



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

CCITT

V.42 *bis*

COMITÉ CONSULTATIF
INTERNATIONAL
TÉLÉGRAPHIQUE ET TÉLÉPHONIQUE

**COMMUNICATION DE DONNÉES
SUR LE RÉSEAU TÉLÉPHONIQUE**

**PROCÉDURES DE COMPRESSION
DES DONNÉES POUR LES ÉQUIPEMENTS
DE TERMINAISON
DU CIRCUIT DE DONNÉES (ETCD) UTILISANT
DES PROCÉDURES DE CORRECTION D'ERREUR**

Recommandation V.42 *bis*



Genève, 1990

AVANT-PROPOS

Le CCITT (Comité consultatif international télégraphique et téléphonique) est l'organe permanent de l'Union internationale des télécommunications (UIT). Il est chargé de l'étude des questions techniques, d'exploitation et de tarification, et émet à ce sujet des Recommandations en vue de la normalisation des télécommunications à l'échelle mondiale.

L'Assemblée plénière du CCITT, qui se réunit tous les quatre ans, détermine les thèmes d'études et approuve les Recommandations rédigées par ses Commissions d'études. Entre les Assemblées plénières, l'approbation des Recommandations par les membres du CCITT s'effectue selon la procédure définie dans la Résolution n° 2 du CCITT (Melbourne, 1988).

La Recommandation V.42 *bis*, que l'on doit à la Commission d'études XVII, a été approuvée le 31 janvier 1990 selon la procédure définie dans la Résolution n° 2.

© UIT 1990

Droits de reproduction réservés. Aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'éditeur.

Recommandation V.42 bis

PROCÉDURES DE COMPRESSION DES DONNÉES POUR LES ÉQUIPEMENTS DE TERMINAISON DU CIRCUIT DE DONNÉES (ETCD) UTILISANT DES PROCÉDURES DE CORRECTION D'ERREUR

Le CCITT,

considérant

- (a) que les ETCD conformes aux Recommandations de la série V sont largement utilisés pour la transmission de données en mode asynchrone dans un réseau téléphonique général avec commutation (RTPG);
- (b) que la Recommandation V.42 [1] définit des procédures de correction d'erreur qui permettent d'améliorer le taux d'erreur;
- (c) qu'il est possible d'améliorer le débit en utilisant des procédures de compression de données;
- (d) qu'il est nécessaire d'assurer l'interfonctionnement avec les ETCD non dotés de systèmes de compression de données,

recommande

d'adopter, dans les ETCD qui utilisent les procédures de correction d'erreur décrites dans la Recommandation V.42, les procédures de compression spécifiées dans la présente Recommandation.

1 Champ d'application

1.1 Considérations générales

La présente Recommandation décrit une procédure de compression des données applicable aux ETCD conformes aux Recommandations de la série V.

Cette procédure présente les principales caractéristiques suivantes:

- a) la compression, qui se fait par algorithme, consiste à coder les chaînes de caractères reçues de l'équipement terminal de traitement de données (ETTD);
- b) la procédure de décodage assure la récupération des chaînes de caractères à partir des mots-codes;
- c) lorsque les données sont détectées comme étant incompressibles, le système fonctionne en mode transparent automatique.

Une explication des paramètres utilisés dans cette Recommandation est donnée au § 10.

1.2 Spécifications des procédures de correction d'erreur

Pour que la fonction de compression des données soit exécutée correctement, il est nécessaire d'appliquer une procédure de correction d'erreur entre les deux entités appliquant la présente Recommandation. Dans le cas des Recommandations de la série V, il faut que les procédures LAPM (procédure d'accès à la liaison pour modems) de correction d'erreur définies dans la Recommandation V.42 ou V.120 [2] soient appliquées.

Remarque — Des erreurs sur les bits non décelées perturberont la fonction de compression des données. L'emploi d'une séquence de contrôle de trame (FCS) à 32 bits telle que définie dans la norme ISO 3309 [3] réduit considérablement la possibilité d'erreurs de ce genre. Il peut donc être souhaitable d'utiliser l'option FCS à 32 bits (qui est une option de la LAPM de la Recommandation V.42) dans un milieu où il existe de sérieuses défaillances.

1.3 ETCD à compression des données

La fonction de compression des données peut être appliquée à un ETCD à correction d'erreur, comme le montre la figure 1/V.42 bis. Les éléments d'un ETCD à correction d'erreur de la série V sont spécifiés dans la Recommandation V.42.

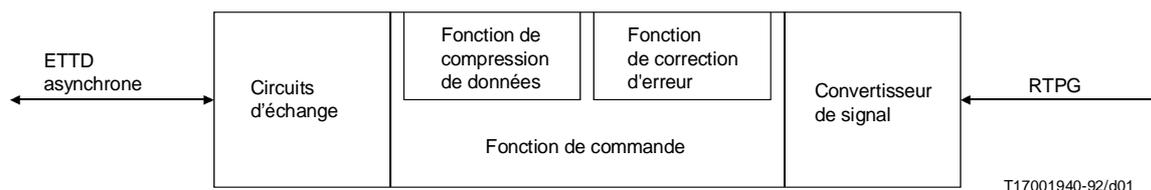


FIGURE 1/V.42 bis
ETCD doté d'un système de compression des données
et de limitation d'erreur

2 Définitions

2.1 caractère

Élément de donnée unique, codé, comportant un nombre prédéfini d'éléments binaires ($N_3 = 8$).

2.2 format: arythmique ou asynchrone

Les formats arythmiques et asynchrones sont définis dans les Recommandations V.7 [4] et V.14 [5].

2.3 valeur ordinale

On entend par valeur ordinale d'un caractère la représentation numérique de son codage binaire. Par exemple, lorsque la représentation codée du caractère «A» est 01000001, sa valeur ordinale est de 65_{10} .

2.4 alphabet

Ensemble de tous les caractères pouvant être émis ou reçus par l'intermédiaire de l'interface ETTD/ETCD. Dans la présente Recommandation, on suppose que les valeurs ordinales de l'alphabet constituent une progression continue de 0 à $N_4 - 1$, N_4 étant le nombre de caractères.

2.5 mot-code

Dans le contexte de la présente Recommandation, un mot-code est un nombre binaire pris dans la gamme de 0 à $N_2 - 1$ qui représente une chaîne de caractères sous forme comprimée. Un mot-code est codé à l'aide d'un nombre de bits, C_2 , dans lequel C_2 est initialement 9 (c'est-à-dire $N_3 + 1$) et augmente jusqu'à un maximum de N_1 bits (voir le § 7).

2.6 mot-code de commande

Un mot-code de commande est réservé pour utilisation dans la signalisation entre des ETCD d'informations de commande liées à la fonction de compression lorsque les équipements sont en mode compression (voir le § 9).

2.7 **code de commande**

Octet qui est utilisé pour la signalisation entre des ETCD d'informations de commande liées à la fonction de compression lorsque les équipements fonctionnent en mode transparent. Les codes de commande se distinguent des caractères normaux par un caractère d'échappement qui les précède (voir le § 2.13).

2.8 **structure arborescente**

Structure de données abstraite qui est utilisée dans la présente Recommandation pour représenter un ensemble de chaînes ayant le même caractère initial (voir la figure 2/V.42 *bis* et le § 6.1).

2.9 **noeud feuille**

Point situé sur un arbre qui représente, dans le contexte de la présente Recommandation, le dernier caractère d'une chaîne (voir le § 6.1).

2.10 **noeud racine**

Point situé sur un arbre qui représente, dans le contexte de la présente Recommandation, le premier caractère d'une chaîne (voir la figure 2/V.42 *bis* et le § 6.1).

2.11 **compression des données**

La compression des données s'effectue dans deux modes. Le passage de l'un à l'autre peut être automatique, selon le contenu des données reçues de l'ETTD (voir le § 7.1).

2.11.1 **mode compression**

Mode de fonctionnement dans lequel les données reçues de l'ETTD sont transmises en modes-codes.

2.11.2 **mode transparent**

Mode de fonctionnement dans lequel la compression a été choisie, mais où les données sont transmises sous forme non comprimée. Des séquences de codes de commande en mode transparent peuvent être insérées dans le train de données.

2.12 **mode sans compression**

Mode de fonctionnement dans lequel la compression n'a pas été choisie. La fonction compression des données est donc inactive.

2.13 **caractère d'échappement**

Dans le cadre de la présente Recommandation, on entend par caractère d'échappement un caractère qui indique le début d'une séquence de codes de commande en mode transparent. Cette séquence, qui a pour valeur initiale «0», est adaptée chaque fois qu'un caractère d'échappement apparaît dans le train de données provenant de l'ETTD, que ce soit en mode transparent ou en mode compression (voir le § 9.2).

3 **Abréviations**

Les abréviations suivantes s'ajoutent à celles de la Recommandation V.42:

EDD Echappement dans les données, code de commande défini au § 9;

PMT Passage au mode transparent, mot-code de commande défini au § 9;

PMC Passage au mode compression, code de commande défini au § 9.

4 Description générale du fonctionnement d'un ETCD à compression des données

4.1 Considérations générales

La figure 1/V.42 *bis* représente un ETCD à compression des données, qui comporte les éléments suivants:

- a) des circuits d'échange ETTD/ETCD;
- b) un convertisseur de signaux;
- c) une fonction de commande;
- d) une fonction de correction d'erreur; et
- e) une fonction de compression des données.

La fonction de commande présente des possibilités additionnelles qui complètent les caractéristiques de l'ETCD à correction d'erreur défini dans la Recommandation V.42. Ces possibilités additionnelles sont décrites au § 5, alors que la fonction de compression des données fait l'objet des § 6 à 9. Le reste du § 4 décrit, dans ses grandes lignes, la fonction de commande et la fonction de compression des données.

4.2 Présentation générale de la fonction de commande

Outre les fonctions désignées au § 6.2 de la Recommandation V.42, la fonction de commande présente les éléments suivants:

- a) négociation de la présence de la fonction de compression des données dans l'ETCD distant et des paramètres associés à la fonction de compression des données;
- b) initialisation ou réinitialisation de la fonction de compression des données;
- c) coordination de l'établissement d'une connexion de correction d'erreur en vue d'une utilisation par les fonctions de compression des données des deux extrémités;
- d) coordination de la remise des données entre l'interface ETTD/ETCD et la fonction de compression des données, conformément aux procédures définies dans la Recommandation V.42, § 6.2 et 8.4, et notamment mise en place des procédures de contrôle de flux définies dans ladite Recommandation;
- e) coordination de la remise des données entre la fonction de compression des données et la fonction de correction d'erreur;
- f) intervention sur détection d'un état d'interruption.

4.3 Présentation générale de la fonction de compression des données

La fonction de compression des données exécute les procédures définies dans la présente Recommandation, qui assurent un codage efficace des données avant acheminement par la connexion à correction d'erreur; cette fonction assure:

- a) l'initialisation de la fonction de compression des données;
- b) le codage et le décodage des données comprimées;
- c) un mécanisme de commutation entre le mode compression et le mode transparent.

4.4 Communication entre la fonction de commande et la fonction de compression des données

La communication entre la fonction de commande et la fonction de compression des données est modélisée par un ensemble de primitives abstraites de forme X-NOM TYPE, qui représentent l'échange logique d'informations et de commandes correspondant à l'exécution de la tâche ou du service demandé. Dans le cadre de la présente Recommandation, la fonction de commande est considérée comme «utilisatrice de service» alors que la fonction de compression des données est considérée comme «fournisseur de service». Les types de primitive sont: demande, indication, réponse et confirmation.

La liste des services attendus de la fonction de commande figure au tableau 1/V.42 *bis*

TABLEAU 1/V.42 bis

Services attendus de la fonction de commande

Service	Primitive	§
Initialisation de la fonction de compression des données	C-INIT	5.2, 5.6
Indication d'une erreur à la fonction de commande	C-ERROR	5.8
Transfert des données non comprimées à destination ou en provenance de la fonction de compression des données	C-DATA	5.4
Transfert des données comprimées à destination ou en provenance de la fonction de compression des données	C-TRANSFER	5.5
Transfert des données restantes: Transfert des données non encore transmises présentes dans l'étage de codage	C-FLUSH	5.7

5 Fonction de commande**5.1 Négociation de la fonction de compression des données**

L'utilisation de la fonction de compression des données et des paramètres associés doit être négociée pendant l'établissement de la liaison par l'intermédiaire d'un protocole (par exemple, au moyen de la procédure XID définie dans la Recommandation V.42). Cette fonction et ces paramètres demeurent ensuite inchangés au cours de la connexion à correction d'erreur.

Le paramètre P_0 précise si la compression doit être utilisée ou non. Ce paramètre spécifie aussi les directions (émission seulement, réception seulement ou les deux directions). La valeur par défaut de P_0 est 0, ce qui signifie qu'il n'y a pas de compression dans l'une ou l'autre direction. Si la compression est proposée pour une seule direction, la seule réponse valable est soit pour la direction proposée, soit sans compression. Si la compression est proposée pour les deux directions, il existe des réponses valables soit pour les deux directions, soit pour une seule direction ou encore sans compression.

Le paramètre P_1 représente la valeur proposée de N_2 , soit le nombre de modes-codes. P_1 aura une valeur par défaut de 512, soit sa valeur minimale. Une valeur maximale n'est pas spécifiée dans la présente Recommandation. Toute tentative de spécifier moins que la valeur minimale doit être considérée comme une erreur de procédure et aboutit à la déconnexion. Lorsque des valeurs de P_1 sont échangées au cours de la procédure de négociation dans une direction ou dans les deux directions de transmission, c'est la valeur inférieure qui est choisie et affectée à N_2 dans les deux ETCD.

Remarque — Voir l'appendice II pour plus de renseignements sur le choix de la valeur de N_2 et sur l'incidence de ce choix sur la qualité de fonctionnement.

Le paramètre P_2 est la valeur proposée pour N_7 , la longueur maximale de la chaîne. La valeur par défaut de P_2 est 6 et l'intervalle admissible est compris entre 6 et 250. Les valeurs qui se trouvent en dehors de cet intervalle ne sont pas valables; toute tentative de spécifier de telles valeurs doit être considérée comme une erreur de procédure et aboutit à la déconnexion. Lorsque des valeurs de P_2 sont échangées pendant la procédure de négociation, c'est la valeur inférieure qui est choisie et assignée à N_7 dans les deux ETCD.

5.2 Initialisation de la fonction de compression des données

Après la négociation des paramètres de compression des données, la fonction de commande adresse à la fonction de compression des données la primitive de demande C-INIT. Cette primitive indique les valeurs des paramètres négociés.

5.3 *Etablissement de la connexion*

A la réception de la primitive de confirmation C-INIT adressée par la fonction de compression des données, la fonction de commande indique à l'ETTD que le transfert de données peut commencer.

5.4 *Coordination du transfert des données entre l'interface ETTD/ETCD et la fonction de compression des données*

Une fois la connexion établie, la fonction de commande demande le codage des données reçues à l'interface ETTD/ETCD.

Pour coder les données, la fonction de commande adresse à la fonction de compression des données une primitive de demande C-DATA. Cette primitive indique les données à coder.

A la réception de la primitive d'indication C-DATA adressée par la fonction de compression des données, la fonction de commande livre les données décodées à l'interface ETTD/ETCD.

Il faudra prévoir des procédures de contrôle de flux pour éviter la perte éventuelle de données due au débordement de la mémoire. Lorsque les procédures définies dans la présente Recommandation sont utilisées en conjonction avec celles de la Recommandation V.42, les procédures de contrôle de flux définies aux § 7.3.1 et 8.4.2 de la Recommandation V.42 s'appliquent.

5.5 *Coordination du transfert de données entre la fonction de compression des données et la fonction de correction d'erreur*

A la réception d'une primitive d'indication C-TRANSFER adressée par la fonction de compression des données, la fonction de commande adresse à la fonction de correction d'erreur une primitive de demande L-DATA.

A la réception d'une primitive d'indication L-DATA adressée par la fonction de correction d'erreur, la fonction de commande adresse à la fonction de compression des données une primitive de demande C-TRANSFER.

5.6 *Réinitialisation de la fonction de compression des données*

La fonction de commande adresse à la fonction de compression des données une demande C-INIT aux conditions suivantes:

- a) indication ou confirmation L-ESTABLISH;
- b) indication ou confirmation L-SIGNAL, quand la primitive indique une forme destructive.

Il appartient aux fonctions de commande de faire en sorte que les primitives de demande C-INIT soient adressées seulement lorsqu'il n'y a pas de données en transit entre les fonctions de compression des données (par exemple, dans les fonctions de correction d'erreur) pour assurer la synchronisation entre les codeurs et les décodeurs.

5.7 *Transfert de données accéléré*

Dans certaines conditions, non visées par la présente Recommandation, il peut être nécessaire de transférer immédiatement certaines données partiellement codées, par exemple, lorsque la fonction de correction d'erreur est inactive: la fonction de commande adresse alors une primitive de demande C-FLUSH à la fonction de compression des données et transfère les données restantes conformément au § 5.5.

5.8 *Action lors de la détection de l'indication C-ERROR*

L'indication C-ERROR sert à informer la fonction de commande que la fonction de compression des données a décelé une erreur (par exemple, une erreur de procédure ou la perte de synchronisation). La fonction de commande prend les dispositions de rétablissement appropriées, y compris le rétablissement de la connexion avec procédure de contrôle d'erreur.

Les conditions suivantes, reconnues par le décodeur, aboutissent à la génération d'une primitive d'indication C-ERROR:

- a) réception d'un mot-code STEPUP entraînant pour la valeur de C_2 un dépassement de N_1 ;
- b) réception, à tout moment, d'un mot-code égal à C_1 ;
- c) réception d'un mot-code représentant une entrée de dictionnaire vide;

- d) réception d'un code de commande réservé.

6 Procédures d'utilisation et de maintenance du dictionnaire

6.1 Observations générales

La fonction de compression des données exploite un algorithme dans lequel les chaînes de caractères reçues de l'ETTD sont codées sous forme de mots-codes de longueur fixe. La procédure fait appel à des dictionnaires dans lesquels les chaînes sont stockées et qui font l'objet de mises à jour dynamiques en mode de fonctionnement normal.

La fonction de compression des données comprend en fait deux dictionnaires. Le premier, localisé au niveau du codeur de compression, assure la compression des données reçues de l'ETTD; le second, au niveau du décodeur de compression, sert au décodage des données reçues de la fonction de correction d'erreur.

Le dictionnaire exécute ou subit un certain nombre d'opérations:

- a) mise en correspondance des chaînes: une séquence de caractères est lue à la sortie de l'ETTD, et le dictionnaire cherche la chaîne résultante (voir le § 6.3);
- b) mise à jour: adjonction d'une nouvelle chaîne au dictionnaire (voir le § 6.4);
- c) suppression des chaînes de caractères rarement utilisées: possibilité de réutilisation de l'espace de stockage (voir le § 6.5).

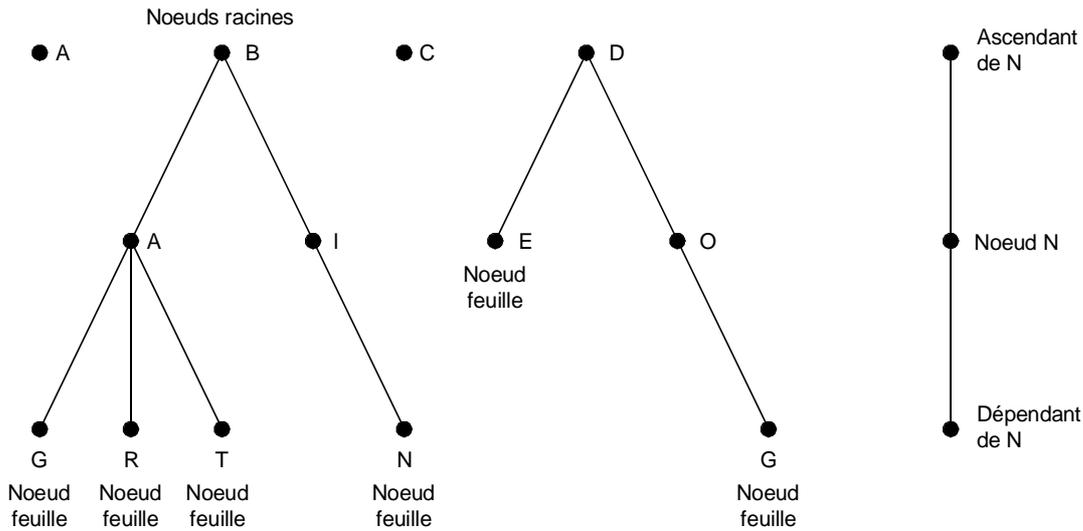
Le dictionnaire utilisé dans le stockage des chaînes de caractères visées par les opérations de codage et de décodage peut être représenté logiquement par une structure de données arbitraires. Le dictionnaire peut être considéré comme contenant un ensemble d'arbres, comme indiqué à la figure 2/V.42 bis, dont chacun a une racine correspondant à un caractère de l'alphabet. Avec le format à 8 bits, le dictionnaire comporte 256 arbres.

Un arbre représente l'ensemble de chaînes connues commençant par un caractère spécifique, et chaque noeud ou point de cet arbre un élément de cet ensemble. Les arbres de la figure 2/V.42 bis représentent les chaînes A, B, BA, BAG, BAR, BAT, BI, BIN, C, D, DE, DO, DOG.

Un noeud dépourvu de noeuds dépendants, représenté par la plus basse couche hiérarchique de l'arbre, est un noeud feuille. Un noeud feuille représente le dernier caractère de la chaîne.

Un noeud sans ascendant, représenté par la couche hiérarchique la plus élevée de l'arbre, est un noeud racine. Un noeud racine représente le premier caractère d'une chaîne.

A chaque noeud est associé un mot-code qui lui confère une identification exclusive. Les assignations mots-codes effectuées dans le dictionnaire de codage de la fonction de compression des données et les assignations correspondantes du dictionnaire de décodage de la fonction de compression homologue active dans l'ETCD distant sont identiques: le mot-code assure donc la réversibilité du codage de la chaîne.



T17001950-92/d02

FIGURE 2/V.42 bis

Représentation de la structure arborescente du dictionnaire

6.2 Procédure d'initialisation du dictionnaire

A la réception d'une primitive de demande C-INIT adressée par la fonction de commande, la fonction de compression des données remet les dictionnaires du codeur et du décodeur à leur état initial.

A l'état initial, chaque arbre du dictionnaire ne comporte qu'un noeud racine. Le mot-code associé à chaque noeud est N_6 (soit le nombre de mots-codes de commande) plus la valeur ordinale du caractère représenté par le noeud. Le compteur C_1 utilisé dans l'attribution de nouveaux noeuds (voir le § 6.5) est réglé à N_5 .

6.3 Procédure de mise en correspondance des chaînes

Cette procédure assure la mise en correspondance d'une séquence de caractères (chaîne) et d'une entrée dans le dictionnaire. La procédure s'amorce sur un seul caractère, représentant le premier caractère de la chaîne. Elle se déroule ensuite comme suit:

- a) une chaîne est formée à partir du premier caractère;
- b) si la chaîne correspond à une entrée dans le dictionnaire et que l'entrée n'est pas celle qui a été créée lors de la dernière application de la procédure de mise en correspondance de la chaîne, le caractère suivant est lu et ajouté à la chaîne et cette étape est reprise;
- c) si la chaîne ne correspond pas à une entrée dans le dictionnaire ou correspond à l'entrée créée lors de la dernière application de la procédure de mise en correspondance de la chaîne, le dernier caractère ajouté à la chaîne doit être enlevé. La chaîne ainsi raccourcie représente la plus longue chaîne de correspondance et le dernier caractère de cette chaîne est le caractère sans correspondance.

Cette procédure met normalement en concordance la plus longue chaîne de caractères. Toutefois, il est deux cas dans lesquels il sera mis fin à l'étape b) avant que la mise en correspondance la plus longue ne soit trouvée.

- i) si un état d'exception se présente, tel qu'une primitive de demande C-INIT ou une primitive de demande C-FLUSH (dans le mode compression seulement);
- ii) en cas de transition entre les modes compression et transparent.

En mode transparent, le codeur utilisera uniquement les critères spécifiés ci-dessus pour mettre fin à la procédure de mise en correspondance de la chaîne. Toutefois, dans le mode compression, il peut employer d'autres critères pour mettre fin à la procédure (par exemple, une temporisation).

Si la procédure de mise en correspondance de la chaîne est terminée avant que la correspondance la plus longue ait été trouvée, le caractère suivant provenant de l'ETTD sera considéré comme étant le «caractère sans correspondance» pour la mise à jour du dictionnaire et la réinitialisation de la procédure de mise en correspondance de la chaîne.

6.4 *Procédure d'adjonction de chaînes au dictionnaire*

Pour maintenir une compression efficace, on adapte le dictionnaire par l'adjonction de nouvelles chaînes. Une nouvelle chaîne est constituée par rattachement d'un caractère unique à une chaîne existante, ce qui a pour effet d'ajouter un nouveau noeud à l'arbre. Le caractère unique est le caractère sans correspondance des chaînes ou encore le préfixe résultant du décodage de la chaîne. Conformément à cette procédure, le seul caractère nécessaire pour réinitialiser la procédure de mise en correspondance de chaînes sera le caractère sans correspondance.

Dans les deux cas suivants, de nouvelles chaînes ne peuvent pas être ajoutées au dictionnaire:

- a) lorsqu'il en résulterait un dépassement de la valeur maximale N_7 de la longueur de chaîne;
- b) lorsque la chaîne en question existe déjà dans le dictionnaire.

Immédiatement après la création d'une entrée de dictionnaire, la procédure de récupération d'entrées dans le dictionnaire sera appliquée.

6.5 *Procédure de récupération d'entrées dans le dictionnaire*

La présente section définit une procédure systématique de récupération d'entrées dans le dictionnaire qui permet de les réutiliser lorsque toutes les entrées disponibles ont été remplies. Lorsque la dernière entrée disponible dans le dictionnaire a été assignée, la présente procédure permet de récupérer une entrée unique, tout en maintenant l'association entre l'entrée vide et son mot-code.

Un compteur C_1 indique le prochain mot-code associé à la prochaine entrée de dictionnaire vide; sa valeur est maintenue dans la fourchette comprise entre N_5 et $N_2 - 1$. Initialement, le compteur C_1 doit être réglé à N_5 .

Cette procédure ne doit être appliquée qu'après la création d'une nouvelle entrée de dictionnaire et comprend les étapes suivantes:

- a) le compteur C_1 est incrémenté;
- b) si la valeur de C_1 dépasse $N_2 - 1$, C_1 est mis à N_5 ;
- c) si le noeud identifié par le mot-code ayant une valeur C_1 est utilisé et n'est pas un noeud feuille, passer à l'étape a);
- d) si le noeud est un noeud feuille, il convient de le détacher de son ascendant.

7 **Fonction de codage**

7.1 *Considérations générales*

La fonction de codage comporte cinq opérations principales:

- a) mise en correspondance des chaînes: une séquence de caractères à la sortie de l'ETTD est mise en correspondance avec une entrée de dictionnaire (voir le § 7.3);
- b) codage: le mot-code de l'entrée de dictionnaire mise en correspondance est représenté comme une valeur binaire d'une longueur égale à C_2 bits (voir le § 7.4);
- c) transmission: transfert du(des) mot(s)-code(s), en mode compression, ou des caractères, en mode transparent, à la fonction de commande (voir le § 7.5);
- d) mise à jour du dictionnaire: création d'une nouvelle entrée de dictionnaire comportant l'entrée mise en correspondance et le caractère sans correspondance (voir le § 7.6);

- e) assignation d'entrée: récupération d'une entrée de dictionnaire en vue de la prochaine mise à jour (voir le § 7.7).

La fonction de codage s'effectue dans l'un des deux modes, à savoir le mode transparent et le mode compression et passe de l'un à l'autre de ces modes sur la base de l'estimation définie en f) ci-dessous. La séquence des opérations et le cycle du caractère d'échappement (voir le § 9) sont identiques pour les deux modes d'exploitation.

Le codeur assurera deux autres opérations, qui ne seront exécutées que pendant la procédure de mise en correspondance de la chaîne, conformément au § 6.3:

- f) vérification de compressibilité des données: estimation de l'efficacité du codage et choix du mode transparent ou du mode compression pour augmenter l'efficacité (voir le § 7.8);
- g) flush: une demande C-FLUSH provenant de la fonction de commande indique que toutes les données restantes seront envoyées (voir le § 7.9).

7.2 Conditions initiales

A la réception d'une demande C-INIT, la fonction de compression des données initialise le codeur à l'état suivant:

- a) le dictionnaire doit être mis dans l'état initial décrit au § 6.2;
- b) la longueur C_2 du mot-code est fixée à $N_3 + 1$;
- c) le seuil C_3 est fixé à $N_4 \times 2$;
- d) la fonction est réglée sur le mode «transparent»;
- e) une valeur ordinaire 0 est affectée au caractère d'échappement.

7.3 Mise en concordance des chaînes

A la réception d'une demande C-DATA, la fonction de compression des données applique la procédure de mise en concordance des chaînes définie au § 6.3. Le caractère initial requis est le caractère sans correspondance résultant de l'application la plus récente de cette procédure.

7.4 Codage

Cette procédure est utilisée dans le mode compression. Elle vise à représenter le mot-code comme une séquence de bits C_2 ; l'ordre et le numérotage des bits sont indiqués dans la figure 3/V.42 bis.

Si le mot-code qui correspond à l'entrée du dictionnaire mise en correspondance est numériquement égal ou supérieur au seuil C_3 :

- a) le mot-code de commande STEPUP sera codé et transféré à l'aide de la longueur du mot-code actuelle (C_2);
- b) la longueur du mot-code, C_2 , sera augmentée de 1;
- c) C_3 sera multipliée par 2;
- d) si le mot-code est encore numériquement supérieur ou égal à C_3 , les étapes a) à c) seront répétées.

Le mot-code est ensuite transmis à la fonction de commande, conformément aux procédures définies au § 7.5.

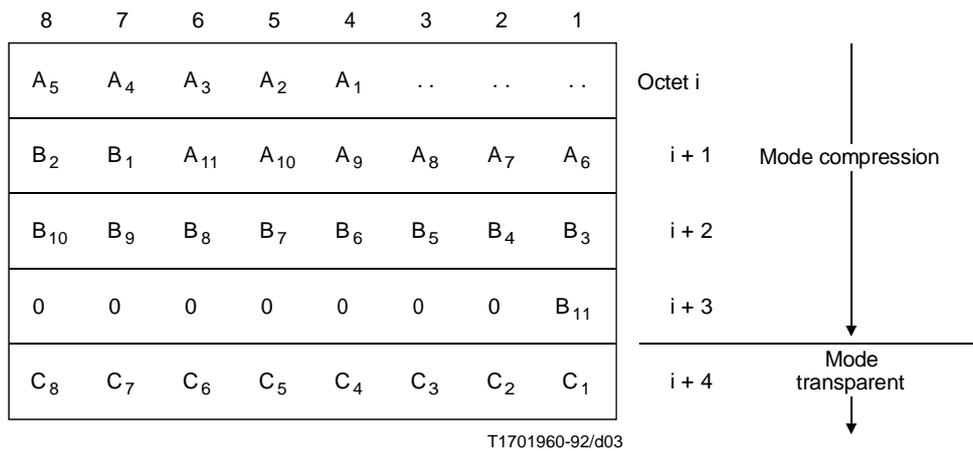


FIGURE 3/V.42 bis

Conversion des mots-codes en octets

7.5 *Transmission*

En mode transparent, les caractères sont adressés à la fonction de correction d'erreur pour transmission sous forme d'octets alignés à l'aide d'une indication C-TRANSFER. Ils peuvent être transmis individuellement pendant la procédure de mise en correspondance de la chaîne, ou en séquence une fois que la procédure de mise en correspondance de la chaîne est terminée.

Dans le mode compression, la chaîne mise en correspondance sera codée conformément à la procédure définie au § 7.4 et adressée à la fonction de correction d'erreur sous forme de paquets, le bit de plus faible poids d'un mot-code suivant immédiatement le bit de plus fort poids du mot-code précédent.

Lorsque le codeur passe du mode transparent au mode compression, le bit de plus faible poids du premier mot-code à transmettre est aligné sur le bit 1 de la prochaine position d'octet disponible.

A la suite de la transmission d'un mot-code de commande FLUSH, ou lorsque le codeur passe du mode compression au mode transparent (voir le § 9) après transmission du mot-code de commande PMT (passage au mode transparent) de la séquence, la transmission d'un nombre suffisant de zéros binaires assure l'alignement en octets du prochain caractère transmis.

La figure 3/V.42 bis illustre le transfert du train de données à la fonction de correction d'erreur pendant un changement de mode dans le sens compression-transparent. Deux mots-codes A et B, de 11 bits, sont transmis sous forme comprimée, puis le système passe en mode transparent. Pour ce faire, il faut transmettre sept zéros binaires qui assurent l'alignement en octets du premier caractère non comprimé envoyé en mode transparent, C.

7.6 *Mise à jour du dictionnaire*

Un nouveau noeud de dictionnaire est créé à partir de la chaîne mise en correspondance et du caractère sans correspondance associé renvoyé par la procédure de mise en correspondance des chaînes, selon les modalités définies au § 6.4.

7.7 *Récupération des noeuds*

Après la création d'un nouveau noeud de dictionnaire, la procédure de récupération des noeuds définie au § 6.5 est appliquée.

7.8 *Essai de compressibilité des données*

La fonction de compression des données effectue périodiquement un essai pour vérifier la compressibilité des données. La nature de cet essai n'est pas spécifiée dans la présente Recommandation mais cet essai pourrait consister en une comparaison des nombres de bits nécessaires pour représenter un segment du train de données avant et après la compression.

7.8.1 *Passage au mode de compression*

Lorsque la fonction de compression des données est en mode transparent et qu'elle établit que la compression des données serait efficace, le système:

- a) applique la procédure de mise à jour du dictionnaire en utilisant la chaîne de caractères déjà mémorisée et le caractère suivant à traiter au moyen de la procédure de mise en correspondance des chaînes de caractères (qui sera le premier caractère de la chaîne représenté par le premier mot-code transmis en mode compression);
- b) indique à la fonction de compression des données homologue qu'un passage au mode compression est nécessaire en utilisant la séquence de commande PMC (passage au mode compression) en mode transparent (voir le § 9.1);
- c) effectue le passage au mode compression.

7.8.2 *Passage au mode transparent*

Lorsque la fonction de compression des données est en mode compression et que l'équipement établit que le train de données n'est pas actuellement compressible, le système:

- a) s'assure que le mot-code représentant toute donnée partiellement codée a été transféré conformément à la procédure décrite aux § 7.4 et 7.5;
- b) applique la procédure de mise à jour du dictionnaire en utilisant la chaîne de caractères déjà mémorisée et le caractère suivant à traiter au moyen de la procédure de mise en correspondance des chaînes de caractères (qui sera le premier caractère de la chaîne transmis en mode transparent);
- c) indique à la fonction de compression des données de l'autre extrémité, par transmission du mot-code de commande PMT (passage au mode transparent) (voir le § 9) un passage au mode transparent;
- d) émet un nombre suffisant de zéros binaires pour rétablir l'alignement des octets (voir le § 7.5);
- e) effectue le passage à l'état transparent.

7.8.3 *Fonction RESET*

En mode transparent, le code de commande RESET est utilisé pour indiquer à la fonction de compression de données homologue que le dictionnaire du codeur est sur le point d'être réinitialisé conformément aux procédures indiquées aux § 6.2 et 7.2. Le code de commande RESET est transmis à l'aide de la valeur du caractère d'échappement avant que la réinitialisation n'ait eu lieu.

Les circonstances dans lesquelles le codeur demande une réinitialisation du dictionnaire ne sont pas définies dans la présente Recommandation mais tiennent généralement au fait que le codeur établit qu'une réinitialisation du dictionnaire améliorerait la qualité de fonctionnement. Les procédures de demande de réinitialisation du dictionnaire lors de l'établissement de la liaison ou de la détection par la fonction de commande d'un état d'erreur sont définies aux § 5.2 et 5.6.

Le code de commande RESET n'est pas transmis lorsque la fonction de commande envoie une demande C-INIT.

7.9 *Action à la réception d'une demande C-FLUSH*

Lorsque le système reçoit de la fonction de commande une demande C-FLUSH, si le codeur est en mode compression et si une chaîne partiellement mise en correspondance est en cours de traitement, la fonction de compression des données:

- a) s'assure que le mot-code représentant toute chaîne partiellement mise en correspondance est transféré en accord avec les procédures décrites aux § 7.4 et 7.5;

- b) applique la procédure de mise à jour du dictionnaire en utilisant la chaîne de caractères déjà mémorisée et le caractère suivant (si disponible) à traiter au moyen de la procédure de mise en correspondance des chaînes de caractères;
- c) si l'étape a) laisse quelques bits en suspens pour la transmission (alignement d'octet non encore réalisé):
 - i) transfère le mot-code FLUSH (voir le § 9);
 - ii) transfère si nécessaire un nombre suffisant de zéros binaires pour rétablir l'alignement des octets (voir le § 7.5).

Si le codeur est en mode transparent, lorsque le système reçoit de la fonction de commande une demande C-FLUSH, la fonction de compression des données transfère toutes les données restantes; la procédure de mise en correspondance des chaînes n'est pas achevée et la mise à jour du dictionnaire ne s'effectue pas.

8 Mise en application de la fonction de décodage

La fonction de décodage doit pouvoir fonctionner en mode compression comme en mode transparent et fonctionne d'une manière conforme aux indications des § 6, 7 et 9.

A la réception de la demande C-INIT de la fonction de commande ou d'un code de commande RESET de la fonction de compression des données, cette dernière initialise la fonction de décodage conformément aux procédures définies aux § 6.2 et 7.2.

En mode transparent, la fonction de décodage applique la procédure de mise en concordance des chaînes indiquée au § 6.3 afin que le dictionnaire du décodeur puisse être maintenu dans un état compatible avec le dictionnaire du codeur (distant). A la réception d'un code de commande PMC ou EDD, la fonction de décodage est effectuée conformément aux opérations de codage définies aux § 7.8.1 et 9.2. De nouvelles entrées de dictionnaire sont créées d'une manière conforme aux procédures définies aux § 6.4 et 7.3.

En mode compression, la fonction de décodage récupère les chaînes codées. Lorsqu'il reçoit les mots-codes PMT ou FLUSH, le décodeur procède d'une manière conforme aux opérations du codeur définies aux § 7.8.2 et 7.9. De nouvelles entrées de dictionnaire sont créées par application de la procédure définie au § 6.4, le premier caractère (préfixe) de la chaîne la plus récemment décodée étant ajouté à la chaîne précédemment décodée.

Le décodeur perçoit le mot-code de commande STEPUP comme une indication de l'augmentation, par le codeur, de la longueur du mot-code en accord avec les procédures définies au § 7.4.

9 Communications entre les fonctions de compression des données de chaque extrémité

9.1 Mots-codes de commande et codes de commande

Les mots-codes de commande et les codes de commande attribués à la communication entre fonctions de compression des données homologues sont donnés dans le tableau 2/V.42 bis.

TABLEAU 2/V.42 bis

Mots-codes de commande (employés en mode compression)		
Mot-code	Désignation	Description
0	PMT	Passage au mode transparent
1	FLUSH	Transfert des données restantes
2	STEPUP	Augmentation de la longueur du mot-code
Codes de commande (employés en mode transparent)		
Valeur	Désignation	Description
0	PMC	Passage au mode compression
1	EDD	Caractère d'échappement en données
2	RESET	Réinitialisation sous contrainte
3 à 255	Réservé	

9.2 Procédure pour l'utilisation de la séquence d'échappement

Une séquence de commande en mode transparent se compose du caractère d'échappement suivi de l'un des codes de commande énumérés au § 9.1 ci-dessus.

Pour limiter l'extension des données résultant du mécanisme d'échappement défini ci-dessous, si le caractère d'échappement en cours est décelé dans le train de données en provenance de l'ETTD, la fonction de compression des données:

- en mode transparent, transmet le caractère d'échappement décelé, émet le code EDD; puis,
- en modes transparent et compression, modifie la valeur du caractère d'échappement en lui ajoutant la valeur décimale 51, adjonction à effectuer modulo 256.

10 Paramètres

Les paramètres suivants sont nécessaires à la fonction de compression des données. Les paramètres N_1 à N_7 et P_0 à P_2 s'appliquent dans les deux sens de transmission, tandis qu'un ensemble séparé de variables C_1 , C_2 et C_3 doit être prévu dans le codeur et dans le décodeur.

N_1 Longueur maximale du mot-code (bits).

N_2 Nombre total d'entrées de mots-codes.

N_3 Longueur des caractères (bits):

$$N_3 = 8.$$

N_4 Nombre de caractères présents dans l'alphabet:

$$N_4 = 2^{N_3}.$$

N_5 Numéro d'indice de la première entrée de dictionnaire utilisée dans le stockage d'une chaîne de caractères:

$$N_5 = N_4 + N_6.$$

N_6 Nombre de mots-codes de commande:

$$N_6 = 3.$$

N_7 Longueur maximale d'une chaîne de caractères.

C_1 Prochaine entrée de dictionnaire vide.

- C₂ Longueur du mot-code courant.
- C₃ Valeur seuil de déclenchement de la modification du mot-code.
- P₀ Demande de compression de données (V.42 *bis*).
- P₁ Nombre de mots-codes (paramètre de négociation).
- P₂ Taille maximale de la chaîne (paramètre de négociation).

ANNEXE A

(à la Recommandation V.42 *bis*)

Procédures de négociation V.42 *bis* en cas d'utilisation concurrentement aux procédures V.42

Lorsque la présente Recommandation sur la compression des données est utilisée avec la correction d'erreur de type V.42, la procédure de négociation XID est appliquée. (Voir les § 7.6, 8.10 et 10 de la Recommandation V.42 et la norme ISO 8885 — 1987 [6, 7].) On utilise à cette fin un sous-champ de couche liaison de données en plus des sous-champs définis dans la Recommandation V.42. Il doit apparaître dans la trame XID immédiatement avant le sous-champ des données d'usager et être codé comme indiqué dans le tableau A-1/V.42 *bis*.

Pendant la phase d'établissement du protocole, la présence du paramètre P₀ dans le sous-champ de couche liaison de données de l'ensemble de paramètres privés de la trame XID indique une demande de compression de données.

Remarque — L'incorporation de la présente annexe dans la Recommandation V.42 est en cours d'examen par la Commission d'études XVII.

TABLEAU A-1/V.42 bis

	Bit 8 1	
ID de Groupe	11110000	Ensemble de paramètres privés (ISO 8885, Addendum 3)
Longueur de Groupe	nnnnnnnn	(MSB) Longueur du champ de paramètres (exclut l'ID de groupe et la longueur)
	nnnnnnnn	(LSB)
ID de paramètre	00000000	Identificateur de l'ensemble de paramètres
Longueur de paramètre	00000011	Longueur de la chaîne
Valeur de paramètre	01010110	V
	00110100	4
	00110010	2
ID de paramètre	00000001	Demande de compression des données (P ₀) Rec. V.42 bis
Longueur de paramètre	00000001	Longueur du champ
Valeur de paramètre	000000nn	Demande de compression dans:
		00 aucune direction
		01 direction négociation entité appelante-entité appelée seulement
		10 direction négociation entité appelée entité appelante seulement
		11 les deux directions
ID de paramètre	00000010	Nombre de mots code (P ₁) Rec.V.42 bis
Longueur de paramètre	00000010	Nombre entier à 16 bits
Valeur de paramètre	nnnnnnnn	(MSB) Valeur du paramètre P ₁
	nnnnnnnn	(LSB)
ID de paramètre	00000011	Longueur maximale de la chaîne (P ₂) Rec. V.42 bis
Longueur de paramètre	00000001	Nombre entier à 8 bits
Valeur de paramètre	nnnnnnnn	Valeur de paramètre P ₂

MSB: Bit de plus fort poids

LSB: Bit de plus faible poids

APPENDICE I

(à la Recommandation V.42 bis)

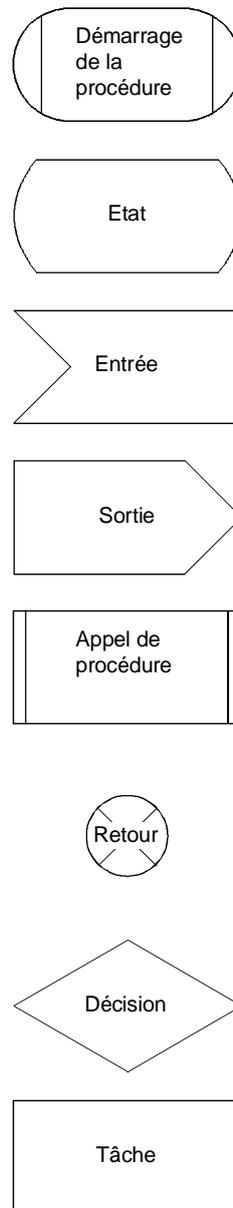
Description du codeur en LSD (Z.100 à Z.104) [8]

La figure I-1/V.42 bis contient les symboles LSD utilisés dans les diagrammes du présent appendice.

Les schémas qui suivent illustrent le fonctionnement du codeur:

- Figure I-2/V.42 bis: Codeur (voir les § 7.2, 7.3, 7.8, 7.9). Ce schéma illustre le fonctionnement des principaux éléments du codeur.
- Figure I-3/V.42 bis: Procédure de traitement des caractères (voir les § 6.3, 6.4, 6.5). Ce schéma illustre le fonctionnement de la procédure de mise en correspondance des chaînes, des conditions dans lesquelles il est mis fin à cette procédure, et les mesures prises.

- c) Figure I-4/V.42 *bis*: Vérification de la longueur du mot-code (voir le § 7.4). Ce schéma illustre le mécanisme d'augmentation de la longueur du mot-code.
- d) Figure I-5/V.42 *bis*: Estimation de compression (voir le § 7.8). Ce schéma illustre les procédures de passage entre le mode transparent et le mode compression, et celle relative à l'utilisation de la commande de réinitialisation (RESET).
- e) Figure I-6/V.42 *bis*: Transfert des données restantes (voir le § 7.9). Ce schéma illustre la procédure appliquée à la réception d'une demande C-FLUSH.
- f) Figure I-7/V.42 *bis*: Procédure de traitement exceptionnel du caractère suivant [voir les § 7.8.1 a), 7.8.2 b), 7.9 b)]. Ce schéma illustre la façon dont s'effectue le traitement des caractères en mode compression après un passage au mode compression ou après une élimination.
- g) Figure I-8/V.42 *bis*: Procédure de caractère d'échappement (voir le § 9.2).
- h) Figure I-9/V.42 *bis*: Procédure de réinitialisation du signal (voir le § 7.8.3).
- i) Figure I-10/V.42 *bis*: Procédure d'adjonction «chaîne + caractère» au dictionnaire (voir les § 6.4 et 6.5).



T1701970-92/d04

FIGURE I-1/V.42 *bis*
Symboles LSD utilisés

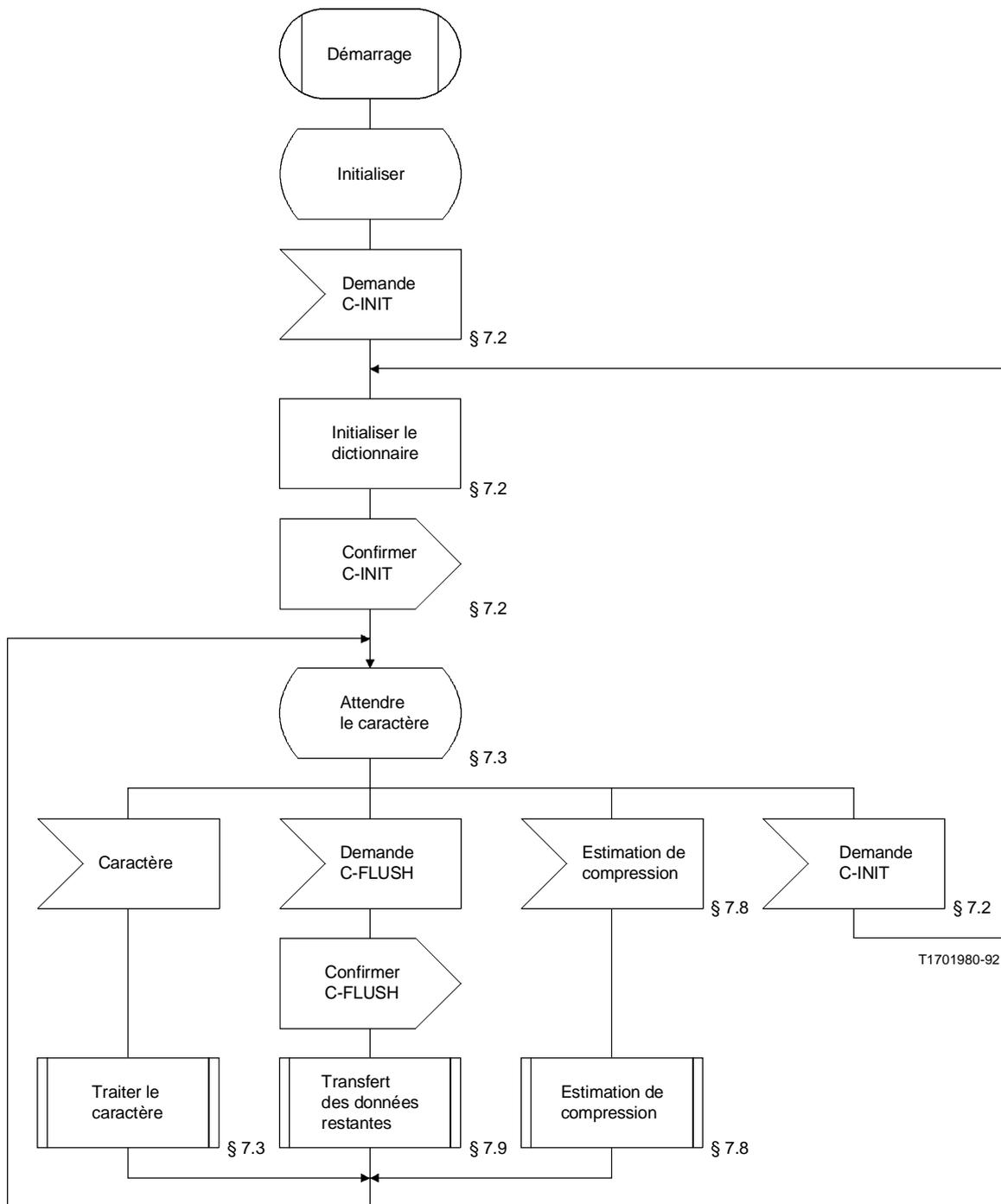
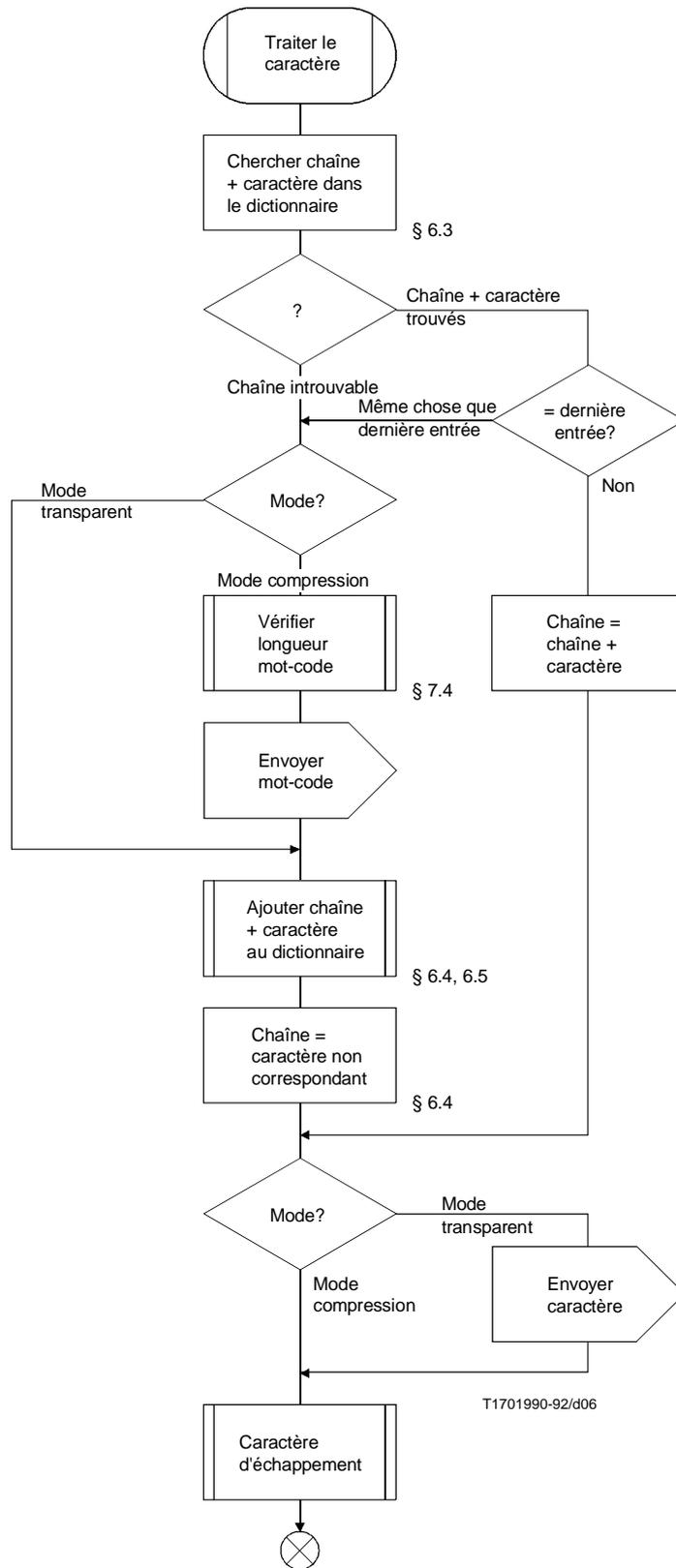


FIGURE I-2/V.42 bis

Codeur



T1701990-92/d06

FIGURE I-3/V.42 bis

Procédure de traitement des caractères

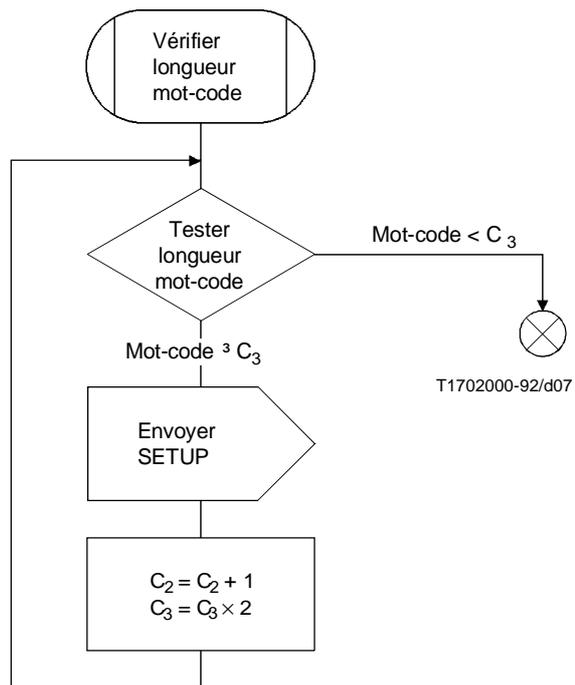
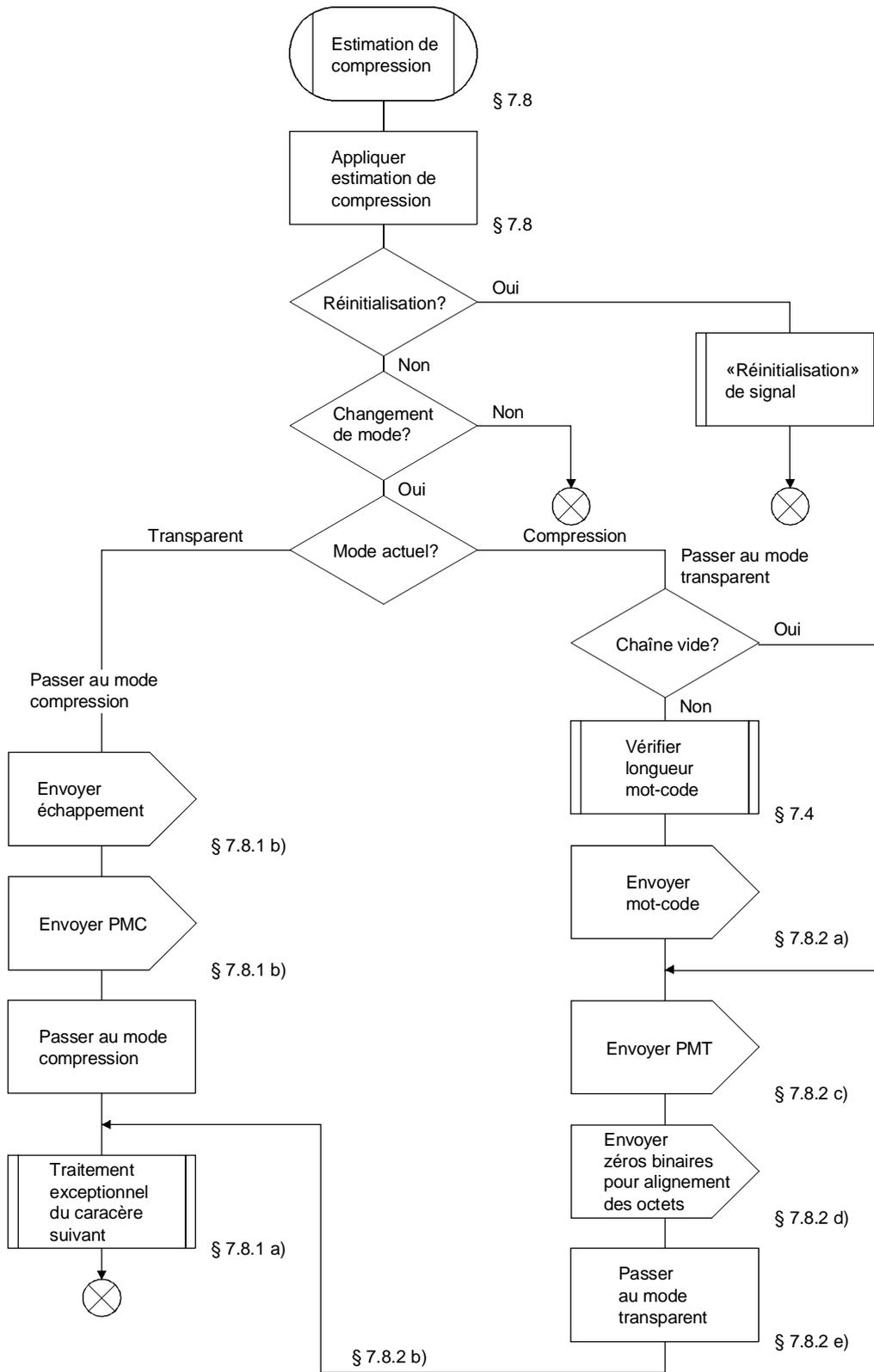


FIGURE I-4/V.42 bis

Procédure de vérification de la longueur du mot-code



T1702010-92/d08

FIGURE I-5/V.42 bis

Procédure d'essai de compression

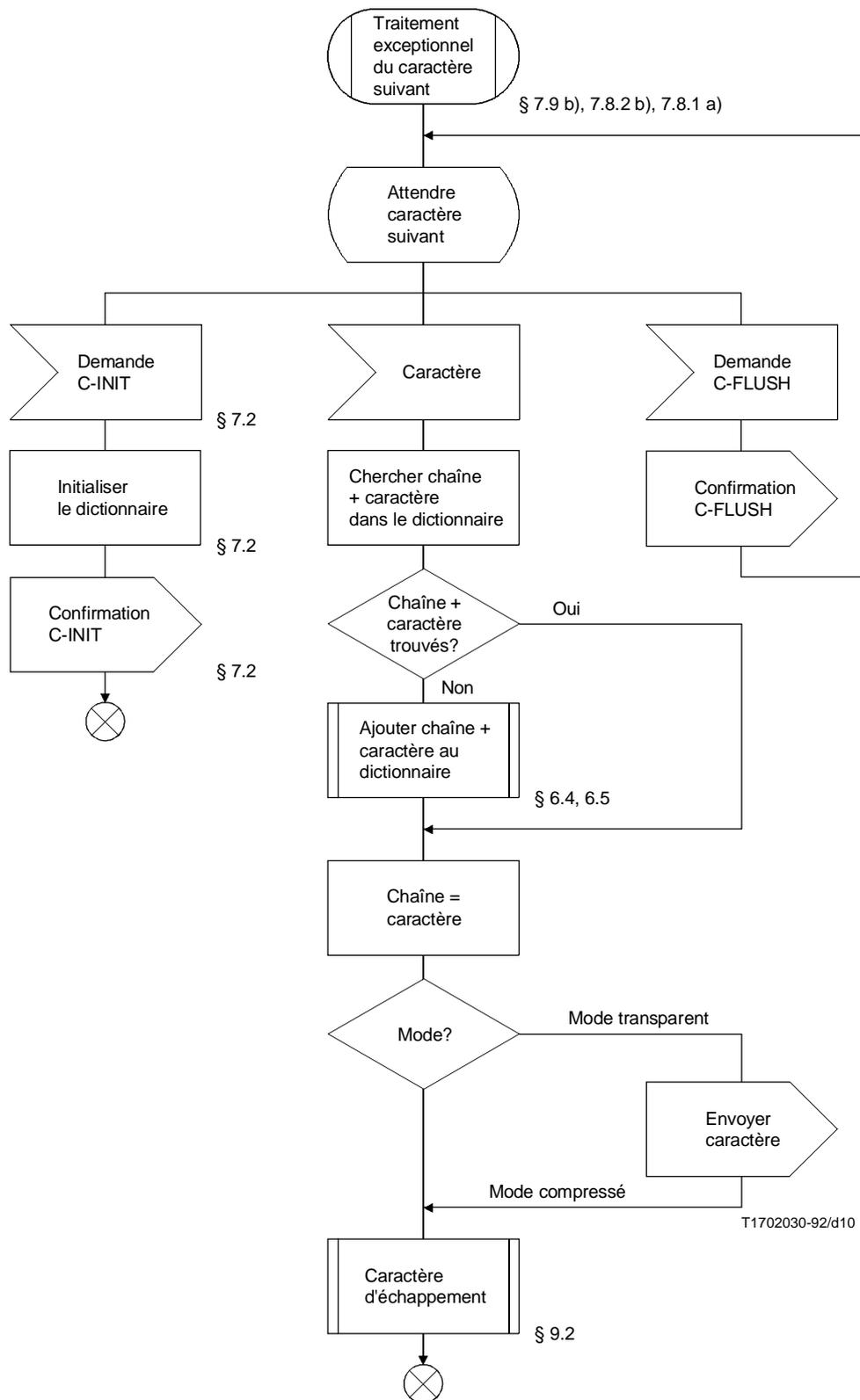


FIGURE I-7/V.42 bis

Procédure de traitement exceptionnel du caractère suivant

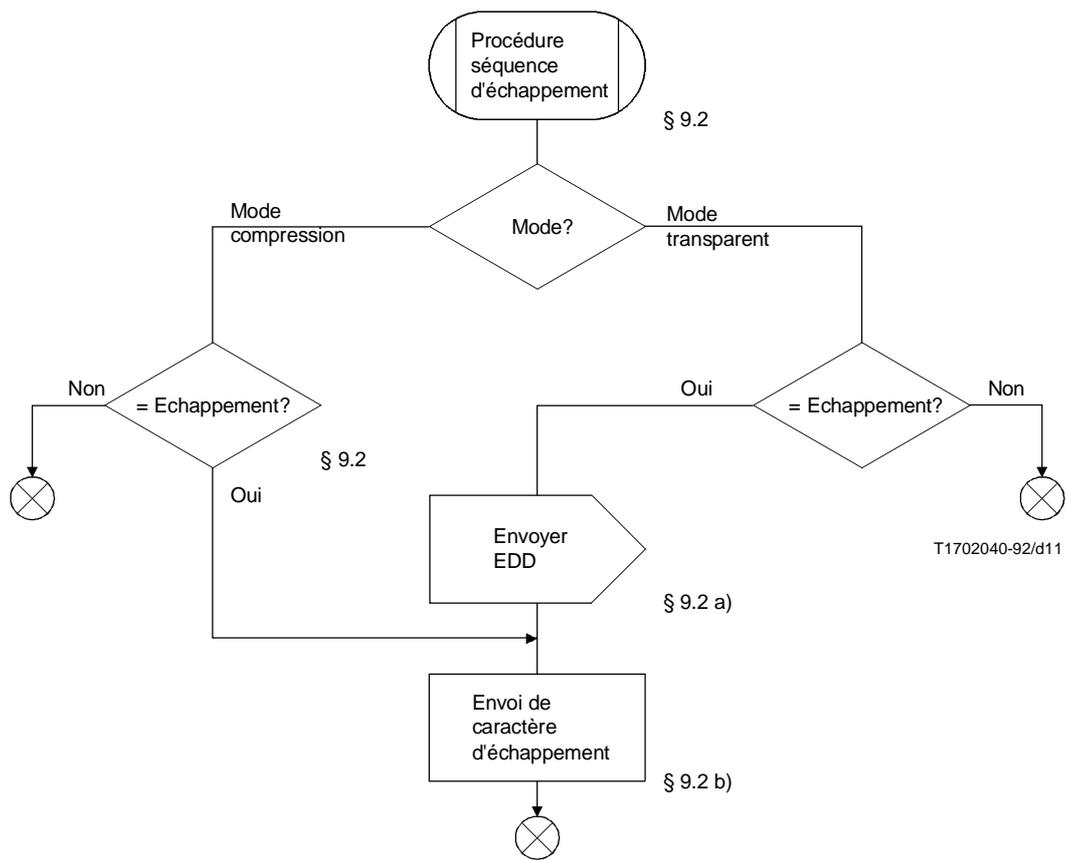


FIGURE I-8/V.42 bis

Procédure d'utilisation de la séquence d'échappement

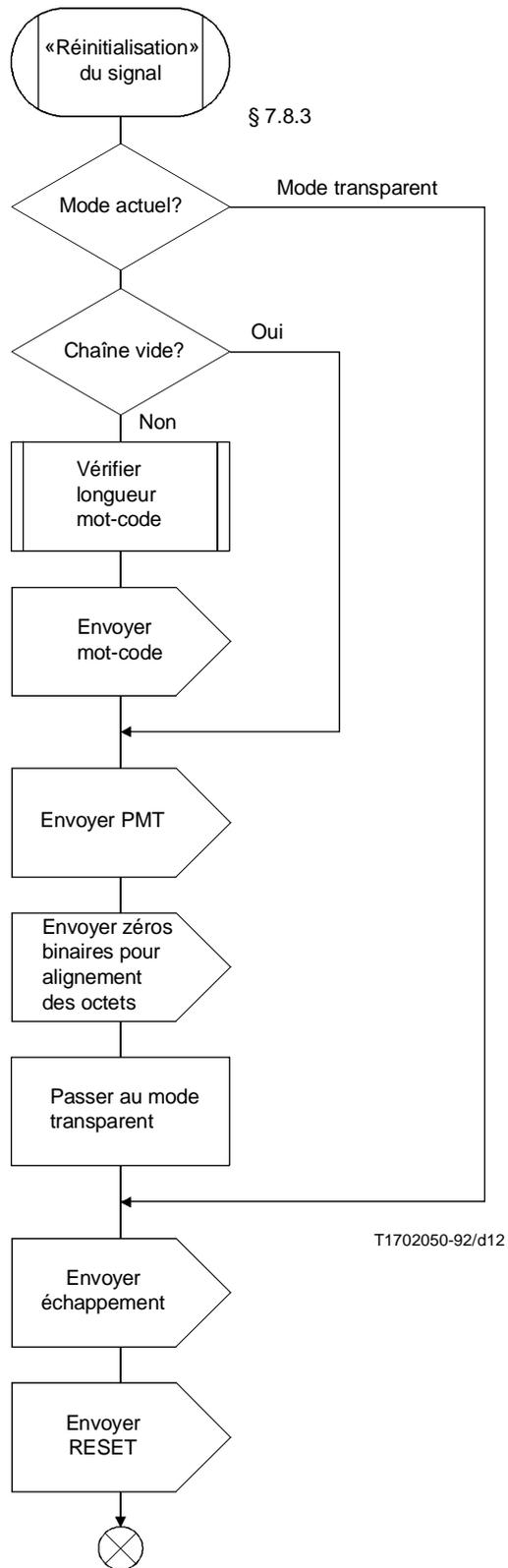


FIGURE I-9/V.42 bis

Procédure de «réinitialisation» du signal

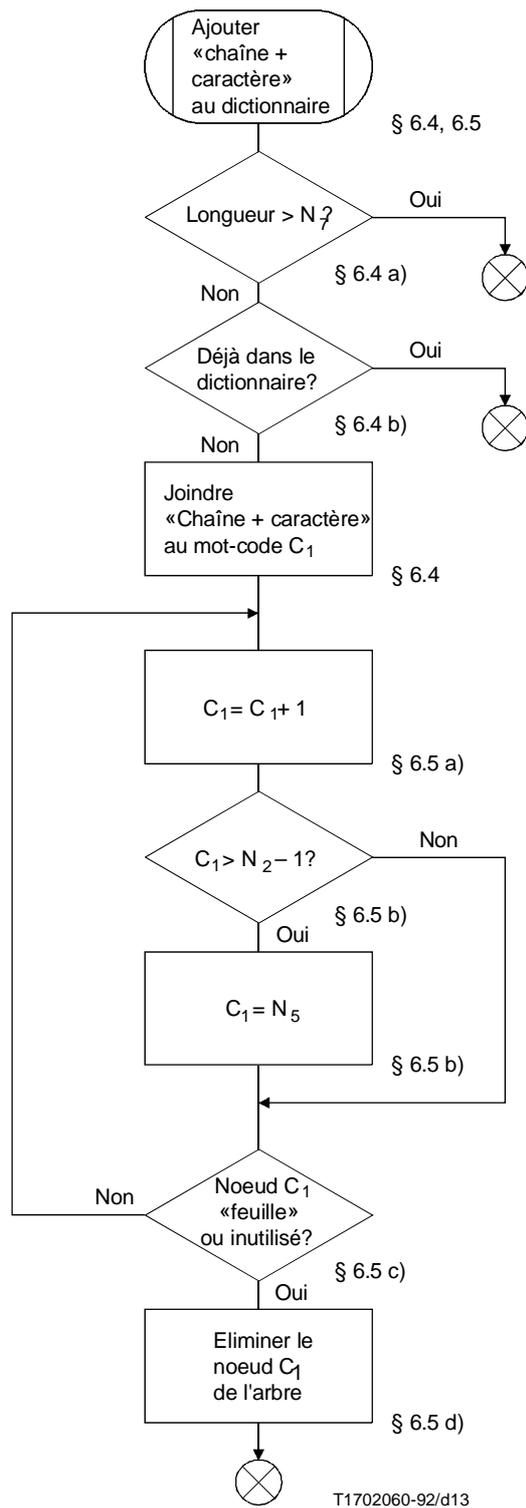


FIGURE I-10/V.42 bis

Procédure d'adjonction «chaîne + caractère» au dictionnaire

APPENDICE II

(à la Recommandation V.42 bis)

Directives pour la mise en oeuvre

Les notes qui suivent contiennent des renseignements sur la mise en oeuvre du plan de compression des données et sur la sélection des paramètres.

II.1 *Sélection de N_2 , le nombre de mots-codes*

La dimension du dictionnaire est égal à $N_2 - N_6$ (en admettant que des entrées ne sont pas prévues pour les mots-codes réservés). Le choix d'une valeur élevée pour N_2 signifie qu'il existe un grand nombre de chaînes disponibles mais aussi que la valeur de N_1 est supérieure. Le gain de qualité de fonctionnement obtenu par la sélection d'un dictionnaire plus grand peut être neutralisé par la dimension plus grande des mots-codes nécessaires et, pour certains types de données, on peut obtenir une meilleure qualité de fonctionnement en utilisant un dictionnaire plus petit. Si l'on choisit pour N_2 des valeurs dans la gamme $2^n + 1$ (n étant un nombre entier) jusqu'à approximativement $1,3 \times 2^n$, il n'y aura pas d'amélioration de la qualité de fonctionnement par rapport au choix de la valeur 2^n . Une valeur de 2048 pour N_2 donne de bons résultats de compression dans un large éventail de types de données.

II.2 *Structure des données*

Le plan de compression des données décrit dans la présente Recommandation est bien adapté à la mise en oeuvre utilisant une structure de données en arbre. Ce type de structure de données offrira une bonne utilisation de l'espace de mémoire et une recherche rapide.

II.3 *Calcul de la performance de compression*

Le calcul de la performance de compression peut être exprimé comme le nombre de caractères reçus par un codeur divisé par le nombre d'octets transférés à partir de ce codeur (à la fonction de limitation d'erreur). Le comptage des caractères et des octets doit être mis à zéro à la réception d'une demande C-INIT.

II.4 *Exemples de l'application du codeur*

Les trois exemples suivants illustrent l'application du codeur. On admet que le dictionnaire se trouve dans l'état indiqué à la figure 2/V.42 bis:

II.4.1 *Cas simple: «BAY» en mode compression*

L'équipement lit le premier caractère «B» et recherche dans le dictionnaire la chaîne «B». Cette chaîne étant présente, l'équipement lit le caractère suivant, «A» et l'ajoute au précédent, formant la nouvelle chaîne «BA». Il cherche dans le dictionnaire la nouvelle chaîne et quand il la trouve, lit le caractère suivant «Y» et l'ajoute aux précédents, formant la nouvelle chaîne «BAY». L'équipement cherche dans le dictionnaire «BAY», qui ne s'y trouve pas. Il élimine «Y» et la procédure de mise en concordance de chaîne se termine avec «BA» comme chaîne de concordance et «Y» comme caractère sans concordance.

Le mot-code pour «BA» est codé en C_2 bits, mis en octets et adressé à la fonction de commande pour transmission. La nouvelle chaîne «BAY» est créée par l'adjonction de «Y» à «BA», par l'affectation d'un mot-code de valeur C_1 à cette nouvelle chaîne. C_1 est incrémenté et le noeud (chaîne) auquel cette valeur est actuellement affectée est mis à l'essai pour vérifier s'il s'agit d'un noeud vide ou d'un noeud feuille. Si le noeud est vide, il sera utilisé dans la mise à jour du dictionnaire suivant. Si le noeud est utilisé et n'est pas un noeud feuille, c'est-à-dire qu'il fait partie d'une chaîne plus longue, C_1 est incrémenté à nouveau et l'essai est répété. Si le noeud est un noeud feuille, il est détaché de son ascendant et sera réutilisé dans la prochaine mise à jour du dictionnaire.

Le caractère «Y» sera utilisé pour faire redémarrer la mise en concordance de chaîne.

II.4.2 Cas simple: «BAY» en mode transparent

En mode transparent, la même séquence d'opération décrite dans le § II.4.1 se produira, la seule différence étant que les caractères «A» et «Y» seront transmis à la place du mot-code pour «BA».

II.4.3 Caractères ou séquences répétés: «CCCC» en mode compression

Le but de cet exemple est d'illustrer un aspect particulier de l'algorithme. Comme le codeur est capable de mettre à jour son dictionnaire de façon prévisionnelle, tandis que le décodeur peut seulement mettre à jour son dictionnaire sur la base de données précédemment décodées, il est nécessaire de s'assurer que le codeur n'utilise pas de nouvelles entrées de dictionnaire avant qu'elles soient transmises au décodeur.

L'équipement lit le premier «C» et le met en concordance avec l'entrée de dictionnaire pour «C». Il lit le second «C», ajouté au premier et cherche dans le dictionnaire «CC». Comme «CC» ne se trouve pas dans le dictionnaire, la procédure de mise en concordance de chaîne se termine avec la chaîne de concordance «C» et le caractère sans concordance «C». «CC» est ajouté au dictionnaire, le mot-code pour «C» est envoyé et la mise en concordance de chaîne reprend avec le second «C».

L'équipement lit le troisième «C», l'ajoute au second «C», formant «CC» et cherche dans le dictionnaire «CC». Comme celui-ci se trouve dans le dictionnaire mais est à l'entrée créée depuis la dernière mise en concordance de chaîne [voir le § 6.3b)], la procédure de mise en concordance de chaîne se termine avec la chaîne en concordance de «C» et le caractère sans concordance «C». «CC» n'est pas ajouté au dictionnaire car il s'y trouve déjà, le mot-code pour «C» est envoyé et la mise en concordance de chaîne commence avec le troisième «C».

L'équipement lit le quatrième «C», l'ajoute au troisième «C», formant «CC» et cherche dans le dictionnaire «CC». Comme «CC» se trouve dans le dictionnaire et ne correspond pas à l'entrée créée depuis la dernière mise en concordance de chaîne (l'opération de mise à jour a été empêchée), le cinquième «C» est lu et ajouté à la chaîne.

Références

- [1] Recommandation du CCITT *Procédures de correction des erreurs pour les ETCD utilisant la conversion asynchrone/synchrone*, tome VIII, Rec. V.42.
- [2] Recommandation du CCITT *Utilisation par l'intermédiaire d'un RNIS d'un équipement terminal de traitement de données avec des interfaces du type de la série V permettant un multiplexage statistique*, tome VIII, Rec. V.120.
- [3] Norme ISO 3309 — *Communication de données — Procédures de commande de liaison de données à haut niveau — Structure de trame*.
- [4] Recommandation du CCITT *Définitions des termes relatifs aux communications de données sur le réseau téléphonique*, tome VIII, Rec. V.7.
- [5] Recommandation du CCITT *Transmission des caractères arithmiques sur des voies supports synchrones*, tome VIII, Rec. V.14.
- [6] Norme ISO 8885 — 1987: *Systèmes de traitement de l'information — Communication de données — Format et contenu du champ d'information de XID pour application générale*.
- [7] Norme ISO 8885 — 1987/ADD3: *Systèmes de traitement de l'information — Communication de données — Format et contenu du champ d'information de XID pour application générale — Addendum 3: Définition d'un sous-champ de la couche liaison de données pour la négociation de paramètres privés*.
- [8] Recommandation du CCITT de la série Z.100 *Langage de spécification et de description fonctionnelles (LDS)*, tome X.