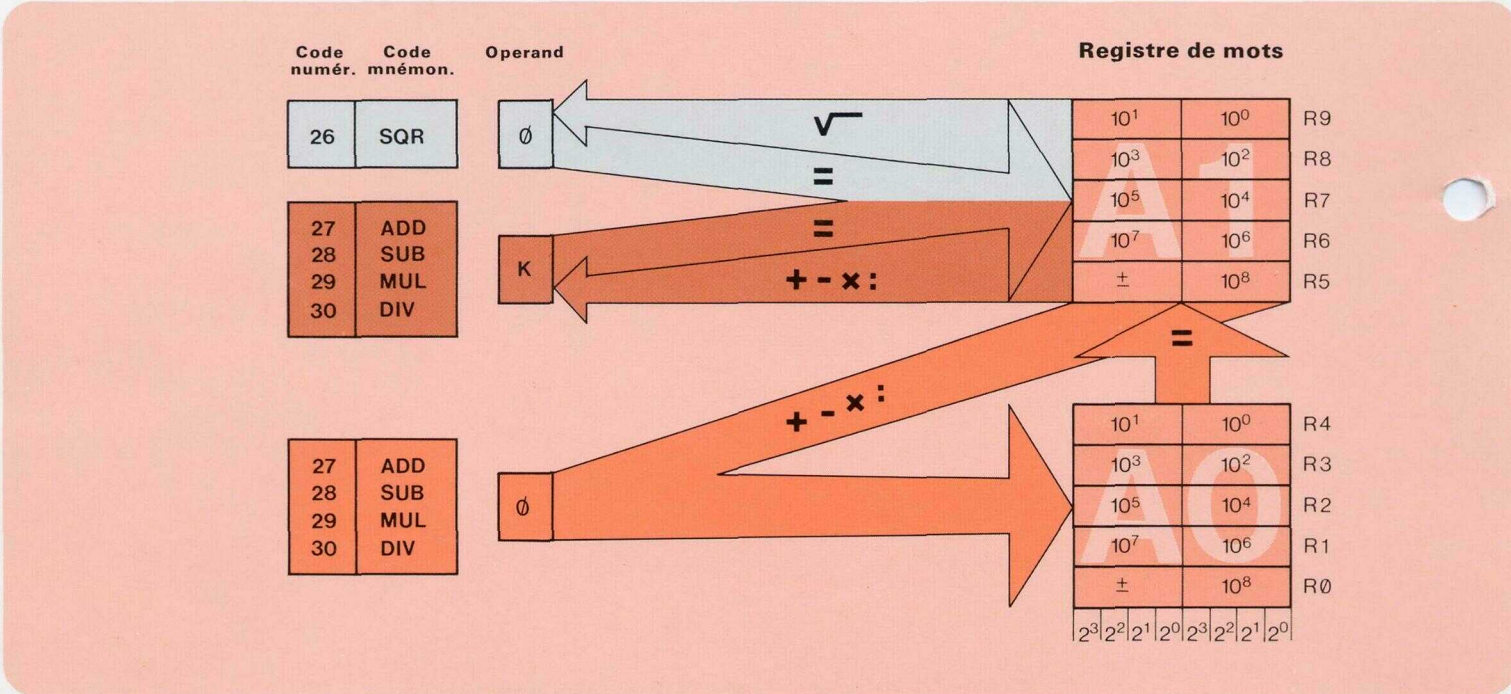
Introduction alphanumérique	Introduction avec le P10	Description
\$A Rn*	36, 65, ...	Contenu du bloc de registres Rn, 10 chiffres décimaux. Les zéros du début du nombre sont <u>affichés</u> .**
\$a Rn*	36, 97, ...	Contenu du bloc de registres Rn, 9 chiffres décimaux (si négatif avec signe). Les zéros du début du nombre sont <u>supprimés</u> (spaces).**
\$R Rn*	36, 82, ...	Contenu du registre Rn, 2 chiffres décimaux.**
\$r Rn*y	36, 114, ..., 0/1	Contenu de la partie LOWER ou UPPER du registre de mots, 1 chiffre décimal, y = 0 → LOWER, y = 1 → UPPER**

\* Pour Rn 3 caractères sont nécessaires, p. ex.: 027

\*\* Si des mots en format binaire atteignent des valeurs entre 10 et 15, les caractères suivants sont sortis: 10 → 11 → 12 → 13 → 14 → 15 →



Code numérique	Code mnémonique	Operand	Instruction en anglais	Description	Format des données	
<b>Instructions d'arithmétiques</b>						
25	CMP	En	Compare A0 with A1	Comparaison de A0 avec A1 $A1 > A0 \Rightarrow \text{En} = 1$ $A1 = A0 \Rightarrow \text{En} - 1 = 1$ $A1 < A0 \Rightarrow \text{En} - 2 = 1$	BCD	i
26	SQR	0	Square Root of A1	$\sqrt{A1} \Rightarrow A1$ , chiffres entiers seulement	BCD	
27	ADD	0 K	Add A0 to A1 Add K to A1	$A1 + A0 \Rightarrow A1$ , <carry> $A1 + K \Rightarrow A1$ , <carry> (K = 1...2047)	BCD BCD	
28	SUB	0 K	Subtract A0 from A1 Subtract K from A1	$A1 - A0 \Rightarrow A1$ , <carry> $A1 - K \Rightarrow A1$ , <carry> (K = 1...2047)	BCD BCD	
29	MUL	0 K	Multiply A1 by A0 Multiply A1 by K	$A1 \cdot A0 \Rightarrow A1$ , <carry> $A1 \cdot K \Rightarrow A1$ , <carry> (K = 1...2047)	BCD BCD	
30	DIV	0 K	Divide A1 by A0 Divide A1 by K	$A1 : A0 \Rightarrow A1$ , reste dans A0, <carry> $A1 : K \Rightarrow A1$ , reste dans A0, <carry> (K = 1...2047)	BCD BCD	



<b>Traitement des registres arithmétiques</b>						
14	CLA	0/1/2	Clear A	Effacement des registres A0 ou A1 ou A2		
15	LAI	K	Load A immediately	Si K ≤ 99 → chargement de R4 avec les données Si K = 100...2047 → chargement de A0 avec les données		
16	DBN	0	Decimal to binary	Conversion de A0 BCD en binaire		
17	BND	0	Binary to decimal	Conversion de A0 binaire en BCD		

<b>Instructions de comptage et de saut</b>						
11	INR	Rn	Increment Register	Augmentation de 1 la valeur BCD de Rn et mettre <carry>	> 99	i
12	DER	Rn	Decrement Register	Diminution au cas où le resultat est	< 0	i
13	SNC	0 En	Skip if no <carry> Skip if En = 0	Sauter l'instruction suivante si <carry> = 0 En = 0		i
24	SEW	0 En	Skip to EWP if no <carry> Skip to EWP if En = 0	Sauter toutes les instructions jusqu'à EWP ou NOP 1248 si <carry> = 0 En = 0		i

<b>Instructions diverses</b>						
00	NOP	1248	-	Changement du processeur de bits au processeur de mots et positionnement ACCU = 1		
00	NOP	0	No operation	Aucune opération		
21	CLK	En	Clock source	Assignation d'une source d'impulsion de temps		
22	SHI	Rn	Shift registers	Décalage des mots entre R20 et Rn d'une adresse vers le haut		i
23	TXT	Txn	Text	Sortie de textes		i
31	EWP	0	End Word Processor	Fin du travail dans le processeur de mots		