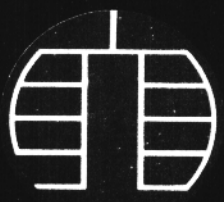


CP8

GUIDE UTILISATEUR DU M.C.S.

MODULE DE CONTROLE

ET DE SECURITE

Bull

Bull CP8

GUIDE UTILISATEUR DU M.C.S.

MODULE DE CONTROLE

ET DE SECURITE

Réf. BULL CP8 : TU 0066 F02
Juin 1987

SOMMAIRE

FIGURES

1. INTRODUCTION

1.1. PRESENTATION DU DOCUMENT

Le présent document est le GUIDE UTILISATEUR du M.C.S. ou Module de Contrôle et de Sécurité.

1.1.1. A QUI EST-IL DESTINE ?

Ce document est destiné à tout concepteur ou développeur d'application mettant en oeuvre les fonctions sécuritaires des cartes CP8 Masque 4/8/9, dans un environnement type réseau ou dans un équipement non connecté, type terminal de paiement.

Le lecteur est supposé connaître les principales caractéristiques des cartes CP8 Masque 4/8/9.

1.1.2. QUE CONTIENT-IL ?

La section 1.3 est un rappel des notions fondamentales de Personnalisation, Authentification, et Certification.

Le lecteur possédant correctement ces notions pourra se reporter directement à la section 2.

La section 2 est une présentation de la mémoire du M.C.S. ainsi que l'organisation et le format des informations qui y sont stockées.

La section 3 présente les aspects généraux du dialogue entre le M.C.S. et l'équipement dans lequel il est implanté.

Les sections 4 et 5 décrivent le format des commandes du M.C.S. ainsi que leurs conditions d'utilisation. Avec la section 2, ce sont les sections les plus importantes du document : une lecture attentive est garante de la bonne compréhension du M.C.S.

La section 6 présente des exemples de formatage de blocs pour un M.C.S. implanté dans un P.S.A., ainsi que des enchaînements de commandes.

Enfin, la section 7 est un guide pour la personnalisation du M.C.S., utilisable pour la préparation des "séances PAM".

1.1.3. BIBLIOGRAPHIE

Les documents référencés ci-dessous sont une présentation technique des produits BULL CP8 entrant dans le champs d'utilisation du M.C.S.

. Le guide utilisateur de la carte CP8 Masque 4 Réf. BULL CP8 TU 0047 F1
. Processeur de sécurité associé 8 cartes Réf. BULL CP8 TU 0033 FØ2
. Projet de norme ISO : Réf. ISO/DIS 7816/1

1.2. PRESENTATION GENERALE DU M.C.S.

1.2.1. CARACTERISTIQUES PHYSIQUES ET FONCTIONNELLES

Le M.C.S. est dérivé du composant MOTOROLA MC 68705P5, appartenant à la famille M 6805 des micro-calculateurs mono-chip. Il est implanté dans un boîtier DIL (Dual In Line).

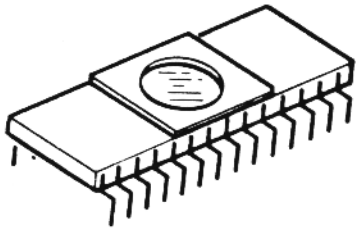


Figure 1.1 : Aspect physique du M.C.S.

Ce boîtier peut être intégré soit directement sur une carte électronique, soit sur l'adaptateur PBL 06 de BULL CP8, assurant la compatibilité physique avec les lecteurs encodeurs de cartes CP8 classiques.

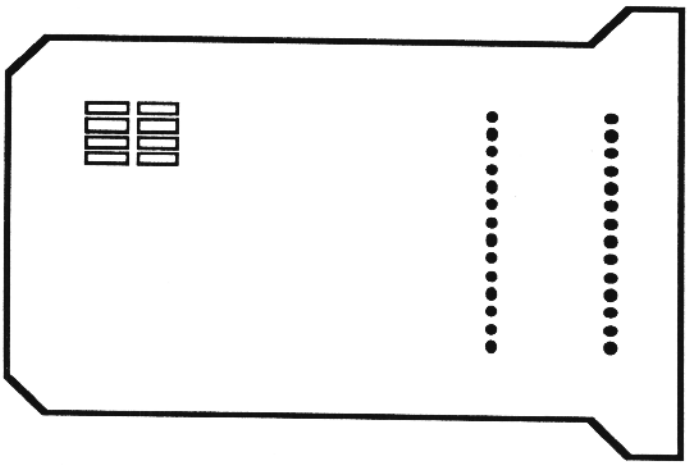


Figure 1.2 : L'adaptateur PDL 06

L'assignation fonctionnelle des broches du M.C.S., identique en son principe à celle d'une carte CP8, est la suivante :

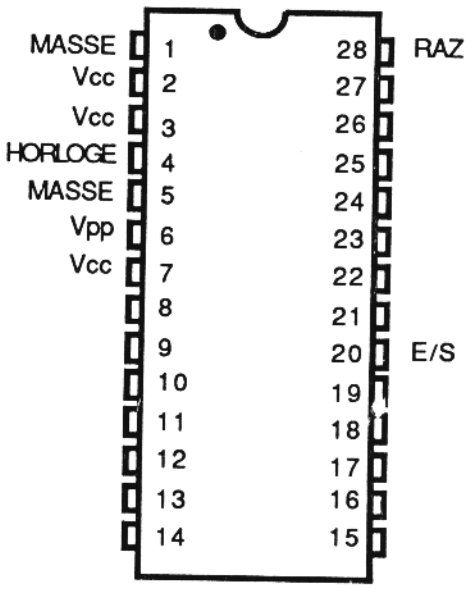


Figure 1.3 : Brochage du M.C.S

RAZ : Une tension appliquée sur cette broche déclenche l'initialisation du M.C.S. A l'issue de cette phase, le M.C.S. envoie un ensemble d'informations définissant ses caractéristiques techniques et "historiques".

HORLOGE : Une base de temps de 3,57 MHz doit être fournie au M.C.S. sur la broche 4.

Vcc : Sera appliquée sur cette broche la tension d'alimentation de base du M.C.S.

Vpp : L'écriture dans l'EPROM nécessitera l'application sur cette broche de la tension de programmation.

MASSE : Comme son nom l'indique.

E/S : Par cette broche transitent les données échangées entre le M.C.S. et le monde extérieur. Elles sont transférées à l'alternat, en mode asynchrone, à la vitesse de 9600 bauds, sous forme de caractères ayant le format suivant :
1 bit start, 8 bits de données, 1 bit de parité paire et 1 bit stop.

Le programme mémorisé dans l'EPROM du M.C.S. assure les échanges d'information entre le M.C.S. et le monde extérieur, conformément au projet de NORME ISO.

Il interprète et exécute un certain nombre d'ordres élémentaires, identifiés par la Classe d'instruction BC (identique à celle des cartes Masque 4/8/9).

On distingue deux types de M.C.S. :
- Le M.C.S. Masque 1, implantable dans un Terminal de paiement, appelé M.C.S. version C.T.U.
- Le M.C.S. Masque 2, implantable dans un P.S.A., appelé M.C.S. version HOST.
Les deux M.C.S. diffèrent sur certains ordres.

Enfin, le M.C.S. est capable d'exécuter la fonction algorithmique TELEPASS (commune à toutes les cartes CP8 Masque 4/8/9), mettant en oeuvre les jeux secrets inscrits définitivement dans sa mémoire EPROM ou recalculés et stockés provisoirement en RAM.

La taille mémoire utilisateur est de 64 mots de 32 bits chacun.

1.2.2. L'OBJECTIF DU M.C.S.

Le M.C.S. permet la réalisation des fonctions suivantes :
- Authentification,
- Certification,
- Calcul de clé de chiffrement,
- Calcul des clés-émetteurs primaires et secondaires (CLE 1A, CLE 1B).

Ce en garantissant l'inviolabilité et l'intégrité de la fonction Telepass, des jeux secrets mères et diversifiés.

Enfin, utilisé comme une carte mère, il peut être utilisé comme outil de diversification en phase de personnalisation.

Afin de bien saisir le sens de cet objectif, il est bon de rappeler les notions de personnalisation, authentification propres aux cartes CP8 Masque 4/8/9.

1.3. RAPPELS

1.3.1. LA PERSONNALISATION DES CARTES CP8 MASQUE 4/8/9

La PERSONNALISATION est la phase de vie d'une carte durant laquelle sont écrits :

- Les clés Emetteur Primaire et Secondaire, le Jeu Secret et le code confidentiel,
- Si nécessaire, les données applicatives des Zones Confidentielles et de Lecture,
- En Zone de Fabrication, les pointeurs définissant les tailles et implantation respectives de chaque Zone ainsi que les indicateurs définissant le type de protection en lecture/écriture de la Zone Transaction.

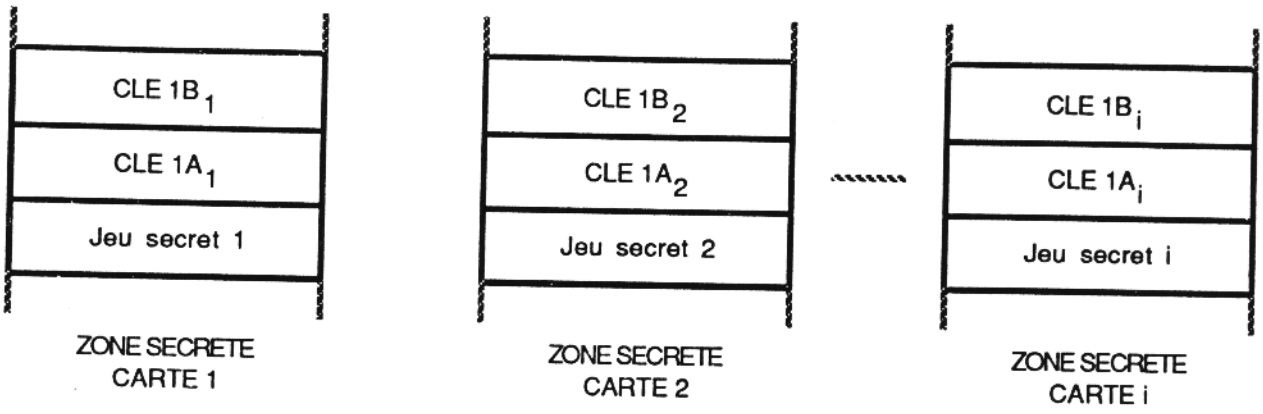
La personnalisation s'achève par le positionnement du lock LC à 0, dans le dernier mot de la mémoire.

Parmi toutes les informations écrites en phase de personnalisation, les Clés et Jeu Secret sont plus particulièrement impliqués dans les mécanismes de Diversification et d'Authentification.

1.3.2. NOTIONS DE DIVERSIFICATION DES CLES ET DU JEU SECRET

Supposons que nous ayons à personnaliser des cartes pour une application donnée.
Les Clés et Jeux Secrets sont des valeurs binaires de taille variable.

Comment définir leur valeur ?
La manière la plus simple serait d'attribuer arbitrairement à chaque carte la même Clé Emetteur Primaire (clé 1A), la même Clé Emetteur Secondaire (clé 1B), le même Jeu Secret, et pourquoi pas, le même code porteur (clé 2A).



$$\text{CLE 1B}_1 = \text{CLE 1B}_2 = \cdots = \text{CLE 1B}_i$$

$$\text{CLE 1A}_1 = \text{CLE 1A}_2 = \cdots = \text{CLE 1A}_i$$

$$\text{Jeu secret}_1 = \text{Jeu secret}_2 = \cdots = \text{Jeu secret}_i$$

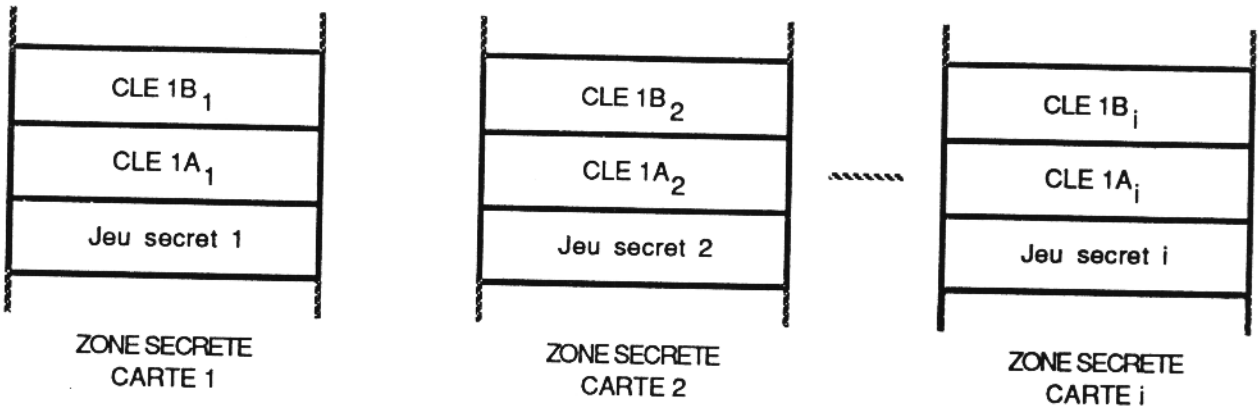
Figure 1.4 : Clés et Jeux Secrets non diversifiés

Le système est relativement fragile, non pas à cause de la carte elle-même, la Zone Secrète étant inviolable, mais un "espion" arrivant à capter la valeur d'une clé transitant dans un terminal (au moment de l'exécution de l'ordre de présentation de clé), connaîtrait par là-même, la valeur de cette clé pour toutes les cartes de l'application concernée.

La parade consiste à faire en sorte que chaque carte ait des Clés et Jeux Secrets différents.

Ainsi, on appelle DIVERSIFICATION, la méthode de génération des Clés et Jeux Secrets consistant à attribuer à chaque carte, des Clés et Jeux Secrets qui soient :

- spécifiques de la carte elle-même,
- et spécifiques à une application donnée.



$$CLE\ 1B_1 \neq CLE\ 1B_2 \neq \cdots \neq CLE\ 1B_i$$

$$CLE\ 1A_1 \neq CLE\ 1A_2 \neq \cdots \neq CLE\ 1A_i$$

$$Jeu\ secret_1 \neq Jeu\ secret_2 \neq \cdots \neq Jeu\ secret_i$$

Figure 1.5 : Clés et Jeux Secrets diversifiés

La méthode la plus "élégante" pour diversifier des Clés et Jeux Secrets en phase de personnalisation consiste à utiliser un outil fiable et puissant quant à ses capacités de diversification : l'algorithme TELEPASS.

Les cartes affectées à cette fonction sont appelées CARTES MERES.
Les cartes à personnaliser sont appelées CARTES FILLES.

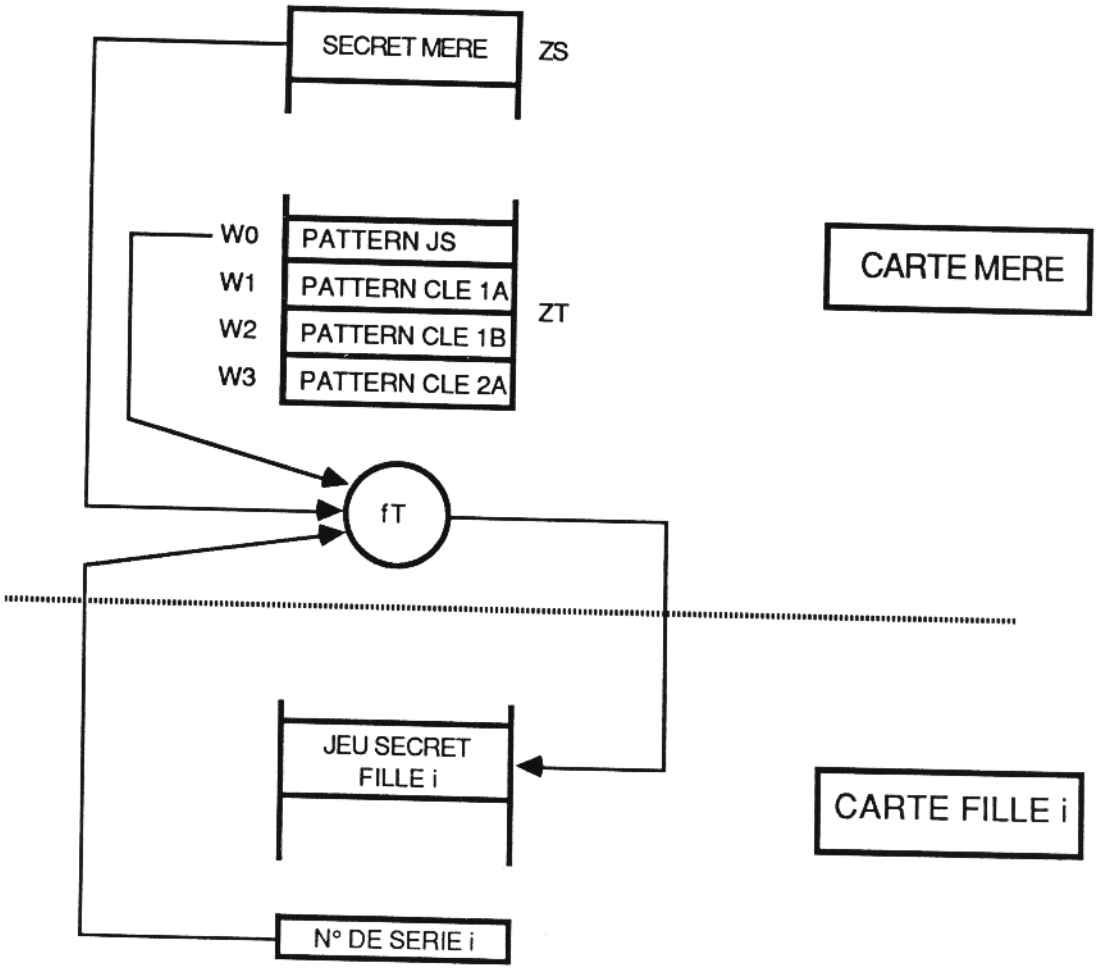
La Carte Mère contient des informations caractéristiques de l'application :

- Le Jeu Secret Mère, qui peut être défini arbitrairement ou calculé selon une méthode particulière,
- Des mots écrits dans la Zone Transaction.

Les informations spécifiques à chaque carte seront, soit le numéro de série (géré et attribué par BULL CP8) ou toute autre information applicative permettant de distinguer une carte d'une autre (numéro ISO des cartes bancaires).
Ces informations sont appelées PARAMETRES DE DIVERSIFICATION.

Ainsi, sont réunis tous les paramètres d'entrée d'une fonction TELEPASS :

- Un Jeu Secret, contenu dans la Zone Secrète de la Carte Mère,
- Des mots écrits dans la Zone Transaction, qui permettront, en faisant varier le paramètre ADR les désignant, l'obtention de R différents, qui seront l'un une clé 1A, l'autre une clé 1B, etc ...
- Un nombre aléatoire E qui aura la valeur du numéro de série, ou autre.



Jeu secret fille $_i$ = F_T (Secret Mère, Pattern JS, W0, N° de série $_i$)

Figure 1.6. : Principe de Diversification

Le contenu des mots Wi de la Carte Mère est déterminé arbitrairement par l'émetteur. Généralement, le mot d'adresse W_0 sert à calculer le Jeu Secret fille, le mot d'adresse W_1, la clé 1A, W_2, la clé 1B et W_3, le code porteur clé 2A.

La figure 1.6. présente le principe de Diversification du Jeu Secret ; le principe est le même pour les clés, seul variant l'adresse W_i (ADR).

1.3.3. PRINCIPES DE SECURISATION D'UN RESEAU

La sécurisation d'un réseau, grâce aux outils BULL CP8, repose sur les trois grands principes suivants :

- L'Authentification
- La Certification
- Le Chiffrement

1.3.3.1. L'authentification

1.3.3.1.1. Le principe

L'Authentification consiste pour un serveur à s'assurer que la carte introduite sur un lecteur-encodeur distant, est une carte CP8, donc capable d'exécuter la fonction TELEPASS, possédant dans sa Zone Secrète un Jeu Secret (de préférence diversifié), appartenant à la famille des Jeux Secrets habilités à accéder au service, et enfin, possédant à un endroit déterminé de sa mémoire (en Zone Transaction par exemple) des droits d'accès.

1.3.3.1.2. Les outils

Pour effectuer un tel traitement, les outils employés seront :

- Du côté carte porteur : un lecteur-encodeur de carte CP8
- Du côté serveur : un P.S.A. ou Processeur de Sécurité Associé contenant le(s) M.C.S. capable(s) d'exécuter la fonction TELEPASS et où est mémorisé le Jeu Secret Mère.

1.3.3.1.3. La mise en oeuvre

Le serveur génère un nombre aléatoire E (éventuellement par l'intermédiaire du P.S.A.) et l'envoie au lecteur distant.

Celui-ci effectue une fonction TELEPASS ayant pour paramètres un nombre aléatoire E et ADR pointant sur un mot de la carte de l'usager contenant des droits d'accès. Si ce mot est dans une zone en lecture protégée, la présentation et la validation d'une clé en lecture est nécessaire à l'exécution de la fonction TELEPASS.

Le lecteur renvoie le résultat du calcul au P.S.A. via le lecteur, ainsi que le nombre aléatoire et les paramètres de diversification de la carte usager (numéro de série ou numéro ISO ...).

A partir de ces données, le P.S.A. recalcule le Secret Diversifié de la carte distante puis effectue un calcul similaire à celui du lecteur distant.

Il compare les deux résultats et renvoie le résultat de la comparaison au serveur.

Ainsi, si le serveur reçoit un résultat de comparaison positif, il est assuré que la carte distante a effectivement été émise par l'organisme habilité et qu'elle contient des droits d'accès corrects.

D'autre part, si la clé présentée et validée en lecture est le code porteur de la carte, recevoir un bon résultat indique au serveur que le porteur de la carte a été identifié.

L'ensemble du dialogue d'Authentification n'est pas reproductible puisqu'un nombre aléatoire E est utilisé comme paramètre.

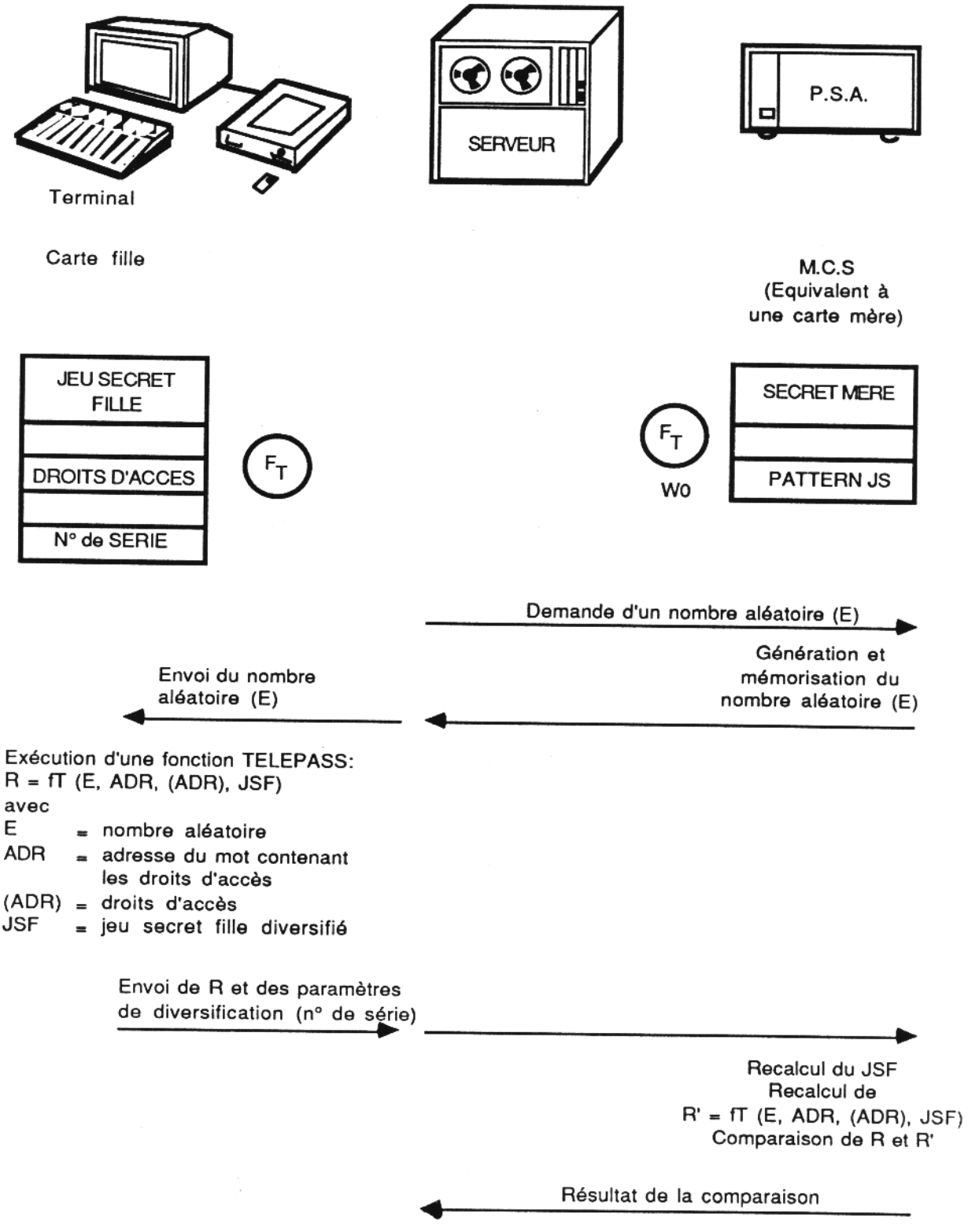


Figure 1.7. : Authentification

1.3.3.2. La certification

1.3.3.2.1. Le principe

La Certification consiste à s'assurer que la bonne information a été écrite à la bonne adresse dans la bonne carte.

1.3.3.2.2. Les outils

Identiques à l'Authentification.

1.3.3.2.3. La mise en oeuvre

Le serveur désire inscrire une transaction T à une adresse ADR de la carte de l'usager.

Le serveur génère un nombre aléatoire E (éventuellement par l'intermédiaire du P.S.A.) et l'envoie, ainsi que T et ADR, au lecteur distant.

Celui-ci inscrit la transaction dans la carte de l'usager à l'adresse indiquée puis effectue une fonction TELEPASS ayant pour paramètre E et le mot contenant la transaction.

Le lecteur renvoie le résultat du calcul au P.S.A. via le serveur, ainsi que la transaction et le numéro de série de la carte de l'usager.

A partir de ces données, le P.S.A. recalcule le Secret Diversifié de la carte distante puis effectue un calcul similaire à celui du lecteur.
Il compare les deux résultats et renvoie le résultat de la comparaison au serveur.

Le schéma de principe est identique à celui de l'Authentification à l'écriture prêt.

1.3.3.3. Le chiffrement

1.3.3.3.1. Le principe

Le Chiffrement consiste à envoyer des données en lignes en évitant le transport hasardeux de clés de chiffrement.

1.3.3.3.2. Les outils

Identiques à l'Authentification.

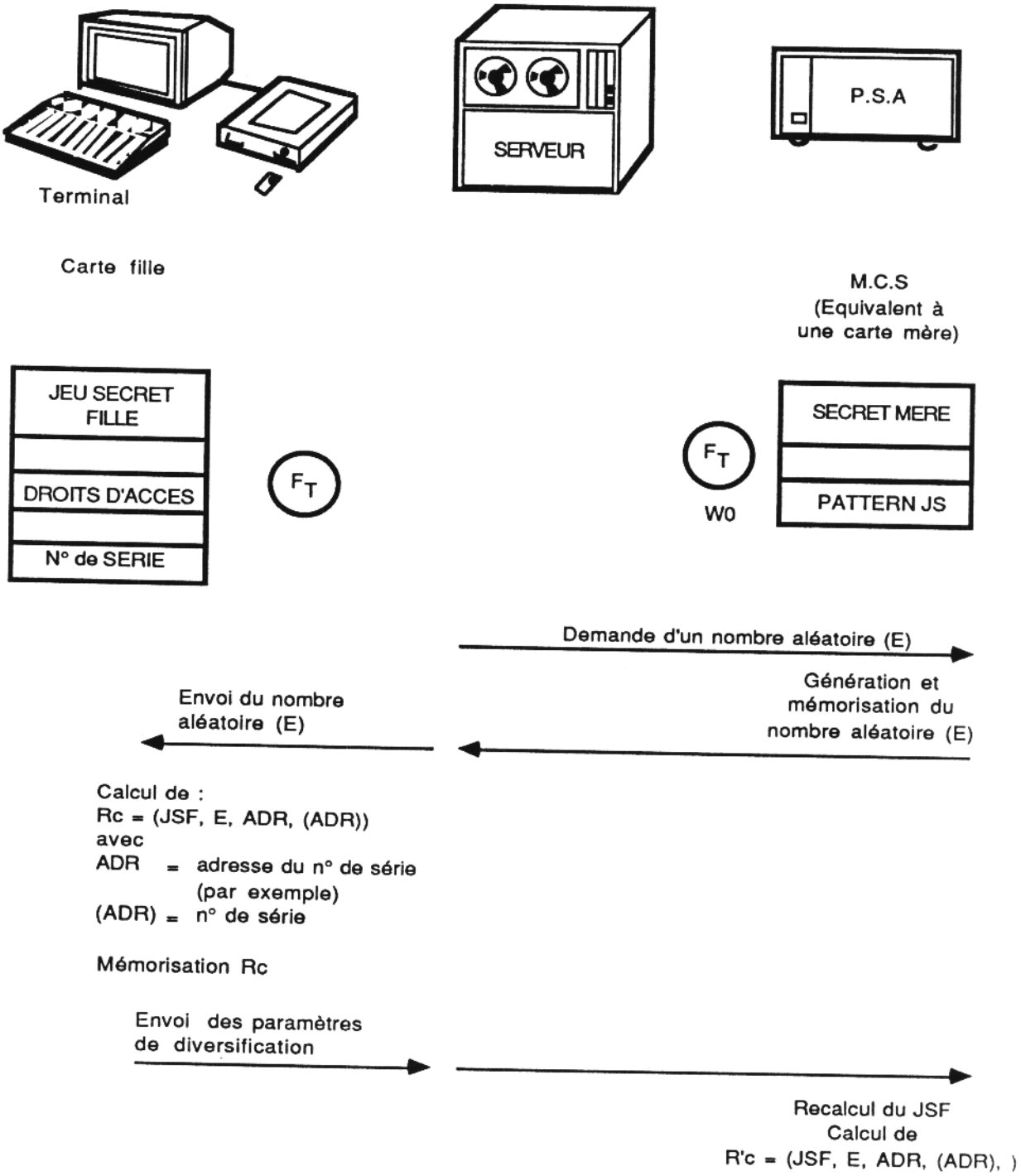
1.3.3.3.3. La mise en oeuvre

Le serveur génère un nombre aléatoire E (éventuellement par l'intermédiaire du P.S.A.) et l'envoie au lecteur distant.

Le P.S.A. et le lecteur effectuent simultanément une fonction TELEPASS ayant pour paramètre E et l'adresse du numéro de série de la carte fille.

Le résultat de ce calcul est par définition la clé de chiffrement qui va servir dans le lecteur et le serveur.

On remarquera que la clé de chiffrement est aléatoire puisque calculée à partir d'un nombre aléatoire. Ainsi on ne peut ni l'espionner, ni la reconstituer.



Rc et R'c sont les clés de chiffrement/déchiffrement

Figure 1.8 : Chiffrement

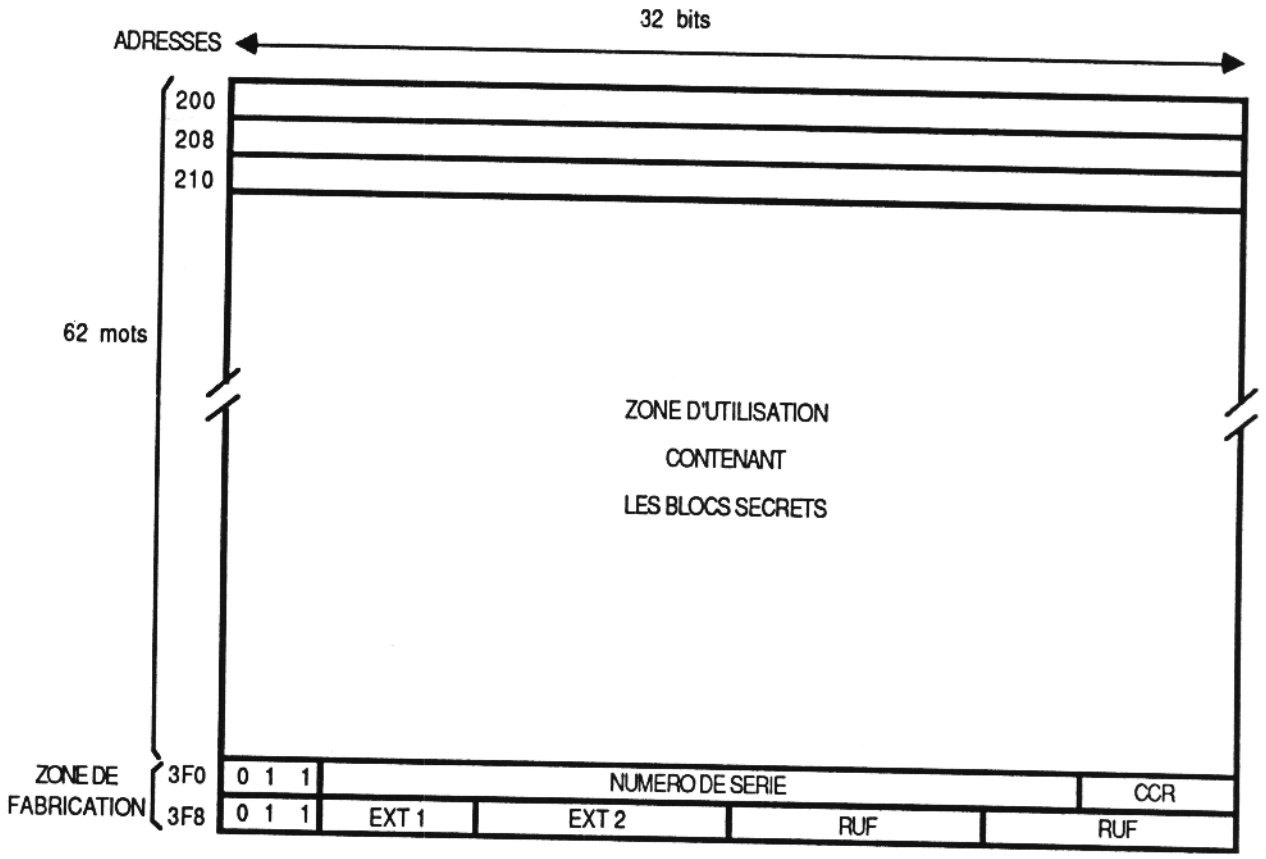
2. ORGANISATION DE LA MEMOIRE EPROM UTILISATEUR DU M.C.S.

La mémoire EPROM utilisateur M.C.S. est constituée de 64 mots de 32 bits chacun.

Elle est subdivisée en deux zones :

- La Zone Utilisation proprement dite, où seront mémorisés le ou les Jeux Secrets Mères, dans des "blocs secrets".

- La Zone de Fabrication contenant le numéro de série et le mot dit "d'extension de masque". Des familles de M.C.S. pouvant différer entre elles sur certaines caractéristiques fonctionnelles, le mot d'extension a été défini afin de les distinguer les unes des autres.



RUF : Réservé pour Utilisation Future
EXT1 et EXT2 : Code des extensions au masque de base
CCR : Code cyclique détecteur d'erreur

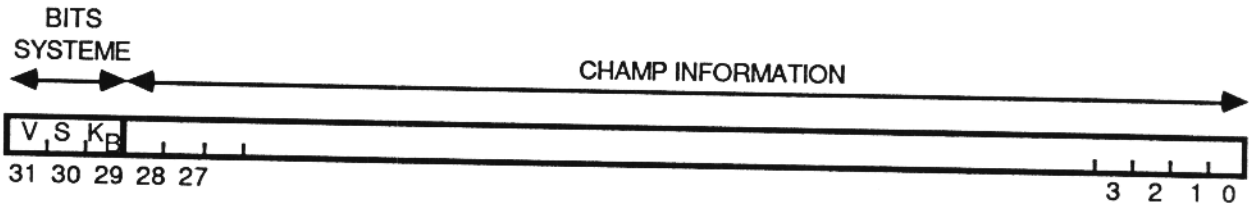
L'adresse de chaque mot est du type "quartet absolu", exprimée en hexadécimal.

Figure 2.1. : Organisation générale de la mémoire EPROM utilisateur

2.1. STRUCTURE DU MOT

Un mot de 32 bits est organisé en deux champs principaux :

- Un champ contenant les informations proprement dites.
- Un champ contenant les 3 bits dit "systèmes" définissant le niveau de protection d'accès au mot considéré.



2.2. BITS SYSTEME ET PROTECTION MEMOIRE

La protection d'accès en lecture / écriture à la mémoire EPROM se fait au niveau du mot, selon le positionnement des bits V et S.

2.2.1 LE BIT V

V = 1 : Le mot est dit " non validé " ; les autres bits système, quelque soit leur valeur n'auront pas de signification. Un mot non validé peut être lu et écrit.

V = 0 : Le mot est dit "validé" ; les autres bits système ont alors, selon leur valeur, les significations présentées ci-après.
Un mot validé ne peut être ni écrit ni modifié. Sa lisibilité dépend du bit S.

2.2.2. LE BIT S

S = 1 : Le mot est dit "non secret". Il peut être lu librement.

S = 0 : Le mot est dit "secret". Il est impossible de le lire. Dans ce cas, le bit correspondant au bit K_B sera considéré comme appartenant au champ information.

2.2.3. LE BIT K_B

K_B = 1: Le mot est un mot "courant" dans un bloc secret.

K_B = 0: Le mot est le premier mot d'un bloc secret.

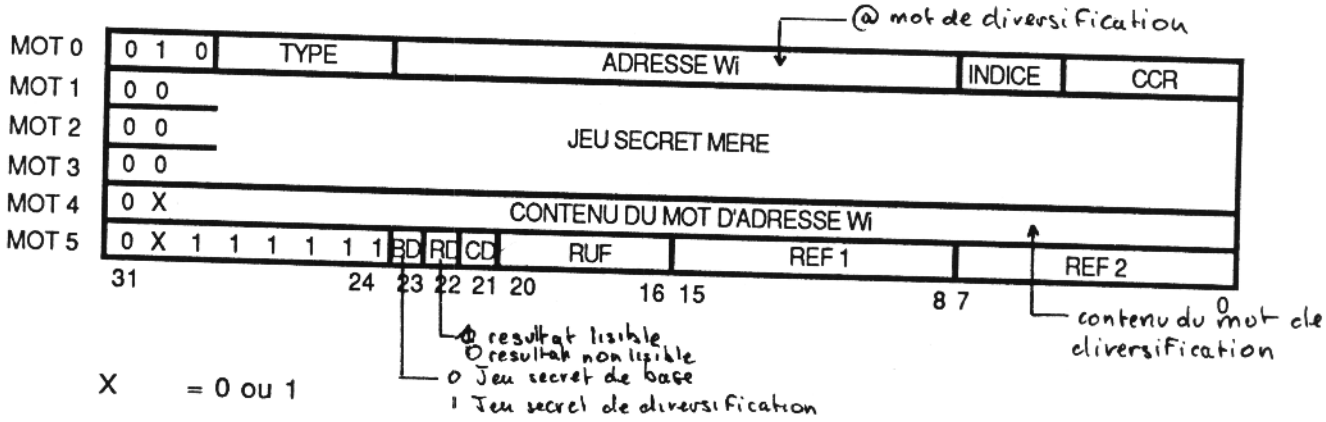
2.3. STRUCTURE D'UN BLOC SECRET

Un bloc secret, composé de 6 mots, contient principalement 2 types d'informations :

- Les informations caractéristiques de l'application, telles que le Jeu Secret Mère, le contenu du mot d'adresse W_i, etc...

- Les informations "systèmes", spécifiant au M.C.S. le mode d'utilisation du bloc , le caractère lisible ou pas des résultats etc...

NOTA : Une parfaite compréhension des informations contenues dans un bloc secret ne se fera qu'en deuxième lecture, après avoir pris connaissance des ordres du M.C.S.



X = 0 ou 1

RUF = Réservé pour utilisation future.

Figure 2.3. : Structure d'un bloc secret

2.3.1. TYPE, INDICE

Ce sont 2 champs, respectivement de 5 et 3 bits, permettant d'identifier et de sélecter le bloc.

2.3.2. Wi

C'est un champ de 16 bits contenant la valeur de l'adresse du mot de la Zone Transaction de la Carte Mère utilisé en phase de personnalisation pour la diversification du Jeu Secret. Dans certains blocs, cette valeur n'est pas utilisée ; dans ce cas Wi = FFFF.

2.3.3. CCR

Code détecteur d'erreur sur 5 bits, calculé sur l'ensemble des bits système (sauf le bit V), type, Wi et indice.

C'est le résultat de la division par le polynôme $G(x) = x^5 + x^2 + 1$.

Le CCR n'est pas contrôlé par le M.C.S., mais peut l'être par l'application.

2.3.4. JEU SECRET MERE

Le Jeu Secret Mère est identique à celui mémorisé dans la Carte Mère utilisée en phase de personnalisation.

On remarquera que ce mot fait obligatoirement 3 mots, alors que les cartes Masque 4/8/9 acceptent en théorie des Jeux Secrets de 1 à 8 mots. De fait donc, les Cartes Mères **auront toujours un Jeu Secret de 3 mots.**

2.3.5. [Wi]

Il contient le pattern du mot correspondant d'adresse Wi de la Zone Transaction de la Carte Mère. Dans certains blocs, cette valeur n'est pas utilisée, dans le cas (Wi) = 7FFF FFFF.

2.3.6. BD

BD = 1 : Lors de l'exécution la commande "Sélection de bloc", le M.C.S. n'effectuera aucun traitement particulié? Lors des commandes "Demande de calcul", le M.C.S. utilisera le Jeu Secret contenu dans le bloc. Dans ce cas, le Jeu Secret du bloc est appelé **Jeu Secret de Base**

BD = 0 : Lors de l'exécution de la commande "Sélection de bloc", le M.C.S. mettra en oeuvre le Jeu Secret contenu dans le bloc pour recalculer le Jeu Secret Diversifié d'une carte fille. Ce Jeu Secret diversifié **sera mémorisé dans la RAM du M.C.S.** Dans le cas, ce Jeu Secret du bloc est appelé **Jeu Secret de Diversification.**

2.3.7. RD

RD = 1 : Le M.C.S. pourra communiquer après réception de la commande "lecture de résultat", le résultat de calculs effectués, suite aux commandes "demande de calcul", "demande de calcul chaîné" et "repli de résultat". On dira alors que les résultats sont **lisibles.**

RD = 0 : Le M.C.S. ne communiquera pas le résultat des commandes de calcul. On dira alors que les résultats sont **non lisibles.**

2.3.8. CD, REF1 et REF2

CD = 1 : l'octet de poids fort, noté MCI, du mot à certifier doit être tel que :
REF1 AND MCI = REF2

CD = 0 : l'adresse ADR du mot à certifier doit être telle que :
PF (ADR) = REF1 et pf (ADR) = REF2.

AND = ET logique
PF : Poids Forts
pf : poids faibles

NB : Les informations RD, CD, REF1 et REF2 ne sont utilisées que par les M.C.S. MASQUE 2 (version HOST), alors qu'elles sont ignorées par les M.C.S. MASQUE 1 (version CTU).

3. ELEMENTS DU DIALOGUE M.C.S - TERMINAL

Ce chapitre est une présentation des octets rendus par le M.C.S lors de la RAZ (Remise à Zéro), et des grandes phases de déroulement d'un ordre. L'ensemble des échanges physiques et logiques entre le TERMINAL et le M.C.S est conforme au projet de norme ISO.

Définition Préliminaire :

Dans la suite du document, on appellera TERMINAL, tout équipement, simple ou complexe assurant l'interface physique et logique avec le M.C.S.

3.1 L' ORDRE RAZ (remise à zéro)

Avant d'exécuter un quelconque ordre, le composant doit s'initialiser et spécifier au TERMINAL ses caractéristiques propres. Tel est l'objet de l'ordre RAZ.
(On remarquera que l'ordre RAZ n'est pas un ordre au sens stricte du terme, car il est activé "hardwarement" par un signal appliqué sur le contact RAZ).

A l'issue de cette RAZ, le M.C.S envoie au TERMINAL 9 octets :
- 4 octets dits "système".
- 5 octets dits "application".

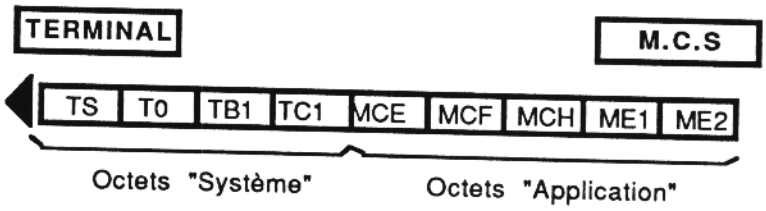


Figure 3.1 : Octets rendus à la RAZ

3.1.1 OCTETS SYSTEMES*

TS=3F : Spécifie que les données échangées seront transférées poids forts en tête, avec une parité paire, et que la fréquence d'horloge à fournir à la carte est 3.579545 Mhz.

T0=65 : Spécifie la présence ou l'absence des octets système TA1, TB1, TC1, TD1, et le nombre d'octets caractérisant l'application.

TB1=35 : Définit la tension et l'intensité de courant de programmation.

TC1=02 : Définit l'intervalle de temps minimum séparant l'émission par le TERMINAL de deux caractères.

* On trouvera une présentation détaillée de chacun de ces octets dans le projet de norme ISO.

3.1.2 OCTETS APPLICATION

3.1.2.1 MCE : MOT DES CARACTERISTIQUES EPROM

MCE = FF ; signifie que l'espace utilisateur fait 256 octets et que l'adresse de base est 200.

3.1.2.2 MCF : MOT DES CARACTERISTIQUES FONCTIONNELLES

Cet octet contient le numéro de masque du M.C.S auquel est associé un jeu d'ordre spécifique.

MCF = 01 ; M.C.S version C.T.U.
02 ; M.C.S version HOST.

3.1.2.3 MCH : MOT CHRONOLOGIQUE

MCH = 00 ; sans signification particulière.

3.1.2.4 ME1, ME2, COMPTE-RENDU D'EXECUTION

Ce sont les deux derniers octets rendus à la RAZ, mais également à la fin de tous les ordres.

ME1 = 90 ; fin normale.
67 ; octets de longueur incorrecte.
6B ; référence (PAR1, PAR2) incorrecte.
6D ; ordre inconnu.
6E ; classe d'instruction inconnue.

Les codes 67 à 6E ne sont susceptibles d'être rendus qu'après l'exécution d'un ordre, le code 90 étant rendu tout le temps (si tout s'est bien passé).

ME2 : Significatif si ME1 = 90.

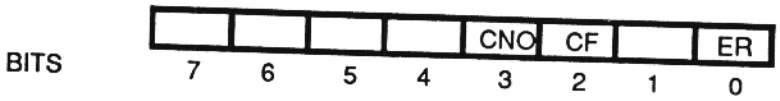


Figure 3.2 : ME2

ER = 1 : Erreur déclarée par le M.C.S, à la suite de laquelle il deviendra **muet** à la réception du prochain ordre, quelqu'il soit. Dans ce cas, seule une nouvelle RAZ permettra de réinstaurer le dialogue.

Si ER=1 immédiatement à la RAZ, le M.C.S **n'est pas utilisable (le rapporter à BULL CP8).**

ER = 0 : Rien à signaler.

CF = 1 : Résultats différents, suite à un ordre de comparaison de résultats.

CF = 0 : Résultats identiques.

CNO = 1 : Bloc secret non trouvé ou inutilisable, suite à un ordre de sélection de bloc.

CNO = 0 : Rien à signaler.

3.2 PRINCIPE GENERAL DE DEROULEMENT D'UN ORDRE

Les échanges entre le M.C.S et le monde extérieur sont réalisés conformément au projet de norme ISO.

Un ordre se déroule en 3 phases :

- Initialisation
- Exécution
- Fin

3.2.1 INITIALISATION

Lors de l'initialisation, le terminal envoie au M.C.S. 5 octets, décomposés comme suit :

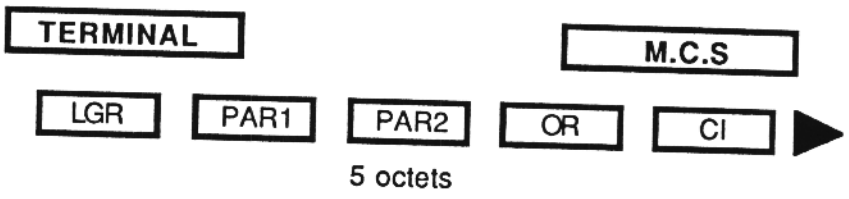


Figure 3.3 : initialisation d'un ordre

- **CI** = BC ; classe d'instruction.
- **OR (ou CODOP)** = code hexadécimal de l'ordre à exécuter.
- **PAR1 (ou A1), PAR2 (ou A2)** = adresse quartet de lecture, d'écriture, ou TYPE-INDICE.
- **LGR** = longueur en octets des données à transférer pendant la phase d'exécution, dans un sens ou dans l'autre, entre le M.C.S et le terminal.

Ces 5 octets constituent la commande indiquant au M.C.S ce qu'il doit faire. Selon l'ordre, des données seront ou ne seront pas échangées par la suite.

3.2.2 EXECUTION ; ORDRE ENTRANT, ORDRE SORTANT

Durant cette phase sont échangées les données entre le M.C.S et le TERMINAL.

On distingue deux types d'ordres.

On dit qu'un ordre est ENTRANT si les DONNEES sont échangées
dans le sens TERMINAL ---> M.C.S.

L'ordre d'ECRITURE est un ordre ENTRANT.
Lorsqu'aucune donnée n'est échangée, l'ordre correspondant est assimillé à un ordre ENTRANT.

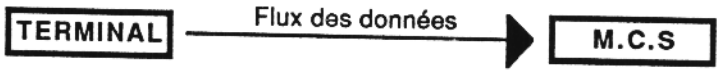


Figure 3.4 : Sens d'échange des données pendant un ordre ENTRANT

Inversement, on dit qu'un ordre est SORTANT si les DONNEES sont échangées
dans le sens M.C.S ---> TERMINAL.

L'ordre de LECTURE est un ordre SORTANT.

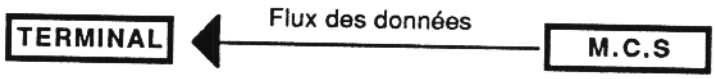


Figure 3.5 : Sens d'échange des données pendant un ordre SORTANT.

3.2.3 FIN

A l'issue de l'exécution d'un ordre, le M.C.S renvoie les 2 octets ME1, ME2.

4. MISE EN OEUVRE DES ORDRES ELEMENTAIRES DU M.C.S. MASQUE 2

Ce chapitre présente de façon détaillée, pour chaque ordre :

a) Le but de l'ordre
b) La fonction exécutée
c) Les conditions d'emploi
d) L'initialisation
e) Ce que le M.C.S renvoie
f) Les cas de mutisme

LE MUTISME :

Le mutisme signifie qu'au bout d'un temps déterminé, le M.C.S n'a envoyé aucun caractère.
Le mutisme est détecté par le logiciel du coupleur interfaçant le M.C.S.
Le M.C.S est muet lorsqu'on n'a pas observé correctement ses conditions d'utilisation.
Dans ce cas, il sera nécessaire de faire une R.A.Z pour réinstaurer le dialogue.

4.1 Lecture : OR = B0

Type : Sortant

a) But :

Lire l'EPROM utilisateur du M.C.S, y compris le numéro de série et le mot d'extension de masque.

b) Fonction exécutée

Lecture, à partir d'une adresse quartet absolue donnée, d'un ou plusieurs octets.
Seuls les octets appartenant à des mots dont le bit S = 1 sont lisibles ; dans le cas contraire, le M.C.S renvoie des octets "forcés" à 00. De même, si le nombre d'octets à lire dépasse l'espace adressable.

c) Conditions d'emploi

L'adresse de lecture spécifiée est une adresse quartet absolue, mot (i.e multiple de 8) appartenant à l'espace adressable (200-3FF).

d) Initialisation

CI	= BC
OR	= B0
PAR1	= Poids forts de l'adresse quartet de début de lecture
PAR2	= Poids faibles de l'adresse quartet de début de lecture
LGR	= nombre d'octets à lire
DONNEES	= aucune

e) le M.C.S renvoie

Les octets lus et ME1, ME2.

f) Cas de mutisme

L'adresse spécifiée est en dehors de l'espace adressable.

4.2 Ecriture : OR = D0

Type entrant

a) But :

Ecrire un mot dans l'espace utilisateur du M.C.S.

Rappel . un mot vierge est un mot dont tous les bits sont à 1. L'écriture consiste à faire passer un ou plusieurs bits de l'état 1 à l'état 0. Cette opération est irréversible.

b) Fonction exécutée

Le M.C.S, en fonction de la configuration binaire fournie en données, positionne à 0 les bits désirés. Le contrôle de l'écriture n'est pas faite par le M.C.S et est donc à la charge du Terminal. On ne peut écrire que dans un mot à la fois. Le bit V n'est jamais positionné à 0 lors de l'écriture.

c) Conditions d'emploi

On ne peut pas écrire dans un mot validé (Bit V = 0). L'adresse à fournir est nécessairement une adresse "quartet absolue mot" donc multiple de 8, appartenant à l'espace adressable (200 3FF).

d) Initialisation

CI = BC

OR = D0

PAR1 = Poids forts de l'adresse quartet absolue du mot où l'on veut écrire

PAR2 = Poids faibles de l'adresse quartet absolue du mot où l'on veut écrire

LGR = 04

DONNEES = 32 bits du mot à écrire. (L'information est fournie sous forme hexadécimale)

e) le M.C.S renvoie

ME1. ME2 : si le mot est déjà validé, le bit ER de ME2 sera positionné à 1 et le M.C.S deviendra muet sur réception du prochain ordre.

f) Cas de mutisme

- L'adresse est en dehors de l'espace adressable.
- L'adresse ne pointe pas sur une frontière mot.
- LGR ≠ 04.

4.3 Validation : OR = 70

Type : entrant

a) But :

Positionner à 0 le bit V d'un mot pour protéger l'information qu'il contient.

b) Fonction exécutée

Le M.C.S positionne à 0 le bit V d'un mot dont l'adresse lui a été fournie et contrôle si l'opération s'est déroulée avec succès.

c) Conditions d'emploi

On ne peut pas valider un mot déjà validé. L'adresse à fournir est nécessairement une adresse "quartet absolue mot", donc multiple de 8, appartenant à l'espace adressable. (200 - 3FF).

d) Initialisation

CI	= BC
OR	= 70
PAR1	= Poids forts de l'adresse quartet absolue du mot que l'on veut valider.
PAR2	= Poids faibles de l'adresse quartet absolue du mot que l'on veut valider.
LGR	= 00
DONNEES	= Aucune

e) le M.C.S renvoie

ME1, ME2 ; si le mot est déjà validé, le bit ER de ME2 sera positionné à 1 et le M.C.S deviendra muet sur réception du prochain ordre.

f) Cas de mutisme

- L'adresse est en dehors de l'espace adressable.
- L'adresse ne pointe pas sur une frontière mot.
- Le mot à valider l'est déjà.

4.4 Sélection d'un bloc Secret : OR = E0

Type : entrant

a) But :

Cet ordre permet de sélectionner un Jeu Secret Mère parmi ceux mémorisés dans l'espace utilisateur, et selon la valeur du bit BD, de recalculer un Jeu Secret Fille diversifié (avec la même méthode utilisée en personnalisation).

Par la suite, toutes les commandes de calcul envoyée au M.C.S seront exécutées soit avec le Jeu Secret Mère de base, soit avec le Jeu Secret Fille diversifié recalculé.

b) Fonction exécutée

Le M.C.S recherche dans l'espace utilisateur le bloc Secret dont les champs TYPE-INDICE sont identiques à ceux fournis en paramètre d'initialisation dans PAR2.

Si le bloc est présent et si BD = 0, autrement dit, si le Jeu Secret contenu est un Jeu Secret de Diversification, le M.C.S recalcule, avec les paramètres de diversification qui lui ont été fournis, le Jeu Secret diversifié de la Carte Fille et le mémorise dans sa RAM ; la longueur de ce Jeu Secret recalculé est implicitement 3 mots.

Si BD = 1, autrement dit si le Jeu Secret est un Jeu Secret de Base, le M.C.S ne fait aucun calcul. Dans ce cas, les 6 octets obligatoires à fournir en données sont non significatifs. Le Jeu Secret Diversifié ou le Jeu Secret de Base ainsi selectionnés seront utilisés lors des ordres de calculs suivants, ce jusqu'à ce qu'une nouvelle sélection soit activée pour mettre en oeuvre un autre bloc. A un instant donné, un bloc Secret et un seul est actif.

Dans le cas où plusieurs blocs ont les mêmes TYPE-INDICE, sera selectionné celui qui a la plus petite adresse.

c) Conditions d'emploi

Le M.C.S doit être correctement personnalisé.

d) Initialisation

CI = BC

OR = E0

PAR1 = 00 (non significatif)

PAR2 = TYPE-INDICE

LGR = 06

DONNEES = 6 octets constituant les paramètres de diversification si sélection d'un Jeu Secret de Diversification, ou 6 octets à 00, si sélection d'un Jeu Secret de Base.

e) le M.C.S renvoie

ME1, ME2 ; le bit CNO de ME2 sera positionné à 1 si :

. le bloc n'existe pas.

. le bloc est incomplet ou non conforme.

4.5 Demande de calcul : OR = 80

Type : entrant

a) But :

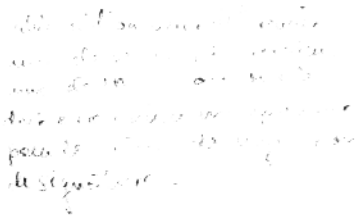
Un bloc Secret ayant été sélectionné, il est alors possible de faire exécuter au M.C.S une fonction TELEPASS mettant en oeuvre, soit le Secret de Base, soit le Secret Diversifié recalculé, ce afin de faire des contrôles de Certificat, calculs de clé de chiffrement, ou calculs de Clés Emetteurs Primaires ou Secondaires Diversifiées.

b) Fonction exécutée

Le M.C.S exécute une fonction TELEPASS, mettant en oeuvre soit un Secret de base, soit un Secret Diversifié, et les données E, ADR, et [ADR] fournies en données d'entrée.

- Si le bit CD du bloc est égal à 1, l'octet de poids forts de [ADR], noté MCI, doit satisfaire :
  MCI AND REF1 = REF2

- Si le bit CD du bloc est égal à 0, l'adresse ADR doit satisfaire :
  Octet de poids forts de ADR = REF1
  Octets de poids faibles de ADR = REF2



Le résultat de 64 bits de la fonction TELEPASS est stocké dans la RAM du M.C.S., mais n'est pas envoyé au terminal.

c) Conditions d'emploi

Un ordre de sélection de bloc Secret a été précédemment correctement exécuté.

d) Initialisation

CI	= BC
OR	= 80
PAR1	= 00 (non significatif)
PAR2	= TYPE-INDICE
LG	= 0C

DONNEES = E, ADR, [ADR]

avec E = nombre de 48 bits (6 octets)

ADR = adresse du mot sur lequel a été exécutée la fonction TELEPASS (dans la carte fille) sur 16 bits (2 octets)

[ADR] = contenu du mot sur 32 bits (4 octets)

e) Le M.C.S renvoie :

ME1, ME2 ; si PAR2 est différent du PAR2 envoyé lors de la précédente sélection de bloc Secret, le bit CNO de ME2 est positionné a 1.

f) Cas de mutisme

Aucune sélection de bloc Secret n'a été faite préalablement.

4.6 Demande de calcul chainé : OR = 84

Type : entrant

a) But :

Supposons que nous fassions exécuter à une Carte fille donnée, n fonctions TELEPASS sur n mots d'adresse et/ou de contenus différents, et que nous condensions les n résultats de 64 bits en un seul par une suite de XOR (ou exclusif), tels que

R = (((R1 XOR R2) XOR R3)... XOR Rn).

Un tel résultat R pourra être contrôlé, en plusieurs étapes, par le M.C.S.

b) Fonction exécutée

La fonction est identique (en particuler mêmes contrôles selon la valeur de CD) à la fonction demande de calcul, mais en plus, le résultat obtenu est "ou-exclusivé"* avec le résultat obtenu lors d'une demande de calcul ou d'une demande de calcul chainé précédemment réalisée.

Supposons que le M.C.S ait mémorisé dans sa RAM un résultat Ri, issu d'une précédente demande calcul, chainée ou pas.

Le M.C.S exécute, avec les nouveaux paramètres envoyés (E, ADR, [ADR]) une fonction TELEPASS et obtient un résultat R. Le résultat R_i +1 = Ri XOR R.

Le nouveau résultat R_i+1 est stocké dans la RAM du M.C.S., mais n'est pas envoyé au terminal.

c) Conditions d'emploi

Un ordre de demande calcul a été précédemment correctement exécuté.

On peut enchaîner, comme leur nom l'indique, les demandes de calcul chaîné.

* le néologisme est laid mais pratique.

d) Initialisation

CI = BC

OR = 84

PAR1 = 00 (non significatif)

PAR2 = TYPE-INDICE

LRG = 0C

DONNEES = E, ADR, [ADR]

avec E = nombre de 48 bits (6 octets)

ADR = adresse du mot sur lequel a été exécutée la fonction TELEPASS (dans la carte fille) sur 16 bits (2 octets).

[ADR] = Contenu du mot sur 32 bits (4 octets).

e) le M.C.S renvoie

ME1, ME2 ; si PAR2 est différent du PAR2 envoyé lors de la précédente solution de bloc Secret, le bit CNO de ME2 est positionné à 1.

f) Cas de mutisme

Aucune sélection de Bloc Secret ou demande de calcul n'a été faite préalablement.

4.7 Repli de résultat : OR = 40

Type : entrant

a) But :

Le résultat de la fonction TELEPASS a une longueur de 64 bits. Toutefois, pour des raisons diverses, on peut être amené à réduire ce résultat sur une longueur moindre, en perdant le minimum d'informations.

La réduction la plus simple consiste à faire un "ou-exclusif" des deux champs de 32 bits constituant le résultat, et ainsi obtenir une valeur sur 32 bits. L'algorithme de repli présenté ci-après effectue ce traitement, mais a été généralisé afin de pouvoir faire des réductions de longueur allant de 63 à 1 bit.

Si un tel type de réduction était utilisé, il faudrait évidement le développer dans le logiciel traitant la Carte fille.

b) Fonction exécutée

En fonction de la longueur L du résultat après repli, exprimée en nombre de bits,
$1 \leq L \leq 63$, le M.C.S. effectue le décalage suivant :

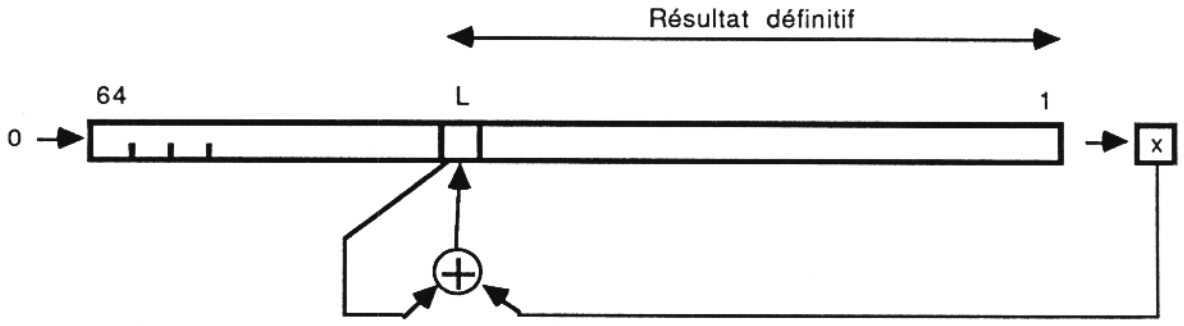


Figure 4.1 : Décalage

Les 64 bits sont décalés (64-L) fois à droite. Pour chaque décalage, le bit x sorti à droite est "ou-exclusivé" avec le bit de rang L et le résultat ainsi obtenu devient lui-même le bit de rang L. Les bits libérés à gauche sont forcés à 0.

Ainsi, à l'issue du repli, les bits restants significatifs sont cadrés à droite.

c) Conditions d'emploi

Une demande de calcul, chaînée ou pas, doit être préalablement correctement exécutée.

d) Initialisation

CI	= BC
OR	= 40
PAR1	= 00 (non significatif)
PAR2	= Longueur du résultat après repli
LGR	= 00
DONNEES	= aucune

e) Le M.C.S renvoie

ME1, ME2

f) Cas de mutisme

Si PAR2 est supérieur à 64 ou égal à 0 le M.C.S est muet.

4.8 Lecture de résultat : OR = C0

Type : Sortant

a) But :

Un ordre de calcul ayant été exécuté, on désire prendre connaissance du résultat qui sera utilisé soit comme clé de chiffrement, clé émetteur ou autre.

b) Fonction exécutée

Le M.C.S. envoi le résultat de la dernière opération de calcul ou de repli effectuée.

c) Conditions d'emploi

Une demande de calcul, suivie ou pas de calculs chainés et/ou de repli de résultat, a été précédemment correctement exécutée.

Le bit RD doit être positionné à 1.

d) Initialisation

CI	= BC
OR	= C0
PAR1	= 00 (non significatif)
PAR2	= 00 (non significatif)
LGR	= 08
DONNEES	= aucune

e) Le M.C.S renvoie :

Le M.C.S renvoie les 64 bits constituant le résultat, suivis de ME1, ME2.

f) Cas de mutisme

S'il n'y a pas de demande de calcul préalable ou si le bit RD est à 0, le M.C.S est muet.

4.9 Comparaison de résultat : OR = F0

Type : entrant

a) But :

Cet ordre permet de faire comparer par le M.C.S un certificat calculé par une carte fille à celui qu'on lui aura précédemment fait recalculer.

b) Fonction exécutée

Le M.C.S compare les 64 bits fournis en données aux 64 bits du résultat du calcul prédédent (demande calcul, calcul chaîné, ou repli).

L'indicateur CF de ME2 fournit le résultat de la comparaison.

Si cette fonction fait suite à un ordre repli, il est évident que la valeur à comparer aura subi le même repli que celle calculée par le M.C.S.

c) Conditions d'emploi

Une demande de calcul, suivie ou pas de calculs chainés et/ou de repli de résultat, a été précédemment correctement exécutée.

d) Initialisation

CI	= BC
OR	= F0
PAR1	= 00 (non significatif)
PAR2	= 00 (non significatif)
LGR	= 08
DONNEES	= 64 bits (8 octets) à comparer

e) Le M.C.S renvoie

ME1, ME2 ;

Le bit CF de ME2 est positionné à 1 si le résultat de la comparaison est négatif. Dans le cas contraire, il reste à 0.

4.10 Principe d'enchainement des ordres

La figure ci-dessous présente la manière dont on peut enchaîner les différents ordres élémentaires du M.C.S. On supposera que le M.C.S a été correctement personnalisé et mis sous tension. L'activation d'un ordre dans un séquencement différent de celui présenté se traduirait automatiquement par un MUTISME du M.C.S.

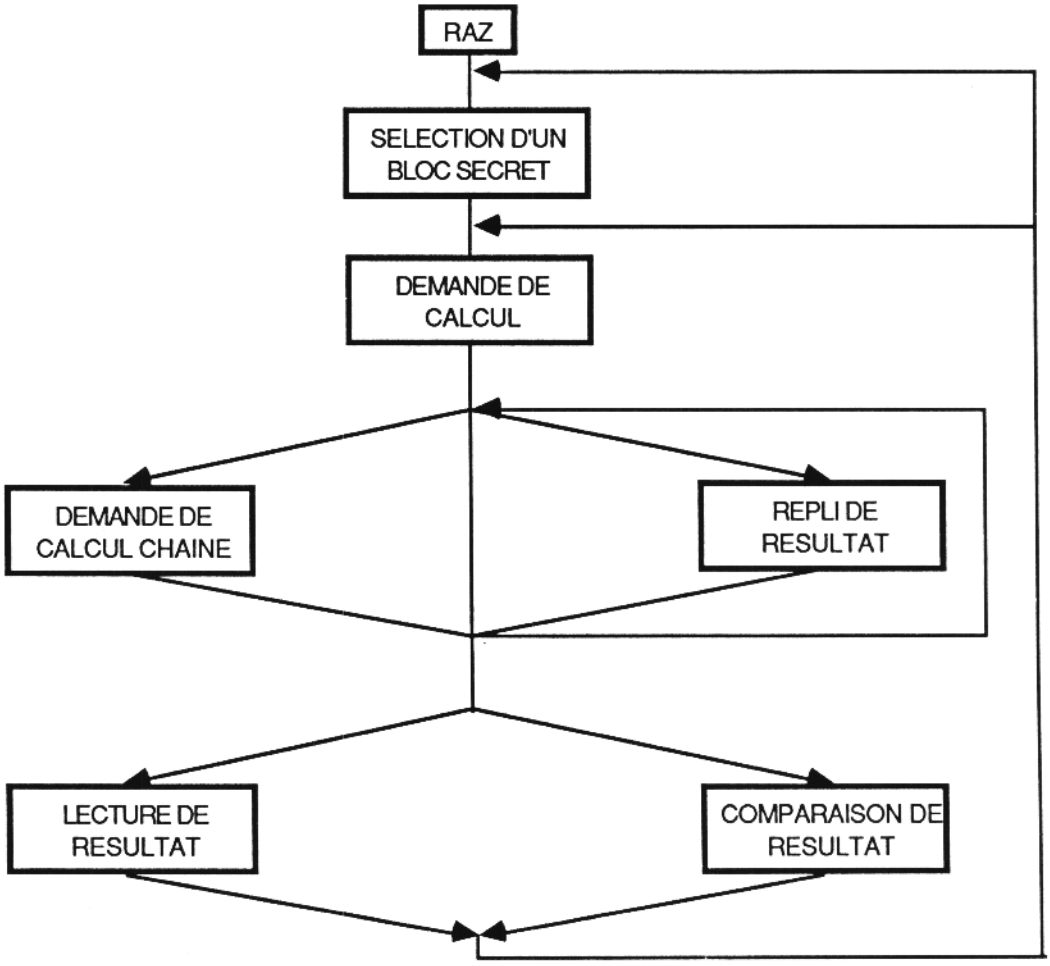


Figure 4.2 : Enchaînement des ordres du M.C.S. masque 2

Ainsi, on voit qu'un bloc secret ayant été sélectionné, on peut effectuer autant de calculs que l'on désire, jusqu'à la sélection d'un autre bloc, et ainsi de suite, tant que le M.C.S est sous tension.

5. MISE EN OEUVRE DES ORDRES ELEMENTAIRES DU M.C.S. MASQUE 1

Le M.C.S. Masque 1 est utilisé sur un terminal non connecté. Dans ces conditions, son jeu d'ordre est légèrement différent du M.C.S. Masque 2.

5.1. ORDRES IDENTIQUES A CEUX DU M.C.S. MASQUE 2

Les ordres cités ci-après sont à tout point de vue identiques aux ordres correspondants du Masque 2 :

- même fonction,
- même initialisation,
- même réponse.

* Lecture (B0).
* Ecriture (D0).
* Validation (70).
* Sélection d'un bloc secret (E0).
* Comparaison de résultat (F0).

5.2. ORDRES DIFFERENTS

5.2.1. DEMANDE DE CALCUL : OR = 80

La seule différence réside dans le fait que les bits du mot à certifier [ADR] correspondants aux bits systèmes V, C, doivent être tels que V = 0 et C = 1.

Autrement dit, cela signifie que le mot a été validé en écriture sous clé émetteur (si la Zone où il se trouve est protégée en écriture).

Si cette condition n'est pas respectée le M.C.S. deviendra MUET.

5.2.2. DEMANDE DE CALCUL CHAINE : OR = 84

Même différence que pour l'ordre de demande de calcul.

5.3. ORDRE NON TRAITE : LECTURE DE RESULTAT

Le M.C.S. masque 1 ne traite pas cet ordre. Si l'ordre lui est envoyé, il devient MUET.

5.4. PRINCIPE D'ENCHAINEMENT DES ORDRES

Identique à celui du masque 2, à l'ordre de lecture de résultat prêt.

6. APPLICATION

Ce chapitre présente un exemple de personnalisation de M.C.S. et les différents séquencement d'ordres à réaliser pour mener à bien une Authentification, Certification, etc...

Supposons que nous voulions, pour une famille donnée de cartes filles toutes diversifiées :
- faire de l'Authentification,
- du Contrôle de certificat,
- du Chiffrement,
- recalculer des Clés Emetteur Primaires ou Secondaires (clé 1A, clé 1B).

Pour cela, nous introduirons dans le M.C.S. (voir le châpitre Personnalisation) 4 blocs Secrets, ayant tous le même Secret Mère, mais avec des paramètres (BD, CD, ...) spécifiques à chaque type de traitement.

Ainsi, nous définirons un bloc pour l'Authentification / Contrôle de certificat, un bloc pour le calcul d'une Clé de Chiffrement, un autre pour le calcul des clés 1A et enfin un dernier pour le calcul des clés 1B.

Le Secret Mère sera identifié par le TYPE : 00001 (valeur binaire).

Les formats de blocs présentés ci-après correspondent à ceux **utilisés dans un P.S.A.**

6.1. BLOC POUR LE CALCUL D'UNE CLE DE CHIFFREMENT

Caractéristiques :

- Indice : 000 (valeur binaire).
- Le Jeu Secret Mère servira à recalculer **les Jeux Secrets Diversifiés.**
- Le résultat des demandes de calcul est **lisible.**
- La clé de Chiffrement sera le résultat d'une fonction TELEPASS exécutée avec pour ADR l'adresse du n° de série de la carte fille (adresse fixe 09F0).

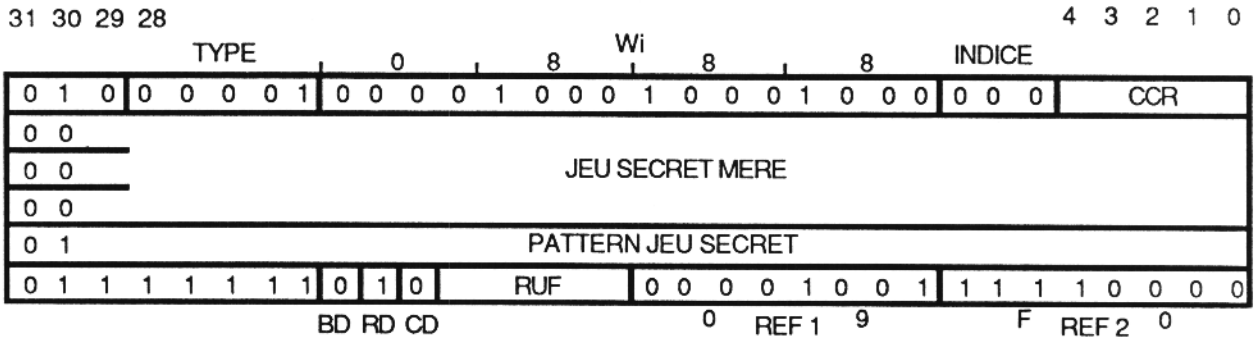


Figure 6.1. : Bloc pour le calcul d'une Clé de Chiffrement

6.2 BLOC POUR L'AUTHENTIFICATION ET/OU LE CONTROLE DE CERTIFICAT

Caractéristiques :

- Indice : 001 (valeur binaire).
- Le Jeu Secret Mère servira à recalculer les **Jeux Secrets Diversifiés.**
- Le résultat des demandes de calcul n'est pas **lisible.**
- REF1 et REF2 seront égaux à 00 pour que le résultat du test (AND) soit vrai, quelque soit la valeur de l'octet de poids fort du mot à certifier.

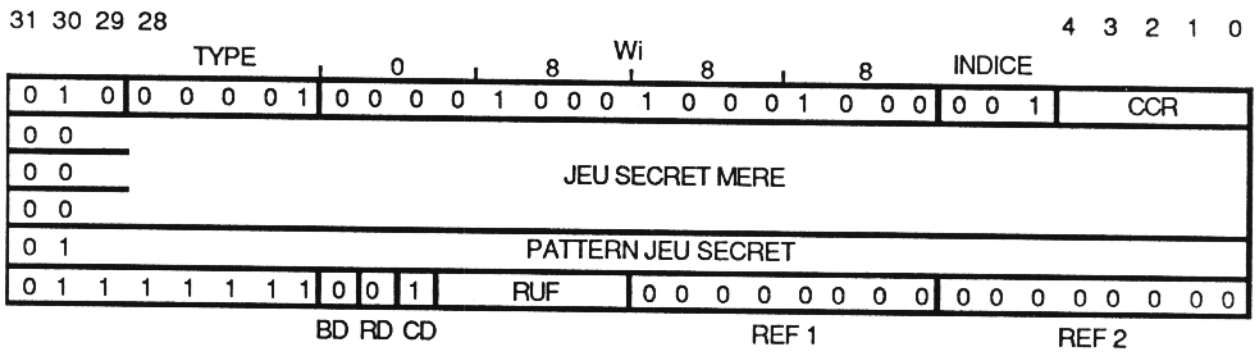


Figure 6.2. : Bloc pour l'authentification et/ou le contrôle de certificat

6.3. BLOC POUR LE CALCUL D'UNE CLE 1A

Caractéristiques :

- Indice : 010 (valeur binaire).
- Le Jeu Secret Mère est utilisé en tant que Jeu Secret de Base.
- Le résultat des demandes de calcul est lisible.
- REF1 et REF2 sont utilisés pour contrôler la cohérence de l'adresse fournie en paramètres au moment de la demande de calcul.

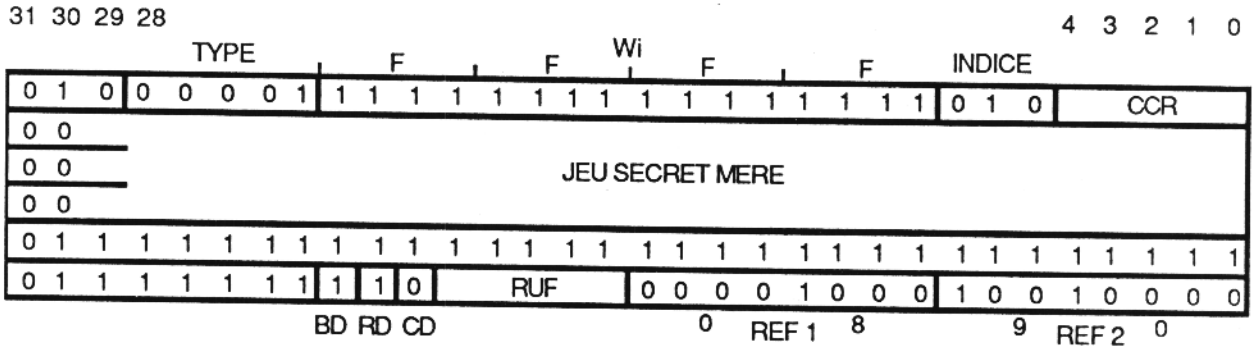


Figure 6.3. : Bloc pour le calcul d'une clé 1A

6.4. BLOC POUR LE CALCUL D'UNE CLE 1B

Caractéristiques :

- Indice : 011 (valeur binaire)
- Idem clé 1A.

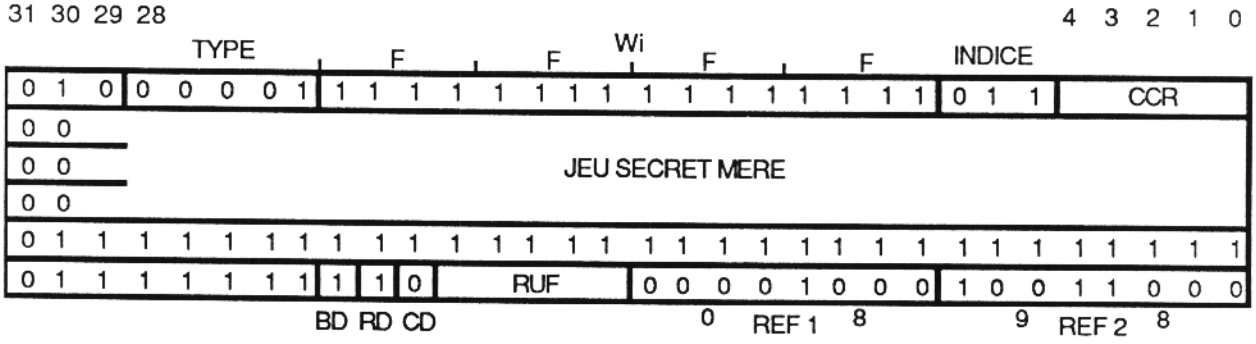


Figure 6.4. : Bloc pour le calcul d'une clé 1B

6.5. EXEMPLE D'ENCHAINEMENT DES ORDRES

Si nous voulons :

- faire une Authentification,
- calculer une clé de Chiffrement,
- calculer une clé 1A,

nous enchainerons les ordres suivants (paragraphe 6.5.1. à 6.5.3.).

6.5.1. AUTHENTIFICATION

1 - Sélection du bloc indice "001" et recalcul du Secret Diversifié.

CI = BC

OR = E0

PAR1 = 00

PAR2 = 09 (|0|0|0|0|1|0|0|1|)

LGR = 06

DONNEES = les 6 octets constituant les paramètres de diversification de la carte à authentifier.

2 - Demande de calcul

CI = BC

OR = 80

PAR1 = 00

PAR2 = 09

LGR = 0C

DONNEES = E, ADR, [ADR]

avec E = nombre aléatoire

ADR = adresse du mot contenant des droits d'accès (par exemple).

[ADR] = droits d'accès.

3 - Comparaison des résultats

CI = BC

OR = F0

PAR1 = 00

PAR2 = 00

LGR = 08

DONNEES = R calculé par la carte distante.

En retour, CF = 0 dans ME2 signifiera que le R calculé par la carte est bon.

6.5.2. CALCUL D'UNE CLE DE CHIFFREMENT

1 - A nouveau, sélection du bloc indice "000" et recalcul du Secret Diversifié.

CI = BC

OR = E0

PAR1 = 00

PAR2 = 08

LGR = 06

DONNEES = les 6 octets constituant les paramètres de diversification de la carte à authentifier.

2 - Demande de calcul

CI = BC

OR = 80

PAR1 = 00

PAR2 = 08

LGR = 0C

DONNEES = E, ADR, [ADR]

avec E, ADR, [ADR] identiques à ceux utilisés par la fonction TELEPASS exécutée par la carte fille distante.

3 - Lecture du résultat qui constituera la clé de Chiffrement

CI = BC

OR = C0

PAR1 = 00

PAR2 = 00

LGR = 08

DONNEES = aucunes

Le M.C.S. renvoie 64 bits.

6.5.3. CALCUL D'UNE CLE 1A

1 - Sélection du bloc indice "010"

CI = BC

OR = E0

PAR1 = 00

PAR2 = 0A

LGR = 06

DONNEES = 6 octets à 00 non significatifs.

2 - Demande de calcul

CI = BC

OR = 80

PAR1 = 00

PAR2 = 0A

LGR = 0C

DONNEES = E, ADR, [ADR]

avec E = paramètres de Diversification de la clé 1A.

ADR = 0890

[ADR] = pattern clé 1A

3 - Lecture du résultat qui constituera (après positionnement à 0 des bits correspondants aux bits V, C à 0) la clé 1A.

CI = BC

OR = C0

PAR1 = 00

PAR2 = 00

LGR = 08

DONNEES = Aucunes

Le M.C.S. renvoie 64 bits.

7. PERSONNALISATION

La personnalisation consiste à écrire dans le M.C.S. les différents blocs contenant le ou les secrets mères. Un M.C.S. peut contenir jusqu'à 10 blocs différents, le paramétrage de chacun des blocs dépendant des nécessités de l'application.

La personnalisation se déroule dans les locaux de BULL CP8 à Trappes, au cours d'une séance dite P.A.M. Il est préférable, avant de venir à cette séance, d'avoir préalablement clairement défini les blocs que l'on désire inscrire dans le M.C.S.

La dernière page de ce document est une fiche permettant la définition des informations qu'il faudra fournir à la machine de personnalisation (machine P.A.M) pour la génération d'un bloc. Il faudra donc remplir autant de fiches qu'il y a de blocs à définir.

La partie "Informations générales" permet d'identifier le M.C.S., le Jeu Secret, la nature du traitement que l'on désire faire avec le bloc correspondant. Elle permet de repérer d'un coup d'oeil une fiche parmi d'autres.

La seconde partie "Information à fournir à la machine P.A.M." contiendra les informations techniques qui seront saisies sur la machine. L'ordre et le format des champs suit exactement celui du menu de la machine P.A.M.

ANNEXE 1

TABLEAU RECAPITULATIF DES ORDRES DES M.C.S.

MASQUE 1 ET 2

Entrant/ Sortant	ordre	CI	OR	PAR1	PAR2	LG
S	Lecture	BC	B0	PF	pf	LGR
E	Ecriture	BC	D0	PF	pf	04
E	Validation	BC	70	PF	pf	00
E	Sélection bloc secret	BC	E0	00	T-I	06
E	Demande de calcul	BC	80	00	T-I	0C
E	Demande de calcul chaîné	BC	84	00	T-I	0C
E	Repli de résultat	BC	40	00	Lg. du repli	00
S	Lecture de * résultat	BC	C0	00	00	08
E	Comparaison de résultat	BC	F0	00	00	08

* Non traité par le masque 1.

ANNEXE 2

PERFORMANCE DU M.C.S. (Masque 1et 2)

Les valeurs présentées ci-après sont des valeurs moyennes, + ou - 1ms. Elles permettent cependant d'avoir une idée précise des performances du M.C.S.

Lecture d'un mot	:	19 ms
Ecriture d'un mot	:	223 ms
Validation d'un mot	:	65 ms

Sélection d'un bloc secret avec diversification :

- Implanté en adresse basse (210)	:	158 ms
- Implanté en adresse haute (3C0)	:	183 ms

Sélection d'un bloc secret de base :

- Implanté en adresse médiane	:	32 ms

Demande de calcul (selon les données envoyées)	:	171 à 189 ms

Repli de résultat sur :

- 32 bits	:	20 ms
- 16 bits	:	23 ms

Lecture de résultat	:	23 ms
Comparaison de résultat	:	24 ms

ANNEXE 3

REFERENCES COMMERCIALES DES MCS

(Marketing Identifier)

MCS MASQUE 1

TLF 0535-01 : MCS avec Secret test B0
0535-02 : MCS avec Secret test Telepass
0535-05 : MCS avec Secret test C0
0535-06 : MCS avec Secret test client

MCS MASQUE 2

TLF 0536-01 : MCS avec Secret test B0
0536-02 : MCS avec Secret test Telepass
0536-03 : MCS avec Secret test B0 pour retrofit PTF --> PSA
0536-05 : MCS avec Secret test C0
0536-06 : MCS avec Secret test client

TLF 0556 : Support MCS PDL 06

FICHE DE PERSONNALISATION DU M.C.S.

INFORMATIONS GENERALES

Date : .. / .. / 19..

Identification du M.C.S. :

Identification du Jeu Secret Mère :

Numéro du bloc dans le M.C.S. (1 à 10) :

Type de traitement à faire avec le bloc :

Calcul de clé de chiffrement ☐

Authentification, certification ☐

Calcul de Clé 1A ☐

Calcul de Clé 1B ☐

Calcul de Jeu Secret ☐

INFORMATIONS A FOURNIR A LA MACHINE P.A.M.

N.B. tous les caractéres saisis sont des chiffres hexadécimaux (0, 1, 2,...A, B,...E, F)

Numéro de Masque (Host = 2, CTU = 1) ☐

Type (2 caractères dont le 1er ≤1) ☐☐

Indice (1 caractère ≤7) ☐

Wi (4 caractères) ☐☐☐☐

[Wi] (8 caractères dont le 1er ≤7) ☐☐☐☐☐☐☐☐

BD (0 = Secret de Diversification
1 = Secret de Base) ☐

RD (0 = Résultat non lisible
1 = Résultat lisible) ☐

CD (0 = Test égalité
1 = AND) ☐

REF1 (2 caractères) ☐☐

REF2 (2 caractères) ☐☐

(Si demi-Secret saisis au clavier)

DEMI-SECRET A (12 caractères)* ☐☐☐☐☐☐☐☐☐☐☐☐
x x

DEMI-SECRET B (12 caractères) ☐☐☐☐☐☐☐☐☐☐☐☐
x

* les cases notées d'une croix correspondent au quartet de poids forts de chaque mot; penser à fournir une valeur telle que V,C = 0,0 i.e. comprise entre 0 et 3 bornes incluses.