

Télé-relevé par liaison téléphonique RTC des appareils de comptage de type « Compteur Jaune Electronique »

Identification : Enedis-NOI-CPT_03E**Version :** 3**Nb. de pages :** 62

Version	Date d'application	Nature de la modification	Annule et remplace
1	15/12/2008	Création	
2	01/04/2008	Prise en compte de l'identité visuelle d'ERDF	NOP-RES_104E
3	01/01/2017	Prise en compte de la nouvelle dénomination sociale d'Enedis	ERDF-NOI-CPT_03E

Résumé / Avertissement :

Cette spécification s'applique aux appareils de comptage électroniques de type « Compteur Jaune Electronique » (CJE) assurant le comptage triphasé de l'énergie active et la quantification des dépassements de puissance pour les utilisateurs du Réseau Public de Distribution de niveau de tension BT.

Elle définit les informations qui sont administrées par cet appareil et auxquelles les utilisateurs des réseaux gérés par Enedis peuvent accéder moyennant certaines dispositions de réception de signaux et de traitement informatique appropriés (interfaces de communication et applications de télé-relevé non décrites dans ce document). Cet accès n'est possible que si le compteur est équipé d'un modem raccordé au réseau téléphonique commuté public (RTC). L'utilisateur du réseau ou son mandataire doit donc, en préalable, demander aux services d'Enedis la vérification du raccordement du compteur à un accès téléphonique et les informations sur les conditions de cet accès (code d'accès et horaire d'appel notamment).

SOMMAIRE

1. Généralités	4
1.1. Domaine d'application	4
1.2. Références normatives	5
2. PARTIE 1 - Données télé-relevables par la liaison téléphonique	6
2.1. Définitions concernant le comptage et le télé-relevé	6
2.1.1. Principes généraux sur le télé-relevé des compteurs	6
2.1.2. Vocabulaire	7
2.1.3. La mesure de l'énergie active et les données gérées	10
2.1.4. Le contrôle des dépassements de puissance et les données gérées	10
2.2. Liste des groupes de données télé-relevables et mode d'accès	13
2.3. Contrats (code 0C)	15
2.4. Structure horosaisonnaire (code 0B)	16
2.4.1. Généralités sur la structure horosaisonnaire	16
2.4.2. Format de la table annuelle	17
2.4.3. Format de la table journalière	18
2.5. Tableau des puissances moyennes (code 08)	18
2.5.1. Cas du compteur en version V1	18
2.5.2. Cas du compteur en version V2	19
2.5.3. Informations supplémentaires (valables pour les compteurs V1 et V2)	19
2.5.3.1. Généralités sur les éléments constituant le tableau	19
2.5.3.2. Élément-puissance	20
2.5.3.3. Éléments de datation	21
2.5.3.4. Gestion du tableau et disposition des différents éléments	22
2.6. Informations de la période P en cours (code 02)	23
2.7. Informations des périodes précédentes P-1 et P-2 (code 01)	24
2.8. Liste des appels téléphoniques (code 07)	26
2.9. Valeurs de référence (code 05)	27
3. PARTIE 2 - Protocoles des couches de communication de la liaison téléphonique	28
3.1. Généralités sur les communications du télé-relevé	28
3.1.1. Structure des échanges	28
3.1.2. Vocabulaire de base et langage de spécification Élément-puissance	28
3.2. Couche Physique	30
3.2.1. Généralités sur la Couche Physique	30
3.2.2. Description de l'environnement V24/V28	30
3.2.3. Protocole de la couche Physique	31
3.2.4. Diagramme des temps	32
3.2.5. Services et primitives de service de la couche Physique	33
3.2.6. Paramètres de la couche Physique	33
3.2.7. Transitions d'état	33
3.2.8. Répertoire et traitement des erreurs	38
3.3. Couche Liaison	39
3.3.1. Généralités sur la Couche Liaison	39
3.3.2. Couche Liaison en mode normal	39
3.3.2.1. Protocole Liaison en mode normal	39



3.3.2.2. Procédure "envoyer et attendre"	39
3.3.2.3. Services et primitives de service de Liaison	40
3.3.2.4. Description des trames	40
3.3.2.5. Gestion des échanges	41
3.3.2.6. Paramètres de Liaison	41
3.3.2.7. Transitions d'état	41
3.3.3. Couche Liaison en mode Echo	47
3.3.3.1. Répertoire et traitement des erreurs	47
3.4. Couche Session	49
3.4.1. Généralités sur la Couche Session	49
3.4.2. Services et primitives de service de Session	49
3.4.3. Description des SPDU	51
3.4.4. Paramètres de la couche Session	52
3.4.5. Transitions d'état.....	53
3.4.6. Répertoire et traitement des erreurs de la couche Session	60
3.5. Liste des erreurs	61
3.6. Principe du CRC	62
3.6.1. Généralités sur le CRC	62
3.6.2. Opérations sur les polynômes.....	62
3.6.3. Procédure de contrôle	62
3.6.4. Paramètres de fonctionnement	62

1. Généralités

1.1. Domaine d'application

La présente spécification décrit les données gérées par un appareil de comptage de technologie électronique et qui peuvent être lues par les utilisateurs des Réseaux de Distribution gérés par Enedis au moyen d'une liaison téléphonique filaire (RTC). Les appareils de comptage concernés sont les compteurs électroniques de type « Compteur Jaune Electronique » (CJE). Ils sont généralement utilisés dans le cas des utilisateurs du Réseau de Distribution de niveau Basse Tension (BT), de puissance souscrite comprise entre 36 kVA et 250 kVA, et bénéficiant du « Tarif Jaune » ou d'une autre tarification. Ces appareils permettent le comptage de l'énergie active, la quantification des dépassements de puissance apparente, ainsi que la gestion d'autres grandeurs pouvant être utiles à la gestion contractuelle ou à la maîtrise des consommations (puissance atteinte, ...).

L'appareil conserve dans sa mémoire un certain nombre d'informations sur une partie des mesures qu'il a effectuées. Ces informations sont accessibles localement. Elles peuvent être transmises par téléphone sur demande d'un système de traitement informatique si l'appareil est équipé d'un modem raccordé au réseau téléphonique commuté public (RTC).

Le compteur assure les différentes fonctions décrites ci-après :

- la mesure de l'énergie active et de la puissance apparente ;
- la gestion de la date et de l'heure de l'appareil (horloge interne) ;
- la gestion du tarif en cours par la sommation des énergies actives, les calculs des dépassements de puissance apparente, et leur répartition par période tarifaire (ou poste horosaisonnier) grâce à des tables internes définissant une structure « horosaisonnaire » et, si nécessaire, des ordres de télécommande externe (cas d'un tarif à effacement) ;
- la mémorisation des grandeurs nécessaires à la gestion contractuelle et à la gestion des consommations en fonction du cycle de gestion prévu par le tarif : les données de consommation servant à la facturation sont mémorisées pour différentes périodes tarifaires successives ;
- la visualisation sur l'afficheur d'une partie des informations mesurées ou élaborées par l'appareil, ainsi que des paramètres de fonctionnement ;
- le relevé des informations à distance : l'appareil doit pour cela être muni d'un modem raccordé au réseau téléphonique commuté permettant, à l'initiative d'un système de traitement informatique central, de réaliser un échange de données bidirectionnel ;
- la programmation à distance d'une partie des paramètres de fonctionnement (même procédé que le relevé à distance) ;
- le relevé des informations ou la programmation des paramètres de fonctionnement par une opération locale (sur Site) au moyen d'une liaison filaire dite « de téléreport » ;
- la mise à disposition d'informations sur un bornier réservé à l'utilisateur du réseau (contacts, sortie de téléinformation client) ;
- la mémorisation des points de puissance moyenne, en puissance apparente ou active et avec une durée différente suivant la version du compteur (cf. les précisions ci-après).

Précisions sur les différentes versions du compteur jaune électronique.

Pour les compteurs mis en exploitation depuis 1992 (arrivée de la première version du compteur jaune électronique) et jusqu'en 2005 (apparition de la deuxième version), la mémorisation s'effectue sur les puissances apparentes moyennes. A partir de fin 2005, une évolution du compteur a remplacé la mémorisation des puissances **apparentes** moyennes par celle des puissances **actives** moyennes, tout en augmentant également la durée de mémorisation.

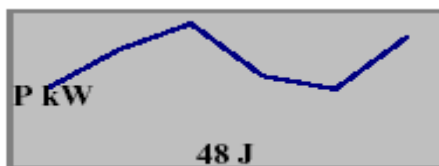
Cette nouvelle version permet la sauvegarde de 48 jours de courbe de charge en puissance active moyenne par période de 10 minutes en lieu et place des 3 jours de puissance apparente moyenne par période de 10 minutes permis par l'ancienne version.

C'est la seule différence fonctionnelle entre ces deux versions du compteur jaune électronique. Dans la suite du document, on appellera « V1 » les compteurs avec mémorisation de puissance apparente moyenne (version de 1992) et « V2 » les compteurs avec mémorisation de puissance active moyenne (version de fin 2005).

Les différents formats d'information fournis par les compteurs V1 et V2 sont décrits au chapitre 2.5.

L'utilisateur peut distinguer un compteur V2 par sa face avant différente portant la mention de la nouvelle spécification de référence (HR-43/00/022/B) et l'apposition d'un petit logo ayant l'apparence suivante :





Le présent document traite uniquement du relevé de certaines grandeurs accessibles à distance par la liaison téléphonique RTC sous réserves de présentation d'un code d'accès appelé aussi « clé client ». Dans le cas du présent compteur, ce code d'accès est aussi appelé « identifiant esclave ».

Le document est structuré en 2 parties principales :

- la partie 1 (chapitre 2) décrit d'abord les groupes de données disponibles, puis les détails des données mises à disposition par le compteur pour tout système de traitement informatique (non décrit dans ce document) qui est relié au modem du compteur par le réseau de téléphonie RTC ;
- la partie 2 (chapitre 3) décrit les protocoles de communication régissant les transferts de données du compteur vers le système de traitement informatique. Tout logiciel de traitement (non décrit dans ce document) du système de traitement informatique doit se conformer à ces protocoles.

1.2. Références normatives

Les normes internationales ou françaises citées en référence contiennent des dispositions qui, par suite de la référence qui en est faite, constituent des dispositions valables pour la présente spécification.

Les normes citées sont sujettes à révision.

NF EN ISO/CEI 7498-1: Technologies de l'information - Modèle de référence de base pour l'interconnexion de système ouverts (OSI) - Partie 1 : le modèle de base (décembre 1995).

NF ISO 7498-2: Systèmes de traitement de l'information - Interconnexion de systèmes ouverts - Modèle de référence de base - Partie 2 : architecture de sécurité (septembre 1990).

NF ISO 7498-3: Systèmes de traitement de l'information - Interconnexion de systèmes ouverts - Modèle de référence de base - Partie 3 : dénomination et adressage (juillet 1989).

UIT-T V23: Modem à 600/1200 bauds normalisé pour usage sur le réseau téléphonique général avec commutation.

UIT-T V24: Liste des définitions des circuits de jonction à l'interface entre l'équipement terminal de traitement de données (ETTD) et l'équipement de terminaison du circuit de données (ETCD).

UIT-T V25: Equipement de réponse automatique et/ou équipement d'appel automatique en mode parallèle sur le réseau téléphonique général avec commutation, y compris les procédures de neutralisation des dispositifs de protection contre l'écho lorsque les appels sont établis aussi bien entre postes à fonctionnement manuel qu'entre postes à fonctionnement automatique.

UIT-T V25bis: Equipement d'appel et/ou de réponse automatique sur le réseau téléphonique général avec commutation, utilisant les circuits de liaison de la série 100.

UIT-T V28: Caractéristiques électriques des circuits de jonction dissymétriques pour transmission par double courant.

UIT-T X25 : Interface entre équipement terminal de traitement de données (ETTD) et équipement de transmission du circuit de données (ETCD) pour terminaux fonctionnant en mode-paquet et raccordés à des réseaux publics pour données par circuit spécialisé.

ISO 2110: Technologies de l'information - Communication de données - Connecteur d'interface ETTD/ETCD à 25 pôles et affectation des numéros de contacts (1989).

Rappel : l'association UIT-T a remplacé le CCITT depuis 1993.

2. PARTIE 1 - Données télé-relevables par la liaison téléphonique

2.1. Définitions concernant le comptage et le télé-relevé

2.1.1. Principes généraux sur le télé-relevé des compteurs

L'utilisateur souhaitant exploiter des données par télé-relevé doit disposer d'un système de traitement informatique adapté. Ce système, développé à partir de la connaissance des protocoles de transmission des données, lui permet alors d'accéder à certaines informations stockées dans le compteur si celui-ci est équipé d'un modem raccordé au réseau téléphonique.

L'utilisateur du réseau, ou son mandataire, désirant accéder aux données d'un compteur par télé-relevé, doit obligatoirement, en préalable, demander aux services d'Enedis :

- la vérification de la présence dans le compteur d'un modem raccordé au réseau téléphonique RTC,
- la fourniture du code d'accès aux données du compteur,
- la ou les plages horaires d'appel qui lui sont attribuées pour l'accès aux données du compteur.

Les codes d'accès

Pour accéder aux données du compteur, il est nécessaire de disposer d'un code d'accès fourni sur demande par Enedis sous le nom de « clé client ». Dans le cas du présent compteur, ce code d'accès est aussi appelé « identifiant esclave » ou « identité de l'esclave ». Il devra être présenté par l'outil de traitement lors des échanges avec le compteur (cf. chapitre 3.4).

Lors de cette opération, un autre code devra être présenté qui est appelé « clé maître », « identifiant maître » ou « identité du maître ». Pour les opérations de lecture des données décrites ci-après, ce second code devra être affecté de la valeur nulle.

Les plages d'appel au compteur

L'accès aux données du compteur par le réseau téléphonique peut être utilisé par différentes entités telles que :

- d'une part, les différents services d'Enedis dans le cadre de leurs missions de relevé de données pour facturation, de relevé ou programmation de données pour contrôle ou maintenance des comptages ou de relevé de données pour l'exploitation du réseau (qualité de fourniture, ..) ;
- d'autre part, l'utilisateur du réseau et ses divers mandataires dans le cadre du suivi de sa consommation ou de prestations diverses.

Afin de garantir pour chacune de ces entités, la meilleure fiabilité possible dans l'exécution de ces accès téléphoniques au compteur et notamment, de ne pas perturber les opérations de télé-gestion du compteur effectuées par Enedis, un partage des possibilités d'accès téléphonique au compteur est mis en œuvre sous la forme d'une répartition temporelle en plages horaires d'appels qui sont réservées à chaque entité.

Enedis assure, pour chaque compteur, la gestion de ces plages et ainsi, la répartition des possibilités d'accès. Chaque entité s'engage à n'appeler le compteur que pendant la ou les plages horaires qui lui sont attribuées par Enedis. Ces plages pourront être plus ou moins importantes suivant le nombre d'entités concernées et le mode de raccordement du compteur au réseau téléphonique : raccordement sur une ligne dédiée au compteur ou sur une ligne partagée temporellement ou physiquement avec d'autres appareils (compteurs, appareils de mesure du distributeur ou matériels de l'installation téléphonique de l'utilisateur du réseau).

Précisions sur les modes de raccordement au réseau téléphonique commuté (RTC)

Le raccordement du compteur au réseau téléphonique commuté peut se faire :

- soit directement sur une ligne dont l'usage est réservé à ce compteur (ligne dédiée au compteur),
- soit sur une ligne téléphonique utilisée par plusieurs appareils.

Dans ce dernier cas, la répartition des appels ou « partage de ligne » peut être effectué par un appareil d'aiguillage physique (aiguilleur, PABX, ...) ou par une fonction de partage temporelle implémentée dans les appareils concernés (cas de la fenêtre d'écoute du compteur).

Dans le cas du compteur jaune électronique, la technique de la fenêtre d'écoute est le plus souvent utilisée.

Généralement, le partage de ligne a lieu :

- soit entre un compteur et les installations téléphoniques de l'utilisateur sur une ligne téléphonique dont il est le souscripteur,
- soit entre plusieurs compteurs sur une ligne téléphonique dont le distributeur est le souscripteur.

En cas de fenêtre d'écoute, Enedis fournira à l'utilisateur du réseau l'horaire exact de la fenêtre d'écoute pendant laquelle il peut appeler le compteur de son Site. Cette fenêtre constituera sa plage d'appel.
De manière générale, en fonction des contraintes générées par la configuration de l'installation téléphonique et les différentes utilisations existantes, Enedis fournira à l'utilisateur du réseau, les plages horaires d'appel pendant lesquelles il peut appeler le compteur de son Site.

Précisions sur le fonctionnement et l'usage de la fenêtre d'écoute

La technique utilisée pour le partage de la ligne filaire du réseau téléphonique commuté (RTC) est celle de l'appel « entrant » réparti suivant le principe d'une fenêtre temporelle d'écoute. L'appareil de comptage est placé en série sur la ligne téléphonique concernée, en amont des autres appareils qui utilisent cette ligne (autres compteurs ou appareils de mesure, ou appareils de l'installation de l'utilisateur du réseau tels que télécopieur ou téléphone).

Dans le compteur sont programmées des « fenêtres d'écoute », c'est à dire des plages temporelles caractérisées chacune par une heure de début et une durée maximale d'une demi-heure. Généralement, une fenêtre est attribuée au distributeur, une autre à l'utilisateur du réseau, mais aucune différence fonctionnelle réelle n'existe entre ces deux fenêtres dans le compteur.

Le compteur est programmé pour un fonctionnement en mode « fenêtre d'écoute ». Il est alors à l'état « transparent », c'est à dire qu'il laisse passer les communications dans les deux sens à tout moment de manière complètement transparente, excepté pendant les périodes temporelles correspondant aux fenêtres d'écoute.

Durant chaque période de 24h, pendant les périodes temporelles correspondant aux fenêtres, le compteur est à l'état "en écoute", c'est à dire qu'il décroche sur le premier appel arrivant du réseau vers l'installation téléphonique concernée (appel « entrant »). Les appareils partageant la ligne en aval du compteur ne recevront pas cet appel : celui-ci ne provoquera donc sur ces appareils aucune sonnerie, ni aucun « décroché ». A la fin de la communication correspondant à cet appel, le compteur raccroche. Il « ferme » la fenêtre d'écoute en cours et repasse à l'état « transparent » jusqu'au début de la prochaine fenêtre d'écoute. Une seule tentative d'appel au compteur est donc permise dans la période temporelle correspondant à une fenêtre d'écoute.

Pour sauvegarder l'usage de l'accès téléphonique par les autres appareils, le compteur ferme la fenêtre d'écoute en cours et repasse à l'état « transparent » dans les cas suivants :

- dès la fin de la communication du premier appel « entrant » (tout deuxième appel « entrant » pourra donc être reçu par les autres appareils),
- dès le début du premier appel « sortant », c'est à dire le premier appel émis de l'installation téléphonique concernée vers le réseau téléphonique commuté.

Dans le cas où, durant la fenêtre d'écoute, un appel « entrant » arrive et ne concerne pas le compteur, celui-ci décroche, identifie l'erreur, interrompt la communication, raccroche et ferme la fenêtre d'écoute. L'appelant peut alors avoir accès aux autres appareils placés en aval en renouvelant la tentative d'appel.

Afin d'éviter des échecs d'appels dus à une désynchronisation des heures courantes du compteur en fenêtre d'écoute appelé et du système de traitement informatique appelant, il est fortement recommandé de :

- maintenir avec précision la justesse de l'heure courante du système de traitement informatique appelant, ainsi que celle des informations qu'il gère concernant les plages horaires d'appel (et fenêtres d'écoute) des compteurs,
- réaliser les appels à l'intérieur de chacune des plages horaires d'appel en préservant une marge d'erreur de 5 minutes, c'est à dire en appelant le compteur, au plus tôt, 5 minutes après le début de la plage d'appel et, au plus tard, 5 minutes avant la fin de la plage d'appel,
- n'effectuer qu'une seule tentative d'appel à un compteur donné au cours d'une même plage d'appel si celle-ci est une fenêtre d'écoute.

2.1.2. Vocabulaire

Le « **poste horaire** » désigne une catégorie d'heures de la journée pendant lesquelles s'applique un tarif donné (par exemple : Heures Pleines, Heures Creuses, Heures de Pointe, Heures de Pointe Mobile, ...).

La « **période tarifaire** » (appelée aussi « **poste horosaisonnier** » ou « **poste tarifaire** ») désigne une combinaison de la « **saison** » (par exemple : « Hiver » de novembre à mars, « Eté » d'avril à octobre, ...) et du « **poste horaire** » (par exemple : Heures Pleines, Heures Creuses) qui permet de déterminer le tarif applicable à l'énergie mesurée à n'importe quel instant de l'année.

Le Compteur Jaune Electronique peut gérer plusieurs **options tarifaires** différentes grâce à la programmation de ses tables internes. Deux options tarifaires sont actuellement mises en œuvre dans le compteur, Base et EJP, qui correspondent chacune à une re-programmation complète de ses tables internes.

Pour chaque option tarifaire, le Compteur Jaune Electronique peut gérer jusqu'à 4 fonctionnements tarifaires distincts appelées « **versions tarifaires** » et basées sur des découpages horosaisonniers différents. Le découpage horosaisonnier prend en compte 2 saisons fixes : « Hiver » (de novembre à mars inclus) et « Été » (d'avril à octobre inclus) et, en plus, dans le cas de l'option EJP, la notion de « jour de pointe » dont la date est variable.

Lors de la mise en œuvre des règles tarifaires, si différents postes horosaisonniers font l'objet d'une gestion contractuelle strictement identique (même puissance souscrite, même valorisation des énergies consommée, ...), le compteur les regroupe en une même « **zone tarifaire** » sur laquelle il applique le même traitement à tous les postes horosaisonniers.

Le « **Tarif Jaune** » géré par le Compteur Jaune Electronique (CJE) comprend actuellement 4 variantes de la tarification qui sont réparties suivant 2 options tarifaires (Base et EJP) et 2 types d'utilisation (« utilisations moyennes » et « utilisations longues»). Les « **types d'utilisation** » sont des notions contractuelles qui ne sont gérées par le compteur que par leur rattachement aux versions tarifaires citées ci-dessus (se reporter au tableau ci-dessous).

L'option Base est déclinée en 2 types d'utilisation appelés « utilisations moyennes » et « utilisations longues » et comporte **4 postes horosaisonniers** :

- Heures Pleines d'Hiver (HPH) qui, dans le cas de la version « utilisations longues » peuvent elles-mêmes comporter 2 sous-postes :
 - Heures de Pointe (P) comprenant 2 fois 2 heures par jour (appelées aussi « Heures de Pointe Fixe »),
 - Heures Pleines d'Hiver hors pointe (HPH) comprenant toutes les heures de la saison « Hiver » qui ne sont ni en poste horaire de Pointe, ni en poste horaire d'heures creuses,
- Heures Creuses d'Hiver (HCH) : 8 heures par jour durant toute la saison « Hiver »,
- Heures Pleines d'Été (HPE) : toutes les heures de la saison « Été » qui ne sont pas en poste horaire d'Heures Creuses,
- Heures Creuses d'Été (HCE) : même définition que celles d'hiver, mais pour la saison « Été ».

L'option EJP (Effacement Jours de Pointe) est déclinée en 1 seul type d'utilisation appelé « utilisations longues », comporte **4 postes horosaisonniers** :

- Heures de Pointe Mobile (PM) : 18 heures par jour pendant 22 jours (non consécutifs) répartis sur les mois de la saison « Hiver »,
- Heures d'Hiver (HH) : toutes les heures de la saison « Hiver » qui ne sont pas en poste horaire d'Heures de Pointe Mobile,
- Heures Pleines d'Été (HPE) : toutes les heures de la saison « Été » qui ne sont pas en poste horaire d'Heures Creuses,
- Heures Creuses d'Été (HCE) : même définition que celles de l'option Base.

Le tableau ci-dessous décrit les correspondances entre les options, les versions tarifaires et les types d'utilisation.

Tableau des correspondances entre options, versions tarifaires et types d'utilisation

Option Tarifaire N° version tarifaire	Option Tarifaire Base	Option Tarifaire EJP
1	Utilisations Moyennes ou Utilisations Longues Sans dénivelé	Utilisations Longues Sans dénivelé
2	Utilisations Longues Avec dénivelé Hiver-Été (HCH / HPE)	Utilisations Longues Avec dénivelé Hiver-Été (HH / HPE)
3	Utilisations Longues Avec dénivelé Heures Pleines d'Hiver-Autres (HPH / HCH)	Utilisations Longues Avec dénivelé Pointe/Autres (PM / HH)
4	Utilisations Longues Avec dénivelé Pointe-Autres (P/HPH)	(inutilisée)

Il convient de noter que :

- les horaires exacts des changements de poste horaire (par exemple, le passage de « heures pleines » à « heures creuses ») sont définis localement et peuvent donc être différents d'un compteur à l'autre ;
- un découpage horosaisonnier différent peut exister sur des territoires non métropolitains.

La « **structure horosaisonnaire** » est composée d'une « **table annuelle** » et de quatre « **tables journalières** » qui permettent de définir le poste horosaisonnier en cours à chaque instant de l'année. C'est grâce à ces tables que le compteur peut, à tout instant, affecter les énergies et puissances mesurées au bon poste horosaisonnier.

La « **Fenêtre d'écoute client** » est utilisée dans le cas d'une ligne téléphonique partagée de manière temporelle (non dédiée durant 24h/24 au seul compteur concerné) ; c'est la plage horaire pendant laquelle l'utilisateur du réseau peut accéder à des données de comptage au moyen de l'accès téléphonique. Elle est caractérisée par une heure de début et une durée en minutes (cf. les précisions au chapitre 2.1.1).

Les « **Valeurs de références** » servent à vérifier que la ligne téléphonique répond correctement et que le dialogue peut bien être établi avec le modem du compteur.

Le **courant de base**, appelé « I_b » est une valeur de référence servant à décrire les plages de précision et les valeurs extrêmes de fonctionnement (en essais en laboratoire ou en exploitation).

Le **temps d'intégration pour la courbe de charge** « T_a » est une période de temps (exprimée en minutes) servant de référence et utilisée pour déterminer les valeurs moyennes de la puissance apparente (en compteur V1) ou de la puissance active (en compteur V2) enregistrées dans les tableaux de puissances moyennes pouvant être télé-relevés. Si T_a est paramétré à la valeur 10 (valeur usuelle), les tableaux de puissances moyennes représenteront une courbe de charge en « points 10mn » (puissance moyenne appelée par période de 10mn).

Le **temps d'intégration pour le calcul des dépassements de puissance** « T_q » est une période de temps (exprimée en minutes) servant de référence et utilisée pour le calcul de la moyenne quadratique de la puissance apparente. Cette valeur sert au calcul des dépassements de puissance. Si T_q est paramétré à la valeur 5 (valeur usuelle), les dépassements de puissance seront déterminés en prenant en compte, pour la puissance appelée, la moyenne quadratique de la puissance apparente sur une période de 5 minutes.

Les « **périodes contractuelles** » sont des périodes de temps pendant lesquelles les paramètres contractuels et techniques restent inchangés. Le compteur gère trois périodes contractuelles : la période en cours appelée « période P », la période précédente appelée « période P-1 » et la période encore précédente appelée « période P-2 ». De plus, il peut affecter certains paramètres à une période future appelée « P+1 ».

Le compteur effectue un changement de période, soit lorsqu'il y a un changement de l'un des paramètres techniques ou contractuels (puissance souscrite, version tarifaire, ...) pouvant impacter la gestion des données contractuelles, soit lorsqu'un changement de période de facturation est demandée (par exemple au moment du relevé servant à la facturation). Lors d'un changement de période contractuelle, le compteur ferme la période contractuelle en cours et ouvre une nouvelle période contractuelle. Pour cela, il effectue une opération appelée « **glissement** » qui consiste à copier les données gérées au titre de la période P en cours dans les données stockées au titre de la période P-1, et à copier les données stockées au titre de la période P-1 dans les données stockées au titre de la période P-2.

Lors du « glissement », les données de la période P en cours évoluent de la manière suivante :

- celles qui font l'objet d'un cumul sur plusieurs périodes contractuelles (données de type « index ») continuent d'évoluer à partir des valeurs contenues au changement de période contractuelle (cas des index d'énergie) ;
- celles qui sont de type cumul par période contractuelle sont remises à zéro, puis évoluent jusqu'au prochain changement de période contractuelle (cas des informations de dépassement de puissance).

Pour chacune de ces périodes contractuelles, le compteur conserve les informations concernant les paramètres contractuels (puissances souscrites, ...), la consommation d'énergie (index, ...) et les dépassements de puissance.

Rappel : « **TRIMARAN** » est le nom du protocole utilisé pour les communications téléphoniques en RTC avec le compteur électronique de type « Compteur Jaune Electronique » (CJE). Les spécifications détaillées de TRIMARAN sont décrites dans la

partie 2 (chapitre 3) du présent document.

2.1.3. La mesure de l'énergie active et les données gérées

Les calibres choisis pour les entrées mesure intensité et tension sont respectivement :

- intensité : courant de base IB = 5 A, courant maximum IB max = 6 A,
- tension : valeur de référence Un = 230 V.

Le compteur fonctionnant derrière un réducteur de courant, un coefficient de transformation « TC » est appliqué pour avoir l'énergie en kWh « primaires ».

Les transits négatifs de l'énergie d'une durée supérieure à une période 50 Hz ne sont pas comptabilisés.

Dans le compteur, il existe 6 variables de calcul à usage uniquement interne : elles sont inaccessibles par le relevé, et représentent les index internes d'énergie (notés EI à E6). Il existe également 6 variables d'usage externe appelées registres d'énergie (notés RE1 à RE6) qui sont accessibles par le relevé et sont utilisées pour la gestion contractuelle comme registres de quantification de l'énergie consommée.

Les registres d'énergie (internes et externes) sont constitués de 3 octets chacun, dont seulement les 20 bits de poids faibles sont utilisés. Ces 20 bits permettent d'atteindre la valeur binaire équivalente au nombre 999 999 kWh qui est la valeur maximale d'un index géré par le compteur. Les variables internes (EI à E6) ainsi que les registres d'énergie externes (RE1 à RE6) sont remis à zéro lors du passage à cette valeur maximale ou lors de chaque réinitialisation du compteur.

En cas de changement de poste horaire, les fractions de kWh sont conservées pour être restituées lors de la réapparition du poste concerné (sauf en cas de changement de saison ou de coupure secteur, auquel cas les fractions peuvent être perdues).

L'affectation des variables internes (E1 à E6) dans les registres d'énergie (RE1 à RE6) servant à la gestion contractuelle est définie par la configuration du compteur à sa mise en service. Cette affectation est décrite dans le tableau suivant. Cette affectation permet le regroupement dans une même zone tarifaire de plusieurs postes horosaisonniers devant faire l'objet d'une gestion contractuelle strictement identique au sens de la mesure des énergies consommées.

Il convient de noter que, dans le cas de la version du Tarif Jaune actuellement implémentée dans le compteur, les versions tarifaires disponibles ne définissent pas plus de 4 registres d'énergie. C'est pourquoi, seulement 4 registres d'énergie à usage externe sont utilisés (RE1 à RE4). Les 2 autres registres (RE5 et RE6) sont visibles et peuvent être affectés de valeurs par défaut, mais restent inutilisées par le compteur.

Affectation des variables internes aux registres externes de gestion de l'énergie

Index interne	Poste horosaisonnier	Registre d'énergie à utiliser
E1	Heures de Pointe Mobile (PM) en option EJP	RE1
E2	Heures de Pointe (P) en option Base	RE1
E3	Heures Pleines d'Hiver en option Base et EJP	RE1
E4	Heures Creuses d'Hiver en option Base et EJP	RE2
E5	Heures Pleines d'Eté en option Base et EJP	RE3
E6	Heures Creuses d'Eté en option Base et EJP	RE4

Dans le tableau précédent, l'existence d'un même registre d'énergie RE1 pour 3 variables internes différentes (EI, E2 et E3) indique que le registre d'énergie RE1 servant à la gestion contractuelle est la somme des index se trouvant dans les 3 variables internes correspondant. En pratique, cela revient à additionner, conformément au tarif, les énergies consommées dans les postes horosaisonniers Heures de Pointe et Heures Pleines d'Hiver de l'option Base, ou Heures de Pointe mobile et Heures Pleines d'Hiver de l'option EJP.

2.1.4. Le contrôle des dépassements de puissance et les données gérées

La **puissance apparente** est mesurée sur les 3 phases : cette mesure est effectuée sur une période **d'une minute**.

La gestion contractuelle nécessite de contrôler le niveau atteint par la puissance apparente appelée dans chacun des postes horosaisonniers en comparaison du seuil contractuel correspondant. Ce seuil est le produit de la puissance souscrite du poste horosaisonnier considéré par un coefficient K de dépassement qui représente une tolérance de dépassement de cette puissance souscrite ($1,00 \leq K \leq 1,60$) avant la comptabilisation effective du dépassement par le compteur. Le compteur peut gérer plusieurs coefficients K spécifiques à certains postes horosaisonniers. Néanmoins, en pratique, le même coefficient est utilisé pour tous les postes horosaisonniers.

La puissance apparente moyenne appelée est mesurée sur des périodes d'une minute pour constitué des « éléments-1-minute ». Un calcul est effectué, à partir des éléments « 1 minute » ainsi obtenus, pour définir la puissance apparente utile sur une période glissante de plusieurs minutes. La durée de la période glissante est définie en minutes par la grandeur « Tq ». Les valeurs possibles sont 1, 5 ou 10 minutes.

Si Tq est égale à 1 minute, la puissance apparente utilisée pour le calcul des dépassements est directement égale à la puissance mesurée par les « éléments-1-minute », sinon, elle est égale à la moyenne quadratique des « Tq » derniers élément-1-minute mesurés.

En cas de changement de poste horosaisonnier, le calcul de la moyenne quadratique est réinitialisé en utilisant d'abord le premier élément-1-minute (n=1), puis les deux premiers (n=2), ainsi de suite, jusqu'à l'établissement de la règle standard lorsque n est égal à Tq. Soit pour n étant libellé en minutes :

$$S_{Tq} = \left(\frac{1}{n} \sum_{i=1}^{i=n} S_i^2 \right)^{\frac{1}{2}} \quad n \leq T_q$$

Les grandeurs résultantes du traitement des puissances sont exprimées en dizaines de volt-ampères (daVA). La troncature se fait suivant la règle suivante :

- ≥ 5 VA : arrondi à la dizaine supérieure,
- ≤ 5 VA : arrondi à la dizaine inférieure.

On considère qu'il y a dépassement dès que la puissance atteinte dans un poste horosaisonnier considéré est supérieure au produit du seuil de déclenchement K par la puissance souscrite Ps en cours pendant la dernière période «1 minute ».

Le contrôle est effectué dans chacun des postes horosaisonniers.

Dans le compteur, il existe 6 zones de calcul à usage uniquement interne concernant les dépassements. Ces zones sont inaccessibles par le relevé, et représentent les registres internes de dépassement (notées ici S1 à S6). Il existe également 4 zones de variables d'usage externe (notées de A à D) qui sont accessibles par le relevé et contiennent les grandeurs nécessaires à la gestion contractuelle des dépassements de puissance.

L'affectation des zones internes de calcul (S1 à S6) dans les zones externes (A à D) accessibles et utilisées pour la gestion contractuelle est définie par la configuration du compteur à sa mise en service. Cette affectation est décrite dans le tableau suivant. Cette affectation permet le regroupement dans une même zone tarifaire de plusieurs postes horosaisonniers devant faire l'objet d'une gestion contractuelle strictement identique au sens des dépassements de puissances.

Il convient de noter que, dans le cas de la version du Tarif Jaune actuellement implémentée dans le compteur, les versions tarifaires disponibles ne définissent pas plus d'un dénivelé de puissance, c'est à dire au maximum deux seuils de puissance souscrite à gérer. C'est pourquoi, seulement 2 zones de variables d'usage externe sont utilisées (A et B). Toutes les grandeurs correspondant aux 2 autres zones (C et D) sont visibles et peuvent être affectées de valeurs par défaut, mais restent inutilisées par le compteur.

Affectations des zones internes (S1 à S6) aux zones tarifaires externes (A à B) utilisées pour le contrôle des dépassements

Zones internes	S1	S2	S3	S4	S5	S6
Poste horosaisonnier	Pointe Mobile (PM)	Pointe (P)	Heures Pleines d'Hiver	Heures Creuses d'Hiver	Heures Pleines d'Eté	Heures Creuses d'Eté
Option	EJP	Base	Base et EJP	Base et EJP	Base et EJP	Base et EJP
Option Base - Version n°1 Sans dénivelé	(A)	A	A	A	A	A
Option Base - Version n°2 Avec dénivelé Hiver-Eté (HCH / HPE)	(B)	B	B	B	A	A
Option Base - Version n°3 Avec dénivelé Heures Pleines d'Hiver-Autres (HPH / HCH)	(B)	B	B	A	A	A
Option Base - Version n°4 Avec dénivelé Pointe-Autres (P/HPH)	(B)	B	A	A	A	A
Option EJP - Version n°1 Sans dénivelé	A	(A)	A	A	A	A
Option EJP - Version n°2 Avec dénivelé Hiver-Eté (HH / HPE)	B	(B)	B	B	A	A
Option EJP - Version n°3 Avec dénivelé Pointe/Autres (PM / HH)	B	(B)	A	A	A	A

Nota : bien que prévu par la programmation, les affectations notées entre parenthèses ne seront en principe jamais être activées.

Exemples d'interprétation du tableau précédent :

- en option EJP et version tarifaire n°1 (sans dénivelé), une seule zone tarifaire est gérée : quelque soit le poste horosaisonnier courant, tous les dépassements sont contrôlés de la même manière (par rapport à une même puissance souscrite PSA suivant un même coefficient de dépassement KA) et tous les dépassements sont comptabilisés ensemble (cumul) dans les registres de la zone tarifaire A ;
- en option Base et version tarifaire n°3 (Avec dénivelé Heures Pleines d'Hiver-Autres), deux zones tarifaires sont gérées : une zone (B) correspond aux deux postes tarifaires Pointe et Heures Pleines d'Hiver et une autre zone (A) correspond aux trois autres postes tarifaires. Lorsque le compteur est dans les postes tarifaires Pointe ou Heures Pleines d'Hiver, les dépassements sont contrôlés de la même manière (par rapport à une même puissance souscrite PSB suivant un même coefficient de dépassement KB) et sont comptabilisés ensemble (cumul) dans les registres de la zone tarifaire B. Lorsque le compteur est dans l'un des trois autres postes tarifaires (Heures Creuses d'Hiver, Heures Pleines d'Eté, Heures Creuses

d'Eté), les dépassements sont contrôlés de la même manière (par rapport à une même puissance souscrite PSA suivant un même coefficient de dépassement KA) et sont comptabilisés ensemble (cumul) dans les registres de la zone tarifaire A.

Concernant la comptabilisation des dépassements, chaque zone tarifaire (A à D) est munie de deux grandeurs de référence et de 2 registres.

Les 2 grandeurs de références sont les suivantes :

- une grandeur de puissance souscrite (notée PSz où z est le nom de la zone tarifaire considérée) contient la valeur de la puissance souscrite de référence pour cette zone, c'est à dire la valeur de la puissance souscrite commune à tous les postes horosaisonniers affectés à cette zone tarifaire ;
- une grandeur de coefficient de dépassement (notée Kz où z est le nom de la zone tarifaire considérée) contient la valeur du coefficient de dépassement de référence pour cette zone, c'est à dire la valeur du coefficient de dépassement commune à tous les postes horosaisonniers affectés à cette zone tarifaire.

Les 2 registres sont les suivants :

- un registre (noté RPMz où z est le nom de la zone tarifaire considérée) contient la valeur de la puissance apparente maximale atteinte pour cette zone tarifaire, c'est à dire la valeur de la plus grande des puissances apparentes maximales atteintes dans chacun des postes horosaisonniers affectés à cette zone tarifaire. Ces valeurs sont exprimées en dizaines de volt-ampère (daVA), ramenées au primaire ;
- un registre (noté RNiz où z est le nom de la zone tarifaire considérée) contient la valeur de la durée totale de dépassement (en nombre de minutes) pour cette zone, c'est à dire la somme des durées de dépassement enregistrées dans chacun des postes horosaisonniers affectés à cette zone.

Le compteur gère donc 8 grandeurs de référence et 8 registres correspondant aux 4 zones tarifaires, notées de A à D. Toutes les grandeurs et registres sont accessibles par le relevé mais seule une partie d'entre elles sont utilisées.

Les grandeurs de références sont programmées aux valeurs contractuelles en cours conformément au tableau des affectations précédent :

- les grandeurs PSA et PSB représentent les puissances souscrites des deux seuils de puissance prévus par le tarif pour les zones tarifaires A et B (PSA seulement en version tarifaire sans dénivelé donc à un seul seuil). Les grandeurs PSC et PSD sont inutilisées ;
- les grandeurs KA et KB représentent les coefficients de dépassement. Conformément aux règles tarifaires, la même valeur de coefficient est utilisée pour tous les postes horosaisonniers, et donc pour toutes les zones tarifaires. Les grandeurs KC et KD sont inutilisées. La valeur du coefficient est exprimée en pourcentage. Par exemple, une valeur égale à 103 signifie que la comptabilisation du dépassement est effectuée à partir de 3% au dessus de la puissance souscrite.

Les registres contiennent les valeurs mesurées par le compteur pour la quantification des dépassements conformément au tableau des affectations précédent :

- les registres RNIA et RNIB contiennent les valeurs du nombre de minutes de dépassement de puissance pour les zones tarifaires A et B (RNIA seulement en version tarifaire sans dénivelé donc à un seul seuil). Les grandeurs RNIC et RNID sont inutilisées ;
- les registres RPMA et RPMB contiennent les valeurs de la puissance apparente maximale atteinte pour les zones tarifaires A et B (RPMA seulement en version tarifaire sans dénivelé donc à un seul seuil). Les grandeurs RPMC et RPMD sont inutilisées.

Seule une partie de ces informations concernent le contrat en cours et les consommations mesurées. Elles sont accessibles à distance via une liaison téléphonique sous la forme de données classées en 7 groupes qui sont, chacun, identifiés par un « code de télé-relevé » correspondant au numéro du groupe de données.

2.2. Liste des groupes de données télé-relevables et mode d'accès

Le compteur gère de nombreuses informations programmables ou relevables sur Site, ou à distance via une liaison téléphonique.

La plupart de ces informations concernent le contrat en cours et les consommations mesurées. Elles sont accessibles à distance sous la forme de données classées en 7 groupes qui sont, chacun, identifiés par un « code de télé-relevé » correspondant au numéro du groupe de données.

Le tableau suivant donne les désignations de ces groupes de données, ainsi que les valeurs des « codes de télé-relevé » associés. Le code de télé-relevé d'un groupe de données devra être fourni au compteur par le système de traitement informatique lors des échanges de communication afin d'identifier les informations à fournir par le compteur.

Liste des données de télé-relevé

Désignation des données télé-relevées	Code de télé-relevé
Contrats	0C
Structure horosaisonnière	0B
Tableau des puissances moyennes	08
Informations de la période P en cours	02
Informations des périodes précédentes P-1 et P-2	01
Valeurs de référence	05
Tableau des appels téléphoniques	07

Précisions sur le mode d'accès aux groupes de données télé-relevables

Pour accéder à un groupe de données télé-relevables, il convient de communiquer au compteur les informations correspondant à ce groupe dans le double-octet de la variable « Code » conformément à ce qui est précisé dans le chapitre 3 décrivant les protocoles utilisés pour la communication, au chapitre 3.4.

Le contenu du 1er octet du double-octet de la variable « Code » est la valeur (codée en hexadécimal) du code de télé-relevé du groupe de données désiré conformément au tableau ci-dessus.

Le contenu du 2ème octet du double-octet de la variable « Code » est variable en fonction du type de compteur et du groupe de données désiré. De manière générale, il peut contenir soit une référence identifiant l'application du compteur, soit un code complémentaire du code de télé-relevé précisant l'identité des données désirées.

Pour le compteur jaune électronique, son contenu est le suivant :

- dans le cas du compteur V1 : cet octet est toujours égal à la référence identifiant l'application du compteur, soit la valeur 10 (codée en BCD) ;
- dans le cas du compteur V2 : cet octet est égal à la référence identifiant l'application du compteur comme pour le compteur V1, excepté dans le cas du groupe de données « Tableau des puissances moyennes » (de code 08) où le contenu de cet octet désigne le bloc de données à transférer (valeur codée en BCD de 10 à 25 pour désigner respectivement du premier au dernier des 16 blocs de données du groupe de données concerné). Se reporter au chapitre 2.5.2 pour plus de précisions.

2.3. Contrats (code 0C)

Ce tableau fournit les informations contractuelles de la période contractuelle en cours, appelée « période P », (octets 1 à 13 du tableau) et de la période contractuelle future, appelée « période P+1 », (octets 14 à 26) sous la forme d'un ensemble de 26 octets définis dans le tableau suivant.

Les informations transmises concernent, pour chaque contrat, les puissances souscrites pour les différentes zones tarifaires.

Tableau des données des contrats

Intitulé	Description de la donnée	Codage ou unité	Taille	Position du 1 ^{er} octet dans la chaîne
TARIF	Version tarifaire de la période contractuelle en cours P		1 octet	1
PSA	Puissance souscrite de la zone tarifaire A de la période contractuelle en cours P	daVA	2 octets	2
PSB	Puissance souscrite de la zone tarifaire B de la période contractuelle en cours P	daVA	2 octets	4
PSC	Puissance souscrite de la zone tarifaire C de la période contractuelle en cours P (inutilisée)	daVA	2 octets	6
PSD	Puissance souscrite de la zone tarifaire D de la période contractuelle en cours P (inutilisée)	daVA	2 octets	8
KA	Coefficient de dépassement de la zone tarifaire A de la période contractuelle en cours P	%	1 octet	10
KB	Coefficient de dépassement de la zone tarifaire B de la période contractuelle en cours P (utilisé seulement si dénivelé)	%	1 octet	11
KC	Coefficient de dépassement de la zone tarifaire C de la période contractuelle en cours P (inutilisé)	%	1 octet	12
KD	Coefficient de dépassement de la zone tarifaire D de la période contractuelle en cours P (inutilisé)	%	1 octet	13
TARIFP	Version tarifaire de la période contractuelle future P+1		1 octet	14
PSAP	Puissance souscrite de la zone tarifaire A de la période contractuelle future P+1	daVA	2 octets	15
PSBP	Puissance souscrite de la zone tarifaire B de la période contractuelle future P+1 (utilisé seulement si dénivelé)	daVA	2 octets	17
PSCP	Puissance souscrite de la zone tarifaire C de la période contractuelle future P+1 (inutilisée)	daVA	2 octets	19
PSDP	Puissance souscrite de la zone tarifaire D de la période contractuelle future P+1 (inutilisée)	daVA	2 octets	21
KAP	Coefficient de dépassement de la zone tarifaire A de la période contractuelle future P+1	%	1 octet	23
KBP	Coefficient de dépassement de la zone tarifaire B de la période contractuelle future P+1 (utilisé seulement si dénivelé)	%	1 octet	24
KCP	Coefficient de dépassement de la zone tarifaire C de la période contractuelle future P+1 (inutilisé)	%	1 octet	25
KDP	Coefficient de dépassement de la zone tarifaire D de la période contractuelle future P+1 (inutilisé)	%	1 octet	26

Le chapitre 2.1.4 précise les définitions des zones tarifaires A et B, ainsi que l'utilisation des coefficients de dépassement en fonction de l'option et de la version tarifaires choisies. La zone B n'est utilisée que pour les versions tarifaires avec dénivelé. Les zones tarifaires C et D ne sont jamais utilisées.

Précisions sur les informations TARIF et TARIFP

Cet octet est codé sous la forme suivante :

bits 7 à 6	bits 5 à 4	bits 3 à 0
Numéro de la saison en cours	Numéro du poste horaire en cours	Numéro de la version tarifaire en cours

Le numéro de la saison en cours est codé de la manière suivante :

- 1 pour l'été, 2 pour l'hiver, 4 pour la période de pointe mobile.

Le numéro du poste horaire en cours est codé de la manière suivante :

- 1 pour « Heures Pleines », 2 pour « Heures Creuses », 3 pour « Heures de Pointe » et 4 pour « Heures de Pointe Mobile ».

Le numéro de la version tarifaire en cours est codé conformément aux numéros précisés dans le tableau du chapitre 2.1.4.

Pour la période contractuelle en cours P, les 3 numéros de l'octet TARIF (saison, poste horaire et version tarifaire) sont renseignés. Pour la période contractuelle future P+1, seul le numéro de la version tarifaire de l'octet TARIFP est renseigné.

2.4. Structure horosaisonnaire (code 0B)

2.4.1. Généralités sur la structure horosaisonnaire

La structure horosaisonnaire est programmée dans le compteur lors de la mise en service ou d'un changement contractuel. Sa valeur dépendra des options et versions tarifaires choisies et des horaires d'heures creuses et d'heures de pointe appliqués pour le Site concerné.

Ce tableau fournit les informations concernant la structure horosaisonnaire de la période contractuelle future (P+1) sous la forme d'un tableau de 72 octets. Si aucune modification de structure n'est prévue (cas général), ce tableau correspond alors à la structure horosaisonnaire active pour la période contractuelle en cours (P).

Rappel : la « structure horosaisonnaire » est composée d'une « table annuelle » et de quatre « tables journalières » qui permettent de définir le poste horosaisonnier en cours à chaque instant de l'année. C'est grâce à ces tables que le compteur peut, à tout instant, affecter les énergies et puissances mesurées au bon poste horosaisonnier.

En fonction de la date courante, la table annuelle permet au compteur de déterminer la saison en cours et la table journalière à utiliser (variable suivant les saisons ou les jours de la semaine). En fonction de l'heure courante, la table journalière ainsi déterminée permet au compteur de déterminer le poste horaire en cours. Ainsi, le poste horosaisonnier en cours est défini.

Les changements de journées tarifaires (ainsi que les changements de mois ou de saisons) interviennent toujours à 2 heures du matin. C'est donc généralement à 2 heures du matin qu'une nouvelle journée tarifaire commence et donc qu'une nouvelle table journalière est activée. C'est pourquoi les tables journalières définissent la répartition des postes horaires dans un cycle journalier 2h00-2h00.

Exceptionnellement, en cas de remise à l'heure ou de changement d'heure légale, une lecture de la structure horosaisonnaire sera effectuée en vue de déterminer le nouveau poste horosaisonnier en vigueur.

Données de la structure horosaisonnaire

Intitulé	Description de la donnée	Codage ou unité	Taille	Position du 1 ^{er} octet dans la chaîne
Table annuelle des 12 mois	Table des mois de janvier à décembre	Cf. § 2.4.2	12 octets	1
Table journalière 1	Table à 5 éléments de 2 octets chacun	Cf. § 2.4.3	10 octets	13
Table journalière 2	Table à 10 éléments de 2 octets chacun	Cf. § 2.4.3	20 octets	23
Table journalière 3	Table à 5 éléments de 2 octets chacun	Cf. § 2.4.3	10 octets	43
Table journalière 4	Table à 10 éléments de 2 octets chacun	Cf. § 2.4.3	20 octets	53

Les 2 sous-chapitres suivants détaillent les structures des octets de la table annuelle et des tables journalières.

2.4.2. Format de la table annuelle

La table annuelle comprend 12 octets. Chaque octet définit les tables journalières à utiliser pour le mois considéré (1er octet pour Janvier, ...). Chaque octet est structuré de la façon suivante :

bits 7 à 6	bits 5 à 4	bits 3 à 2	bit 1	bit 0
Numéro de Saison	Numéro de table Principale	Numéro de table secondaire	Numéro de table du samedi	Numéro de table du dimanche

Il est défini deux types de table journalière :

- une « table principale » qui est systématiquement utilisée pour les jours du lundi au vendredi,
- une « table secondaire » qui peut être utilisée (ou non) pour les jours de week-end (samedi et dimanche).

Les bits 7 à 6 définissent la saison à laquelle appartient le mois considéré conformément au codage suivant :

bit 7	bit 6	SAISON
0	0	Saison n° 1
0	1	Saison n° 2
1	0	Saison n° 3
1	1	Saison n° 4

Rappel (cf. chapitre 2.3) : le numéro de la saison est codé de la manière suivante (1 pour été, 2 pour hiver, 4 pour pointe mobile).

Les bits 5 à 2 définissent les numéros des tables journalières (parmi les 4 définies plus loin) qui sont choisies comme « table principale » (bits 5 à 4) et comme « table secondaire » (bits 3 à 2) conformément au codage suivant :

bit 5 (ou bit 3)	bit 4 (ou bit 2)	Table principale ou secondaire
0	0	Table journalière n° 1
0	1	Table journalière n° 2
1	0	Table journalière n° 3
1	1	Table journalière n° 4

L'usage de la table principale ou de la table secondaire pour les jours de week-end (samedi et dimanche) est défini conformément au codage suivant :

bit 1 (ou bit 0)	Table utilisée pour le samedi (ou le dimanche)
0	Table principale
1	Table secondaire

2.4.3. Format de la table journalière

Chaque table journalière comprend 5 ou 10 éléments de 2 octets chacun qui permettent de définir les changements de poste horaire durant tout le cycle journalier entre 2h00 et 2h00.

Chaque cycle commence à 2h00. Chaque élément est constitué d'un horaire H et d'un poste horaire P. L'insertion de l'élément dans la table ordonne au compteur d'appliquer le poste horaire P jusqu'à l'horaire H.

Le codage de l'horaire et du poste horaire dans le double-octet de l'élément est réalisé de la manière suivante :

bits 15 à 13	bits 12 à 8	bits 7 à 2	bit 1 à 0
Inutilisé	Heure	Minute	Poste horaire

Les bits 15 à 13 de l'élément sont inutilisés. L'heure de l'horaire de l'élément est codée en binaire sur 5 bits, Les minutes de l'horaire sont codées en binaire sur 6 bits. Le poste horaire de l'élément est codé en binaire sur 2 bits conformément au codage ci-dessous :

bit 1	bit 0	Poste horaire
0	0	Poste horaire n° 1
0	1	Poste horaire n° 2
1	0	Poste horaire n° 3
1	1	Poste horaire n° 4

Rappel (cf. chapitre 2.3) : le numéro du poste horaire est codé de la manière suivante : 1 pour « Heures Pleines », 2 pour « Heures Creuses », 3 pour « Heures de Pointe » et 4 pour « Heures de Pointe Mobile ».

2.5. Tableau des puissances moyennes (code 08).

2.5.1. Cas du compteur en version V1

Ce tableau fournit des informations sur les puissances moyennes mesurées par le compteur.

Les valeurs fournies dans ce tableau correspondent à des **moyennes de la puissance apparente atteinte appelée par le Site concerné** pour chaque période de temps d'une durée égale à « Ta » minutes. Cette constante « Ta » est appelée « temps d'intégration pour la courbe de charge ».

Les valeurs possibles de « Ta » sont : 5, 10 et 15. La valeur usuelle est 10. Les informations du tableau permettent alors de reconstituer la courbe de charge du Site considéré par période de temps de 10 minutes.

Pour calculer ces moyennes « Ta minutes », le compteur effectue automatiquement des mesures sur une période de base de durée égale à une minute. Puis, il calcule les moyennes « Ta minutes » en cumulant les mesures effectuées sur la période de base.

Les valeurs de puissance apparente sont fournies avec une résolution de 1 kVA pour des puissances atteintes de 0 à 2047 kVA grâce à un codage des valeurs en binaire sur 11 bits.

Le tableau des puissances moyennes est constitué de 1024 octets, répartis en 512 éléments constitués chacun d'un double-octet (16 bits). L'octet de poids faible de chaque double-octet est envoyé en tête lors de la transmission. La description précise des éléments du tableau des puissances moyennes est fournie au chapitre 2.5.3.

La durée totale de l'enregistrement représenté par le tableau est variable en fonction de la valeur de la constante « Ta ». Le tableau ci-dessous indique les valeurs approximatives de la durée totale d'enregistrement du compteur. La valeur exacte dépendra également des « événements » survenus durant la période considérée : coupure d'alimentation, événements nécessitant l'insertion dans le tableau d'un élément de datation (cf. au chapitre 2.5.3 les précisions sur ces événements).

Valeur de « Ta »	Durée totale d'enregistrement du compteur
5 minutes	1 jour et demi
10 minutes	3 jours
15 minutes	4 jours et demi

Pour éviter toute perte d'information, il convient de relever le tableau des puissances moyennes suivant une périodicité inférieure ou égale à la durée totale d'enregistrement indiquée ci-dessus.

2.5.2. Cas du compteur en version V2

Par rapport au chapitre précédent, la seule différence consiste au passage d'une courbe de charge de 3 jours (par période de 10 minutes) en puissance apparente à une **courbe de charge de 48 jours** (toujours par période de 10 minutes) en **puissance active**.

Il s'agit de puissances actives moyennes. Elles sont calculées par intégration sur une période fixe, contrairement au calcul de la puissance apparente moyenne qui, lui, s'effectue sur une fenêtre glissante.

Les valeurs possibles du temps d'intégration pour la courbe de charge sont : 5, 10 et 15. La valeur usuelle est 10. Les informations du tableau permettent alors de reconstituer la courbe de charge du Site considéré par période de temps de 10 minutes.

Le tableau ci-dessous indique les valeurs approximatives de la durée totale d'enregistrement du compteur en fonction de la valeur de la constante « Ta » (temps d'intégration pour la courbe de charge).

Valeur de « Ta »	Durée totale d'enregistrement du compteur
5 minutes	24 jours
10 minutes	48 jours
15 minutes	72 jours

La taille du tableau des puissances moyennes est de 16384 octets, répartis en 8192 éléments constitués chacun d'un double-octet (16 bits).

La description précise des éléments du tableau des puissances moyennes est identique au cas du compteur V1 et fournie au chapitre 2.5.3. Seule l'unité de la valeur de la puissance moyenne diffère, celle-ci étant exprimée en kW.

Afin que, malgré les arrondis, l'énergie totale consommée se retrouve entièrement comptabilisée dans la courbe de charge, un report des arrondis est mis en œuvre d'une période d'intégration à la suivante. L'arrondi se fait à la valeur entière la plus proche. Le report d'arrondi peut donc être positif ou négatif.

L'accès au tableau s'effectue par accès partiel à différents blocs de transfert de taille identique. La taille du bloc de transfert est, elle, conservée identique à celle du tableau du compteur V1, c'est-à-dire 1024 octets répartis en 512 éléments constitués chacun d'un double-octet (16 bits). Le tableau complet est donc constitué de 16 blocs de format identique. Chaque bloc de transfert est identifié par un numéro de bloc.

L'accès à un bloc de transfert s'effectue par la transmission de son numéro dans le 2ème octet de la variable « Code » conformément à la description du chapitre 2.2.

Pour obtenir le premier bloc de données correspondant aux dernières valeurs enregistrées dans la courbe de charge, il convient d'affecter au 2ème octet de la variable « Code » la valeur **10** (codage en BCD). Cette valeur assure la compatibilité avec le fonctionnement du compteur V1.

Pour obtenir le bloc de transfert contenant les valeurs enregistrées précédemment, il convient d'affecter au 2ème octet de la variable « Code » la valeur **11** (codage en BCD). Il convient de procéder de la même manière pour les blocs contenant les valeurs enregistrées précédemment jusqu'à la valeur **25** (codage en BCD) pour le 16ème bloc contenant les valeurs les plus anciennes.

Entre le début et la fin d'une communication, l'enregistrement d'un nouvel élément dans le tableau n'affecte pas le contenu des blocs de transfert afin que ceux-ci restent consécutifs.

2.5.3. Informations supplémentaires (valables pour les compteurs V1 et V2)

2.5.3.1. Généralités sur les éléments constituant le tableau

Afin de permettre la reconstitution complète de la courbe de charge, le tableau relevé comporte en plus des informations de

puissance appelées « **élément-puissance** », des informations de datation appelés « **élément-datation** » contenant des informations de date, heure et de saison tarifaire. Les informations fournies permettent également de distinguer les périodes « Ta minutes » pendant lesquelles s'est produite une coupure d'alimentation.

Le tableau est organisé suivant une pile de type FIFO (First In First Out). Chaque insertion d'un nouvel élément en tête du tableau provoque la sortie en queue de tableau de l'élément le plus ancien.

2.5.3.2. Élément-puissance

Un élément-puissance mémorise des informations de puissance apparente (V1) ou active (V2) par période de « Ta » minutes.

Il est caractérisé par la présence du bit 15 à la valeur 0.

La composition du double-octet de l'élément-puissance est la suivante :

bit 15	bits 14 à 13	bits 12 à 11	bits 10 à 0
0	Information de coupure	Poste Horaire	Valeur de la puissance moyenne (en kVA ou kW)

Les bits 14 et 13 du double-octet informent de la présence d'une éventuelle coupure d'alimentation pendant tout ou partie de la période « Ta minutes » considérée (cf. plus loin).

Les bits 12 et 11 du double-octet informent du poste horaire en vigueur lors de la minute précédant l'échéance « Ta min » considérée suivant le codage décrit dans le tableau ci-après. En règle générale, ce poste horaire est celui en vigueur pendant la période de « Ta minutes » considérée.

bit 12	bit 11	Poste horaire
0	0	Poste horaire n° 1
0	1	Poste horaire n° 2
1	0	Poste horaire n° 3
1	1	Poste horaire n° 4

Rappel (cf. chapitre 2.3) : le numéro du poste horaire est codé de la manière suivante : 1 pour « Heures Pleines », 2 pour « Heures Creuses », 3 pour « Heures de Pointe » et 4 pour « Heures de Pointe Mobile ».

Les bits 10 à 0 du double-octet informe de la valeur de la puissance apparente (ou active) moyenne atteinte sur la période de « Ta minutes » considérée. Ils contiennent cette valeur, codée en binaire sur 11 bits avec une résolution de 1 kVA (ou 1 kW).

Précisions sur l'information de coupure d'alimentation

Cette information est fournie par les bits 14 et 13 du double-octet de l'élément-puissance.

Le compteur n'est pas sensible aux coupures de durée inférieure à 0,5 seconde. Il est défini un seuil « D » (exprimé en secondes) permettant de distinguer, d'après leur durée, les coupures caractérisées comme « petites » et les coupures caractérisées comme « grandes ». Ce seuil est habituellement fixé à 60 secondes.

On distingue donc deux sortes de coupure :

- les « petites » coupures de durée comprise entre 0,5 s et le seuil « D » de 60 secondes,
- les « grandes » coupures de durée supérieure au seuil « D » de 60 secondes.

Les valeurs des bits 14 et 13 du double-octet de l'élément-puissance sont définies en fonction de la durée de coupure conformément au codage suivant :

bit 14	bit 13	Durée de coupure
0	0	Pas de coupure
0	1	Petite coupure
1	0	Grande coupure
1	1	Cas particulier : élément-puissance tronqué (cf. § 2.5.3.4)

2.5.3.3. Éléments de datation

Deux sortes d'éléments de datation sont fournis dans le tableau des puissances moyennes : des éléments-heure et des éléments-date. Ils mémorisent les informations d'horodate permettant de reconstituer la courbe de charge.

Les éléments de datation sont constitués d'un double octet et sont caractérisés par la présence du bit 15 à la valeur 1.

Composition de l'élément-heure

La composition du double-octet de l'élément-datation de type élément-heure est la suivante :

bit 15	bit 14	bit 13	bits 12 à 8	bits 7 à 4	bits 3 à 2	bits 1 à 0
1	1	/	Heure courante	Minute courante	Saison courante	Type

Le bit 14 de l'élément est à la valeur 1.

Le bit 13 de l'élément est inutilisé.

La valeur de l'heure courante est codée en binaire sur les bits 12 à 8.

La valeur de la minute courante est codée en binaire sur les bits 7 à 4. La valeur renseignée est indiquée en nombre de fraction de « Ta » minutes. La valeur renseignée prend donc des valeurs entre 0 et $(60/Ta - 1)$. Par exemple, si Ta vaut 10 minutes, la minute 40 sera renseignée par la valeur 4.

La saison courante (saison à laquelle appartient le mois en cours) est codé en binaire sur les bits 3 à 2 conformément au codage suivant :

bit 7	bit 6	SAISON
0	0	Saison n° 1
0	1	Saison n° 2
1	0	Saison n° 3
1	1	Saison n° 4

Rappel (cf. chapitre 2.3) : le numéro de la saison est codé de la manière suivante (1 pour été, 2 pour hiver, 4 pour pointe mobile).

Le type d'élément-heure indique la raison de l'insertion de cet élément-heure dans le tableau (événement déclencheur). Il est codé en binaire sur les bits 1 à 0 conformément aux informations du chapitre 2.5.3.4.

Composition de l'élément-date

La composition du double-octet de l'élément-datation de type élément-date est la suivante :

bit 15	bit 14	bit 13	bits 12 à 8	bits 7 à 4	bits 3 à 0
1	0	/	Jour courant	Mois courant	Année courante (unités)

Le bit 14 de l'élément est à la valeur 0.

Le bit 13 de l'élément est inutilisé.

La valeur du jour courant est codée en binaire sur les bits 12 à 8 (valeur de 0 à 31).

La valeur du mois courant est codée en binaire sur les bits 7 à 4 (valeur de 1 à 12).

La valeur des unités de l'année courante est codée en binaire sur les bits 3 à 0 (valeur de 0 à 9).

2.5.3.4. Gestion du tableau et disposition des différents éléments

Les éléments-datation sont insérés entre les deux éléments-puissance qui encadrent l'instant considéré lors de chacun des évènements suivants :

- chaque changement de poste tarifaire : le type de l'élément-heure est alors égal à 00,
- chaque début de jour civil : le type de l'élément-heure est alors égal à 00,
- chaque changement de saison : le type de l'élément-heure est alors égal à 00,
- chaque remise à l'heure de l'horloge interne de l'appareil (programmation externe) : le type de l'élément-heure est alors égal à 11 (le premier élément apparaît lors de la première échéance Ta minutes suivante),
- chaque changement d'heure légale (changement d'heure courante réalisé automatiquement par le compteur en fonction de ses programmations internes, lors du passage de l'heure d'été à l'heure d'hiver et du passage de l'heure d'hiver à l'heure d'été) : le type de l'élément-heure est alors égal à 11 (le premier élément apparaît lors de la première échéance Ta minutes suivante),
- lors du retour de la tension après une coupure supérieure à une heure : le type de l'élément-heure est alors égal à 10.

Cas particulier : gestion d'une coupure d'alimentation de durée supérieure à « Ta minutes »

Dans le cas d'une coupure d'alimentation d'une durée supérieure à « Ta minutes », les éléments-puissance de toutes les périodes « Ta minutes » entièrement comprises dans la coupure sont affectés d'une valeur de puissance moyenne nulle.

Au moment du retour de l'alimentation, le compteur met à jour le tableau des puissances moyennes. Pour cela, il effectue les opérations suivantes :

- il insère dans le tableau autant d'éléments-puissance que de périodes « Ta minutes » entièrement comprises dans la coupure et ces éléments seront affectés d'une valeur de puissance nulle dans les bits 10 à 0 et des valeurs respectives 0, 1 et 0 dans les bits 15, 14 et 13 pour signifier que les périodes correspondantes ont fait l'objet d'une « grande » coupure,
- en cas de coupure de durée supérieure à 1h, seuls les éléments-puissance de la première heure sont présents dans le tableau (nombres d'éléments-puissance insérés : 60 / Ta).

Cas particulier : gestion du mode contrôle

Le mode contrôle est un mode de fonctionnement particulier du compteur dont l'activation et l'usage sont réservés au distributeur d'électricité exploitant ce compteur afin de lui permettre de vérifier le bon fonctionnement des composantes matérielles effectuant la mesure de l'énergie et de la puissance.

Pendant toute la durée de fonctionnement en mode contrôle, le compteur insère dans le tableau des éléments-puissance affectés d'une valeur de puissance nulle dans les bits 10 à 0 et des valeurs respectives 0, 0 et 0 dans les bits 15, 14 et 13. De plus, lors de l'entrée et de la sortie du mode contrôle, le compteur insère dans le tableau deux éléments de datation : un élément-date et un élément-heure.

Cas particulier : les périodes tronquées

Des périodes « Ta minutes » peuvent être déclarées « tronquées » lors d'évènements particuliers : par exemple, lors d'un changement de mode (passage du mode standard vers le mode contrôle ou réciproquement) ou d'une intervention sur les date et heure. Les éléments-puissance correspondants encadrent généralement des éléments de datation. Ces périodes sont signalées dans le tableau par l'affectation des valeurs respectives 0, 1 et 1 dans les bits 15, 14 et 13.

Cas particulier du tableau incomplet

Lorsque que le compteur fait l'objet d'une programmation affectant des paramètres de gestion du tableau des puissances moyennes (période Ta, seuil de coupure D, ...), ce tableau peut faire l'objet d'une réinitialisation complète par le compteur. Dans ce cas, tous les éléments-puissance sont affectés de la valeur par défaut. Cette valeur correspond à une mise à la valeur 1 de l'ensemble des bits du double-octet.

Lors d'un relevé effectué peu après une programmation de ce type, le tableau peut contenir certains éléments (parmi les plus anciens) contenant cette valeur par défaut.

2.6. Informations de la période P en cours (code 02)

Ce tableau fournit les informations contractuelles et de consommation de la période contractuelle en cours, appelée « période P » sous la forme d'un ensemble de 61 octets définis dans le tableau suivant.

Les données transmises sont codées en binaire, à l'exception de la date et de l'heure codées en BCD. L'octet de poids faible de chaque double-octet est envoyé en tête lors de la transmission.

Données contractuelles et de consommation de la période en cours

Intitulé	Description de la donnée	Codage ou unité	Taille	Position du 1 ^{er} octet dans la chaîne
JJ, MM, AA	Jour, mois, an indiquant la date de début de la période P	BCD	3 octets	1
HH, MN	Heure, minute indiquant la date de début de la période P	BCD	2 octets	4
TARIF	Version tarifaire et poste tarifaire en cours lors du télé-relevé		1 octet	6
RE1	Energie active en Heures de Pointe Mobile (PM) pour la période P	kWh	3 octets	7
RE2	Energie active en Heures de Pointe (P) pour la période P	kWh	3 octets	10
RE3	Energie active en Heures Pleines d'Hiver pour la période P	kWh	3 octets	13
RE4	Energie active en Heures Creuses d'Hiver pour la période P	kWh	3 octets	16
RE5	Energie active en Heures Pleines d'Eté pour la période P	kWh	3 octets	19
RE6	Energie active en Heures Creuses d'Eté pour la période P	kWh	3 octets	22
RNIA	Durée totale de dépassement de la puissance souscrite dans la zone tarifaire A pour la période en cours P	mn	2 octets	25
RNIB	Durée totale de dépassement de la puissance souscrite dans la zone tarifaire B pour la période en cours P (utilisé seulement si dénivelé)	mn	2 octets	27
RNIC	Durée totale de dépassement de la puissance souscrite dans la zone tarifaire C pour la période en cours P (inutilisé)	mn	2 octets	29
RNID	Durée totale de dépassement de la puissance souscrite dans la zone tarifaire D pour la période en cours P (inutilisé)	mn	2 octets	31
RPMA	Puissance apparente maximale atteinte dans la zone tarifaire A pour la période en cours P	daVA	2 octets	33
RPMB	Puissance apparente maximale atteinte dans la zone tarifaire B pour la période en cours P (utilisé seulement si dénivelé)	daVA	2 octets	35
RPMC	Puissance apparente maximale atteinte dans la zone tarifaire C pour la période en cours P (inutilisé)	daVA	2 octets	37
RPMD	Puissance apparente maximale atteinte dans la zone tarifaire D pour la période en cours P (inutilisé)	daVA	2 octets	39
PSA	Puissance souscrite dans la zone tarifaire A pour la période en cours P	daVA	2 octets	41
PSB	Puissance souscrite dans la zone tarifaire B pour la période en cours P (utilisé seulement si dénivelé)	daVA	2 octets	43
PSC	Puissance souscrite dans la zone tarifaire C pour la période en cours P (inutilisé)	daVA	2 octets	45
PSD	Puissance souscrite dans la zone tarifaire D pour la période en cours P (inutilisé)	daVA	2 octets	47
KA	Coefficient de dépassement dans la zone tarifaire A pour la période en cours P	%	1 octet	49
KB	Coefficient de dépassement dans la zone tarifaire B pour la période en cours P (utilisé seulement si dénivelé)	%	1 octet	50
KC	Coefficient de dépassement dans la zone tarifaire C pour la période en cours P (inutilisé)	%	1 octet	51
KD	Coefficient de dépassement dans la zone tarifaire D pour la période en cours P (inutilisé)	%	1 octet	52

CP	Version tarifaire pour la période future P+1		1 octet	53
TFA	Temps de fonctionnement pendant la zone tarifaire A pour la période en cours P	heure	2 octets	54
TFB	Temps de fonctionnement pendant la zone tarifaire B pour la période en cours P (utilisé seulement si dénivelé)	heure	2 octets	56
TFC	Temps de fonctionnement pendant la zone tarifaire C pour la période en cours P (inutilisé)	heure	2 octets	58
TFD	Temps de fonctionnement pendant la zone tarifaire D pour la période en cours P (inutilisé)	heure	2 octets	60

Pour toutes les informations concernant la période contractuelle en cours P, la valeur fournie correspond à la valeur courante arrêtée au moment du relevé des informations.

Les informations TARIF et CP sont codées tel qu'indiqué pour le tableau des données des contrats au chapitre 2.3. Pour la période contractuelle en cours P, les 3 numéros de l'octet TARIF (saison, poste horaire et version tarifaire) sont renseignés. Pour la période contractuelle future P+1, seul le numéro de la version tarifaire de l'octet CP est renseigné.

Les informations de registres d'énergie RE1 à RE6 contiennent les index d'énergies actives tel qu'indiqué au chapitre 2.1.3.

Pour plus de précisions concernant les informations RNIA à RNID, RPMA à RPMD, PSA à PSD et KA à KD, il convient de se reporter au chapitre 2.1.4 sur le contrôle des dépassements de puissance.

Les informations TFA à TFD sont des registres comptabilisant le temps (en heures) pendant lequel le compteur a fonctionné dans chacune des différentes zones tarifaires A à D. Ces registres s'incrémentent de manière continue sur les différentes périodes contractuelles (P, P-1, P-2). Ils sont remis à zéro le 1^{er} octobre de chaque année à 2 heures du matin.

2.7. Informations des périodes précédentes P-1 et P-2 (code 01)

Ce tableau fournit les informations contractuelles et de consommation des 2 périodes contractuelles précédentes appelées « période P-1 » et « période P-2 » sous la forme d'un ensemble de 107 octets définis dans le tableau suivant.

Les données transmises sont codées en binaire, à l'exception de la date et de l'heure codées en BCD. L'octet de poids faible de chaque double-octet est envoyé en tête lors de la transmission.

Données contractuelles et de consommation des périodes précédentes

Intitulé	Description de la donnée	Codage ou unité	Taille	Position du 1 ^{er} octet dans la chaîne
JJ, MM, AA	Jour, mois, an indiquant la date de début de la période P-1	BCD	3 octets	1
HH, MN	Heure, minute indiquant la date de début de la période P-1	BCD	2 octets	4
TARIF	Version tarifaire et poste tarifaire en cours lors du télé-relevé		1 octet	6
RE1'	Energie active en Heures de Pointe Mobile (PM) pour la période P-1	kWh	3 octets	7
RE2'	Energie active en Heures de Pointe (P) pour la période P-1	kWh	3 octets	10
RE3'	Energie active en Heures Pleines d'Hiver pour la période P-1	kWh	3 octets	13
RE4'	Energie active en Heures Creuses d'Hiver pour la période P-1	kWh	3 octets	16
RE5'	Energie active en Heures Pleines d'Eté pour la période P-1	kWh	3 octets	19
RE6'	Energie active en Heures Creuses d'Eté pour la période P-1	kWh	3 octets	22
RNIA'	Durée totale de dépassement de la puissance souscrite dans la zone tarifaire A pour la période P-1	mn	2 octets	25
RNIB'	Durée totale de dépassement de la puissance souscrite dans la zone tarifaire B pour la période P-1 (utilisé seulement si dénivelé)	mn	2 octets	27
RNIC'	Durée totale de dépassement de la puissance souscrite dans la zone tarifaire C pour la période P-1 (inutilisé)	mn	2 octets	29

RNID'	Durée totale de dépassement de la puissance souscrite dans la zone tarifaire D pour la période P-1 (inutilisé)	mn	2 octets	31
RPMA'	Puissance apparente maximale atteinte dans la zone tarifaire A pour la période P-1	daVA	2 octets	33
RPMB'	Puissance apparente maximale atteinte dans la zone tarifaire B pour la période P-1 (utilisé seulement si dénivelé)	daVA	2 octets	35
RPMC'	Puissance apparente maximale atteinte dans la zone tarifaire C pour la période P-1 (inutilisé)	daVA	2 octets	37
RPMD'	Puissance apparente maximale atteinte dans la zone tarifaire D pour la période P-1 (inutilisé)	daVA	2 octets	39
PSA'	Puissance souscrite dans la zone tarifaire A pour la période P-1	daVA	2 octets	41
PSB'	Puissance souscrite dans la zone tarifaire B pour la période P-1 (utilisé seulement si dénivelé)	daVA	2 octets	43
PSC'	Puissance souscrite dans la zone tarifaire C pour la période P-1 (inutilisé)	daVA	2 octets	45
PSD'	Puissance souscrite dans la zone tarifaire D pour la période P-1 (inutilisé)	daVA	2 octets	47
RE1''	Energie active en Heures de Pointe Mobile (PM) pour la période P-2	kWh	3 octets	49
RE2''	Energie active en Heures de Pointe (P) pour la période P-2	kWh	3 octets	52
RE3''	Energie active en Heures Pleines d'Hiver pour la période P-2	kWh	3 octets	55
RE4''	Energie active en Heures Creuses d'Hiver pour la période P-2	kWh	3 octets	58
RE5''	Energie active en Heures Pleines d'Eté pour la période P-2	kWh	3 octets	61
RE6''	Energie active en Heures Creuses d'Eté pour la période P-2	kWh	3 octets	64
RNIA''	Durée totale de dépassement de la puissance souscrite dans la zone tarifaire A pour la période P-2	mn	2 octets	67
RNIB''	Durée totale de dépassement de la puissance souscrite dans la zone tarifaire B pour la période P-2 (utilisé seulement si dénivelé)	mn	2 octets	69
RNIC''	Durée totale de dépassement de la puissance souscrite dans la zone tarifaire C pour la période P-2 (inutilisé)	mn	2 octets	71
RNID''	Durée totale de dépassement de la puissance souscrite dans la zone tarifaire D pour la période P-2 (inutilisé)	mn	2 octets	73
RPMA''	Puissance apparente maximale atteinte dans la zone tarifaire A pour la période P-2	daVA	2 octets	75
RPMB''	Puissance apparente maximale atteinte dans la zone tarifaire B pour la période P-2 (utilisé seulement si dénivelé)	daVA	2 octets	77
RPMC''	Puissance apparente maximale atteinte dans la zone tarifaire C pour la période P-2 (inutilisé)	daVA	2 octets	79
RPMD''	Puissance apparente maximale atteinte dans la zone tarifaire D pour la période P-2 (inutilisé)	daVA	2 octets	81
PSA''	Puissance souscrite dans la zone tarifaire A pour la période P-2	daVA	2 octets	83
PSB''	Puissance souscrite dans la zone tarifaire B pour la période P-2 (utilisé seulement si dénivelé)	daVA	2 octets	85
PSC''	Puissance souscrite dans la zone tarifaire C pour la période P-2 (inutilisé)	daVA	2 octets	87
PSD''	Puissance souscrite dans la zone tarifaire D pour la période P-2 (inutilisé)	daVA	2 octets	89
KA'	Coefficient de dépassement dans la zone tarifaire A pour la période P-1	%	1 octet	91
KB'	Coefficient de dépassement dans la zone tarifaire B pour la période P-1 (utilisé seulement si dénivelé)	%	1 octet	92
KC'	Coefficient de dépassement dans la zone tarifaire C pour la période P-1 (inutilisé)	%	1 octet	93

KD'	Coefficient de dépassement dans la zone tarifaire D pour la période P-1 (inutilisé)	%	1 octet	94
KA''	Coefficient de dépassement dans la zone tarifaire A pour la période P-2	%	1 octet	95
KB''	Coefficient de dépassement dans la zone tarifaire B pour la période P-2 (utilisé seulement si dénivelé)	%	1 octet	96
KC''	Coefficient de dépassement dans la zone tarifaire C pour la période P-2 (inutilisé)	%	1 octet	97
KD''	Coefficient de dépassement dans la zone tarifaire D pour la période P-2 (inutilisé)	%	1 octet	98
CP	Option tarifaire pour la période future P+1		1 octet	99
TFA	Temps de fonctionnement pendant la zone tarifaire A pour la période en cours P	heure	2 octets	100
TFB	Temps de fonctionnement pendant la zone tarifaire B pour la période en cours P (utilisé seulement si dénivelé)	heure	2 octets	102
TFC	Temps de fonctionnement pendant la zone tarifaire C pour la période en cours P (inutilisé)	heure	2 octets	104
TFD	Temps de fonctionnement pendant la zone tarifaire D pour la période en cours P (inutilisé)	heure	2 octets	106

Pour toutes les informations concernant les périodes contractuelles précédentes, la valeur fournie correspond à la valeur courante à l'instant de la fin de la période concernée, c'est à dire à l'instant de l'opération appelée « glissement » (cf. chapitre 2.1.2).

Pour toutes les informations concernant la période contractuelle en cours P, la valeur fournie correspond à la valeur courante arrêtée au moment du relevé des informations.

Les informations TARIF et CP sont codées tel qu'indiqué pour le tableau des données des contrats au chapitre 2.3. Pour la période contractuelle en cours P, les 3 numéros de l'octet TARIF (saison, poste horaire et version tarifaire) sont renseignés. Pour la période contractuelle future P+1, seul le numéro de la version tarifaire de l'octet CP est renseigné.

Les informations de registres d'énergie RE1' à RE6' et RE1'' à RE6'' contiennent les index d'énergies actives tel qu'indiqué au chapitre 2.1.3.

Pour plus de précisions concernant les informations RNIA' à RNID', RPMA' à RPMD', PSA' à PSD', KA' à KD', RNIA'' à RNID'', RPMA'' à RPMD'', PSA'' à PSD'' et KA'' à KD'', il convient de se reporter au chapitre 2.1.4 sur le contrôle des dépassements de puissance.

Pour plus de précisions concernant les informations TFA à TFD, il convient de se reporter au chapitre 2.6 sur les informations de la période P en cours.

2.8. Liste des appels téléphoniques (code 07)

Ce tableau fournit des informations sur les appels téléphoniques récemment reçus par le compteur sous la forme d'un ensemble de 50 octets définis tel que décrit ci-après.

Ce tableau permet de connaître les instants où ont eu lieu les 10 derniers appels reçus par le compteur, quelle qu'en soit la provenance (Enedis, l'utilisateur du réseau ou un mandataire autorisé par celui-ci).

Pour chacun de ces appels, les informations sont mémorisées sous forme d'un élément dans un tableau de 10 éléments. Chaque élément est constitué de 5 octets comprenant la date et l'heure du début d'une communication. Le tableau est organisé suivant une pile de type FIFO (First In First Out). Chaque insertion d'un nouvel élément en tête du tableau provoque la perte de l'élément le plus ancien par sa sortie en queue de tableau.

Ce tableau contient les dates des communications initiées, c'est-à-dire celles qui se sont au moins concrétisées par un début de dialogue (au moins un décroché du compteur).

Chaque élément de 5 octets a le format suivant :

- 1^{er} octet : JOUR Jour du mois du début d'une communication. Information codée en BCD,
- 2^{ème} octet : MOIS Mois du début d'une communication. Information codée en BCD,
- 3^{ème} octet : HEURE Heure du début d'une communication. Information codée en BCD,
- 4^{ème} octet : MINUTE Minute du début d'une communication. Information codée en BCD,
- 5^{ème} octet : COM Compte rendu de l'appel.

Structure du cinquième octet COM constituant le compte-rendu de l'appel

Cet octet est destiné à donner des informations en logique combinatoire sur le déroulement de l'appel téléphonique reçu par le comptage :

- le bit n° 0 est égal à 1 si l'appel n'a pas été jusqu'à son terme,
- le bit n° 1 est égal à 1 si l'appel concernait un relevé,
- le bit n° 2 est égal à 1 si l'appel comprenait l'activation de la commande déverrouillage (réservé à Enedis),
- le bit n° 3 est égal à 1 si l'appel concernait une programmation (réservé à Enedis).

2.9. Valeurs de référence (code 05)

Ce relevé se résume à la transmission par le compteur d'un ensemble de 256 octets dans le seul but de vérifier que le compteur est en mesure de transmettre des données au système de traitement informatique. La structure des données transmises suit une progression arithmétique de valeur initiale égale à 0 et de raison 1.

Les valeurs des octets, codées en hexadécimal, sont les suivantes :

- 1^{er} octet : 00,
- 2^{ème} octet : 01,
- 3^{ème} octet : 02,
-
- 256^{ème} octet : FF.

3. **PARTIE 2 - Protocoles des couches de communication de la liaison téléphonique**

3.1. Généralités sur les communications du télé-relevé

3.1.1. Structure des échanges

Le télé-relevé fait appel à une structure simplifiée d'échanges de données à trois couches : une couche **Physique**, une couche **Liaison** et une couche **Session**. En effet les couches « présentation », « réseau » et « transport » n'ont pas de raison d'être dans cette procédure de télé-relevé pour les raisons suivantes :

- l'absence de conversions de données importantes et systématiques à réaliser,
- le système ou l'appareil de relevé automatique sont reliés au compteur par une simple liaison point à point ; la numérotation, la commutation du central téléphonique et le décrochage de l'appelé sont équivalents du point de vue logique à la phase d'établissement de la connexion physique ; une fois que cette connexion est établie, les informations transitent toutes par le même chemin et ne sont à aucun moment mémorisées dans un nœud intermédiaire.

Compte tenu des particularités du télé-relevé, la couche « application » telle que décrite dans la référence ISO 7498-1 se trouve intégrée ipso facto dans la couche session.

Les échanges au niveau de chaque couche sont décrits par des **transitions d'état** représentées sous forme de tableaux. La syntaxe utilisée pour la constitution de ces tableaux est définie par un **langage de spécification**. En cas de divergence d'interprétation entre le contenu d'un tableau de transitions d'état et une partie du texte d'accompagnement, il convient de considérer que le tableau constitue une référence qui prévaut sur le texte d'accompagnement.

Le réseau téléphonique commuté (RTC) est employé comme média de communication pour les échanges de données.

3.1.2. Vocabulaire de base et langage de spécification Élément-puissance

Toute communication fait intervenir deux équipements représentés par les expressions « système **Appelant** » et « système **Appelé** ». L'Appelant est le système qui décide d'initialiser une communication avec un équipement distant dit **Appelé** ; ces dénominations restent valables pendant toute la durée de vie de la communication.

Une communication est décomposée en un certain nombre de transactions. Chaque transaction se traduit par une émission de l'**Emetteur** vers le **Récepteur**. Au gré de l'enchaînement des transactions, les systèmes Appelant et Appelé jouent tour à tour le rôle d'Emetteur et de Récepteur.

Les termes **Maître** et **Esclave** caractérisent l'utilisation des commandes au niveau session, ainsi qu'un rôle différencié en mode test et en mode écho. Le Maître est l'appelant, l'esclave l'appelé.

Pour décrire sans ambiguïté le rôle de chaque couche du protocole mis en œuvre pour les échanges de données, la spécification utilise un **formalisme en tableau** modélisant le comportement réel par un **automate** à nombre fini d'états.

A chaque automate, correspond un unique **tableau logique** qui peut éventuellement être représenté sous la forme de plusieurs **tableaux physiques**. Ce découpage se justifie lorsque le tableau logique est particulièrement conséquent.

A chaque **occurrence d'automate** correspond une **instance** (copie active distincte) du tableau logique de l'automate de référence.

Chaque tableau physique est composé de lignes appelées **lignes d'état**. Chaque ligne d'état décrit la **condition de déclenchement** (colonne 2) pour que la machine passe d'un **état initial** (colonne 1) à un **état final** (colonne 4) en exécutant un **ensemble d'actions** (colonne 3).

Le premier état initial est l'**état de démarrage** de l'automate. Cet état est unique ; il est particularisé au moyen de l'attribut souligné.

Un **état d'arrêt** de l'automate est un état final pour lequel aucune ligne d'état n'est définie avec cet état comme état initial. Un **automate** est **infini** lorsqu'il ne possède aucun état d'arrêt. Un **automate fini** peut posséder un ou plusieurs états d'arrêt. Ces états sont également représentés avec l'attribut souligné. Compte tenu de cette convention, l'ordre dans lequel sont présentés les états dans un tableau physique n'a aucune importance.

Cette même règle s'applique lorsque plusieurs lignes d'état réfèrent le même état initial car les conditions de déclenchement sont toujours **exclusives** les unes des autres. L'ordre des lignes d'un tableau physique n'est

donc guidé que par de simples considérations de présentation. Il est cependant logique de commencer par décrire les transitions de l'état de démarrage.

Un ensemble d'actions d'une ligne d'état doit être considéré comme une **section critique** (c'est-à-dire une séquence non interruptible). Les actions qui y sont décrites **doivent être exécutées dans l'ordre séquentiel où elles sont écrites**. Une action est définie par un appel à une **procédure nommée** instanciée avec une liste de zéro, un ou plusieurs **paramètres** entre parenthèses. Toute procédure nommée référencée doit faire l'objet d'une description séparée. Il existe cependant deux actions prédéfinies : **l'affectation =** et **l'action vide \$none()** (absence d'action).

La condition de déclenchement associée à une ligne d'état peut éventuellement être composée de plusieurs sous-conditions. L'évaluation d'une condition de déclenchement composée passe toujours par l'évaluation de toutes les sous-conditions qu'elle contient. Ainsi, l'ordre d'écriture des sous-conditions est sans importance.

Les opérateurs supportés pour exprimer une condition composée sont d'une part les opérateurs logiques & (et logique), | (ou logique), not() (non logique) et d'autre part les opérateurs de comparaison (<, >, <=, >=, = et <>).

Il existe deux types de condition de déclenchement. Une condition de type simple est par définition évaluée instantanément. Elle peut éventuellement être composée mais, dans ce cas, toutes les sous-conditions sont de type simple. Une **fonction nommée booléenne** est un exemple de condition de type simple. Toute fonction nommée booléenne référencée doit faire l'objet d'une description séparée.

Une **condition de type événementiel** exprime l'attente d'un événement. Elle peut éventuellement être composée de plusieurs sous-conditions événementielles ou simples.

Lorsque l'évaluation d'une condition de déclenchement conduit à un résultat vrai, la condition se trouve **réalisée**. La réalisation d'une condition de déclenchement contient toujours une **transition d'état**. Un **événement** peut être défini comme étant un élément contribuant à la réalisation d'une condition de déclenchement de type événementiel.

Lorsqu'un événement est inclus dans une condition de déclenchement de type événementiel qui se trouve réalisée, il est automatiquement **consommé**. Un événement ne peut être consommé **qu'une seule fois**.

Tout événement, survenant lorsque l'occurrence d'automate qui est susceptible de le consommer se trouve dans un état où cette consommation est impossible, est stocké chronologiquement dans une zone appelée **file inter-automate**.

Ainsi, chaque automate dispose d'une unique file qu'il partage entre ses propres occurrences d'automate. La taille de cette file est supposée quasi-infinie ; son organisation et sa gestion ne sont pas décrites dans le présent document. Toutefois, il est indiqué qu'une **purge partielle** de la file (c'est-à-dire liée aux seuls événements concernant l'occurrence d'automate courante) est automatiquement effectuée pour toute transition d'état partant de l'état de démarrage.

Il doit exister également un mécanisme d'**auto-purge** conduisant à la suppression automatique des événements entrants et manifestement non consommables. En outre, il existe une procédure nommée prédéfinie \$purge() qui correspond à l'action de **purge totale** de la file inter-automate courante. Toutes les occurrences de l'automate correspondant se retrouvent alors dans l'état de démarrage.

La **production** d'événements est assurée par certaines des actions décrites dans un ensemble d'actions associé à une ligne d'état. Un **événement interne** ne peut être consommé que par l'automate qui l'a produit. Un **événement externe** est toujours consommé par un autre automate que celui qui l'a produit.

A noter que **l'absence d'un événement**, exprimée par une sous-condition de type événementiel encapsulée dans l'opérateur logique not(), est toujours une sous-condition de type simple.

Lorsque, pour un état initial, il existe une ligne d'état où la condition de déclenchement est d'un certain type, alors toutes les lignes d'état du même état doivent posséder des conditions de déclenchement du même type.

Lorsque ce type est simple, l'état initial est appelé **sous-état**. Un sous-état est particularisé au moyen de l'attribut itاليque. Il est **transitoire** et peut toujours être éliminé. Sa présence dans un tableau physique n'est justifiée que par un souci de clarté de la présentation. Dans le cas particulier d'un sous-état de démarrage, une condition particulière a été prédéfinie ; il s'agit de la condition \$true() qui est toujours vraie.

Enfin, les **variables** référencées dans les conditions de déclenchement et les actions décrites dans un tableau physique restent **locales** à chaque **occurrence d'automate**. Il existe également une variable prédéfinie (la **variable non liée**) destinée à remplacer tout paramètre inexploité dans n'importe quelle fonction ou procédure nommée.

USART : Universal Synchronous Asynchronous Receiver Transmitter (identifié à **Sérialisateur** dans cette spécification)

CRC : Cyclical Redundancy Check. Sa mise en œuvre permet de détecter toutes les erreurs simples, doubles ou triples dans des blocs de données de 128 octets, ainsi que toutes les erreurs sur des suites de 16 bits

SPDU : Session Protocol Data Unit

SSDU : Session Service Data Unit

SPDU : Data Signal Display Unit

En l'absence de mention particulière concernant la valeur des octets, dans le cas des champs ou variables dont la taille dépasse un octet, les octets de poids le plus faible sont placés en tête.

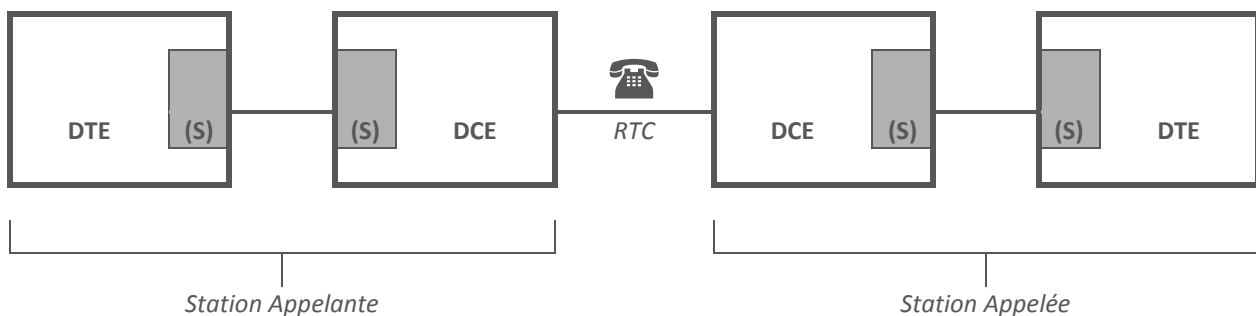
3.2. Couche Physique

3.2.1. Généralités sur la Couche Physique

Les équipements (compteurs et système de traitement informatique) échangent leurs données à travers le réseau téléphonique commuté public grâce à des modems semi-duplex. L'établissement de la communication se fait en appel/réponse automatique, conformément à l'avis V25 ou V25bis de l'Union Internationale des Télécommunications (UIT).

3.2.2. Description de l'environnement V24/V28

Un système communiquant sur réseau téléphonique commuté public peut-être modélisé comme indiqué sur la figure suivante :



- DTE** Data Terminal Equipment (**ordinateurs**, terminaux ou imprimantes). Un DTE est un système qui envoie ou reçoit des informations. Certains manuels le nomment aussi **ETTD** (Equipement Terminal de Traitement de Données).
- DCE** Data Communication Equipment (interface communicante ou **modem**). Un DCE est un équipement utilisé comme interface sur un réseau de communication. Son rôle est de transférer les données. Il est aussi nommé **ETCD** (Equipement de Terminaison du Circuit de Données).
- (S)** Un sérialisateur est généralement un circuit intégré qui synchronise l'émission et la réception de bits.
La fonction « transmetteur » convertit des octets en une suite de 8 bits en série. La fonction « récepteur » convertit une série de 8 bits en un octet.

Rappel des principaux circuits de l'interface V24

Circuit	Désignation	Abréviation	DTE/DCE *
Terre			
101	Terre de Protection	TP	-
102	Terre de Signalisation	TS	-
Signaux de données			
103	Emission des Données	ED	S/E
104	Réception des Données	RD	E/S
Signaux de contrôle			
105	Demande Pour Emettre	DPE	S/E
106	Prêt A Emettre	PAE	E/S
107	Poste de Données Prêt (modem prêt)	PDP	E/S
108	Terminal de Données Prêt	TDP	S/E
Autres signaux de contrôle			
109	Détection de Signal	DS	E/S
125	Indicateur de Sonnerie	IS	E/S
Signaux d'horloge			
113	horloge émetteur (DTE source)	-	S/E
114	horloge émetteur (DCE source)	-	E/S
115	horloge récepteur (DCE source)	-	E/S
Notes :			
S ≡ le circuit défini est une sortie			
E ≡ le circuit défini est une entrée			

3.2.3. Protocole de la couche *Physique*

Le protocole de la couche *Physique* est conçu pour un fonctionnement asynchrone (étant donné le faible débit) en mode semi-duplex (une ligne bifilaire et une seule fréquence porteuse). Il est identique pour l'Appelant et pour l'Appelé (comportement totalement symétrique).

Ce protocole comprend :

- une interface de communication conforme aux avis V24 et V28 de l'UIT ainsi qu'à la norme ISO 2110 ;
- un échange systématique de jeton par échange de porteuse.

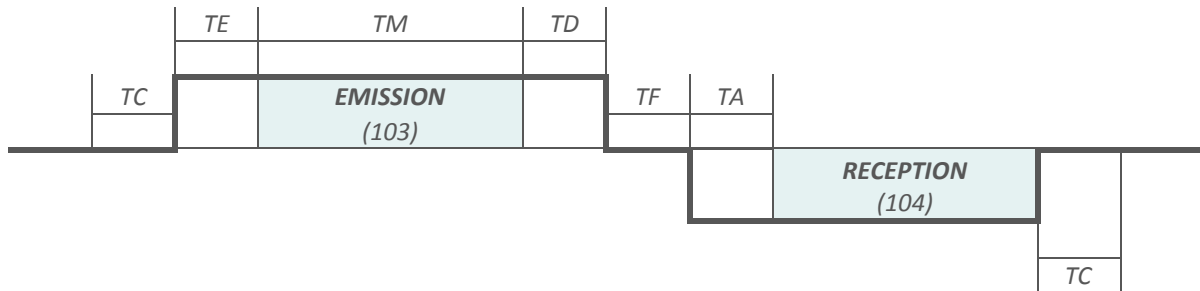
Le protocole de la couche *Physique* agit directement sur le sérialisateur relié au modem. Le terminal et le modem communiquent à travers l'interface V24 avec un débit d'échange de 1200 bits/s.

Le modus operandi de la transmission asynchrone consiste en un regroupement par paquets de 8 des bits transmis, sans parité (octets non codés en ASCII). Chaque caractère comprend un bit de *start* en tête et un bit de *stop* en queue, qui correspondent respectivement à une mise à 0 et une mise à 1 du signal de données en sortie du sérialisateur. Ces deux bits font office de

"points de synchronisation". Le moment *start* permet essentiellement de synchroniser les horloges alors que le moment *stop* laisse un temps suffisant au récepteur pour mettre en parallèle le caractère avant de traiter le suivant.

3.2.4. Diagramme des temps

La figure suivante schématise les temporisations et les réveils utilisés pour la gestion des échanges en régime permanent



Chaque trame est concrétisée par l'émission d'une porteuse non modulée d'une durée minimum *TE*, suivie par la transmission d'un bloc de données d'une durée *TM* bornée et d'une temporisation *TD* avant la coupure de la porteuse (afin d'éviter tout risque de perte du dernier octet émis).

TC : temps de coupure de porteuse à l'alternat. Ce temps est contrôlé par un réveil.
TC = 40 ms.

TE : durée d'émission de la porteuse non modulée. Cette durée est une temporisation.
TE = 300 ms.

TM : durée de modulation pour l'émission des données. Cette durée dépend de la taille de la trame à émettre (comprise entre 4 et 127 octets) et de la vitesse de transmission.

TD : durée d'attente avant la coupure de la porteuse. Cette durée est une temporisation équivalente au minimum à la transmission de 2 octets.
TD = 20 ms.

TF : durée de sécurité entre l'émission et la réception. Cette durée est une temporisation.
TF = 150 ms.

TA : durée d'attente de stabilisation de la porteuse après détection du signal DCD. Cette durée est une temporisation qui permet d'éliminer les caractères parasites.
TA = 60 ms.

Il existe également quatre réveils (*TB*, *TR*, *TR1* et *TCM*) non représentés sur cette figure.

TB : temps de réception maximum admissible. Ce temps est contrôlé par un réveil armé du côté récepteur lorsque la porteuse est détectée et désarmé au passage en émission.
TB = 2 s.

TR : délai d'attente maximum de retour de la porteuse après une émission. Ce temps est contrôlé par un réveil.
TR = 500 ms. La valeur de *TR* a été spécifiée initialement à 200 ms, mais c'est la valeur 500 ms qui est utilisée par les applications existantes.

TR1 : temps de réponse maximum admissible entre d'une part la montée du signal 105 par le DTE et la montée du signal 106 par le DCE, et d'autre part la descente du signal 108 par le DTE et la descente du signal 107 par le DCE. Ce temps est contrôlé par un réveil.

TR1 = 1 s.

TCM : durée maximale d'une communication. Ce temps est contrôlé par un réveil.

TCM = 10 minutes.

3.2.5. Services et primitives de service de la couche Physique

L'utilisateur du protocole de la couche **Physique** dispose des services et primitives de service suivants :

Services	Primitives de service
Phy_DATA	Phy_DATA.req(Frame) Phy_EndDATA.ind(Frame) Phy_DATA.ind(Frame, ErrpliP)
Phy_ABORT	Phy_ABORT.req() Phy_ABORT.ind(ErrorNb)

Le rôle attribué à chaque primitive est le suivant :

Phy_DATA.req(Frame) permet à la couche **Liaison** de demander à la couche **Physique** l'émission d'une trame Frame ;

Phy_EndDATA.ind() permet à la couche **Physique** d'informer la couche **Liaison** de la fin de l'émission de la dernière trame ;

Phy_DATA.ind(Frame, ErrpliP) permet à la couche **Physique** d'informer la couche **Liaison** qu'une trame Frame est disponible, avec éventuellement une erreur détectée à la réception (ErrpliP différent de 0) ;

Phy_ABORT.req() permet à la couche **Liaison** de demander à la couche **Physique** de mettre fin à son activité ;

Phy_ABORT.ind(ErrorNb) permet à la couche **Physique** d'informer la couche **Liaison** de l'occurrence d'une erreur fatale repérée par le numéro ErrorNb.

3.2.6. Paramètres de la couche Physique

Les valeurs des temporisations et des réveils ont déjà été définies dans le diagramme des temps.

La valeur de la taille maximale **MaxIndex** d'une trame en réception est fixée à 127.

Le nombre maximum d'émissions successives sans réponse **MaxErrpj** est fixé à 7.

3.2.7. Transitions d'état

La machine d'état de l'Appelant est strictement identique à celle de l'Appelé. Les temps sont exprimés en millisecondes dans le tableau suivant (les 2 autres tableaux qui lui font suite donnent respectivement les définitions des procédures « ensemble d'actions » et la signification du terme « Etat » utilisé dans la première colonne et la dernière colonne de ce tableau).

Transitions d'état décrivant le protocole de la couche Physique

Etat initial	Cond. de déclenchement	Ensemble d'actions	Etat final
<i>Init</i>	\$true()	MaxErrpj = 7 MaxIndex =127 TD = 20 TC = 40 TA = 60 TF = 150 TR = 200 TE = 300 TR1 = 1000 TB = 2000 TCM = 600000	Stopped
Stopped	Connection_event & Master	Errpj = 0 ErrpliP = 0 init_timer(TCM) set(RTS) init_timer(TR1)	W.CTS
Stopped	Connection_event & not(Master)	Errpj = 0 ErrpliP = 0 init_timer(TCM) init_timer(TR) wait_time(TF)	W.DCD
W.CTS	CTS_activation_event	stop_timer(TR1) wait_time(TE)	<i>M.Send</i>
W.CTS	Phy_ABORT.req()	stop_timer(TCM) stop_timer(TR1) clear(RTS) clear(DTR) init_timer(TR1)	W.DSR
W.CTS	time_out(TCM)	Phy_ABORT.ind(EP-R7F) stop_timer(TR1) clear(RTS) clear(DTR) init_timer(TR1)	W.DSR
W.CTS	time_out(TR1)	Phy_ABORT.ind(EP-R6F) stop_timer(TCM) clear(RTS) clear(DTR) init_timer(TR1)	W.DSR
W.DCD	DCD_activation_event	stop_timer(TR) init_timer(TB) Errpj = 0 Index = 1 Frame="" Veloc = FAUX wait_time(TA)	Receiving

W.DCD	time_out(TR) & Errpj < 6	Errpj = Errpj + 1 set(RTS) init_timer(TR1)	W.CTS
W.DCD	time_out(TR) & Errpj >= 6	Phy_ABORT.ind(EP-R5F) stop_timer(TCM) clear(DTR) init_timer(TR1)	W.DSR
W.DCD	Phy_ABORT.req()	stop_timer(TR) stop_timer(TCM) clear(DTR) init_timer(TR1)	W.DSR
W.DCD	time_out(TCM)	Phy_ABORT.ind(EP-R7F) stop_timer(TR) clear(DTR) init_timer(TR1)	W.DSR
M.Send	Phy_Data.req(Frame)	Size = size(Frame) Index=1 send_octet(Frame, Index)	Sending
M.Send	not(Phy_Data.req(Frame))	clear(RTS) init_timer(TR) wait_time(TF)	W.DCD
Sending	octet_sent_event & Index < Size	Index=Index+1 send_octet(Frame, Index)	Sending
Sending	octet_sent_event & Index >= Size	wait_time(TD) Phy_EndData.ind() clear(RTS) init_timer(TR) wait_time(TF)	W.DCD
Sending	Phy_ABORT.req()	stop_timer(TCM) clear(DTR) init_timer(TR1)	W.DSR
Sending	time_out(TCM)	Phy_ABORT.ind(EP-R7F) clear(DTR) init_timer(TR1)	W.DSR
Receiving	octet_received_event & Index <= MaxIndex & Veloc =FALSE	Index = Index +1 read_data(RecB) concat(Frame, RecB)	Receiving
Receiving	octet_received_event & Index > MaxIndex & Veloc =FALSE	read_data(RecB) Veloc =TRUE	Receiving
Receiving	octet_received_event & Veloc =TRUE	read_data(RecB)	Receiving
Receiving	DCD_inactivation_event	init_timer(TC) Veloc = TRUE	M.RecC

Receiving	time_out(TB)	Phy_ABORT.ind(EP-R4F) stop_timer(TCM) clear(DTR) init_timer(TR1)	W.DSR
Receiving	Phy_ABORT.req()	stop_timer(TB) stop_timer(TCM) clear(DTR) init_timer(TR1)	W.DSR
Receiving	time_out(TCM)	Phy_ABORT.ind(EP-R7F) stop_timer(TB) clear(DTR) init_timer(TR1)	W.DSR
Receiving	err_USART_evt	ErrpliP = ErrpliP + EP_R1 Veloc = TRUE	Receiving
M.RecC	DCD_activation_event & not(time_out(TC))	stop_timer(TC)	Receiving
M.RecC	time_out(TC) & Index < 4	stop_timer(TB) set(RTS) init_timer(TR1) ErrpliP = 0	W.CTS
M.RecC	time_out(TC) & Index >= 4	Phy_DATA.ind(Frame, ErrpliP) stop_timer(TB) set(RTS) init_timer(TR1) ErrpliP = 0	W.CTS
M.RecC	Phy_ABORT.req()	stop_timer(TC) stop_timer(TB) stop_timer(TCM) clear(DTR) init_timer(TR1)	W.DSR
M.RecC	time_out(TCM)	Phy_ABORT.ind(EP-R7F) stop_timer(TC) stop_timer(TB) clear(DTR) init_timer(TR1)	W.DSR
M.RecC	time_out(TB)	Phy_ABORT.ind(EP-R4F) stop_timer(TC) stop_timer(TCM) clear(DTR) init_timer(TR1)	W.DSR
W.DSR	DSR_inactivation_event	stop_timer(TR1) set(DTR)	Stopped
W.DSR	time_out(TR1)	set(DTR)	Stopped

Définition des procédures, des fonctions et des événements classés dans l'ordre alphabétique

Procédure, fonction ou événement	Définition
clear(RTS)	désactivation du circuit 105 du modem
clear(DTR)	désactivation du circuit 108 du modem
concat(Frame, RecB)	concaténation de l'octet RecB dans la trame Frame en cours de constitution
connection_event	événement signalant que la communication téléphonique est établie
CTS_activation_event	événement issu du modem informant de l'activation du circuit 106
DCD_activation_event	événement issu du modem informant de l'activation du circuit 109
DCD_inactivation_event	événement issu du modem informant de la désactivation du circuit 109
DSR_inactivation_event	événement issu du modem informant de la désactivation du circuit 107
err_USART_evt	événement issu de l'USART signalant un problème
(Phy_Data.req(Frame))	consommation d'un événement de type Phy_DATA.req(Frame)
init_timer(TB), init_timer(TR), init_timer(TR1) ou init_timer(TC), init_timer(TCM)	armement du réveil TB, TR, TR1, TC ou TCM
octet_received_event	événement informant qu'un octet a été reçu
octet_sent_event	événement informant qu'un octet a été émis
read_data(RecB)	traitement de l'événement octet_received_event par lecture de l'octet RecB reçu
send_octet(Frame, Index)	Emission de l'octet de rang Index dans la trame Frame
set(RTS)	activation du circuit 105 du modem
set(DTR)	activation du circuit 108 du modem
size(Frame)	calcul du nombre d'octet de la trame Frame
stop_timer(TB), stop_timer(TR), stop_timer(TR1) stop_timer(TC) ou stop_timer(TCM)	désarmement du réveil TB, TR, TR1 TC ou TCM
time_out(TB), time_out(TR), time_out(TR1) time_out(TC), ou time_out(TCM)	déclenchement du réveil TB, TR, TR1, TC ou TCM
wait_time(TA), wait_time(TE), wait_time(TD) ou wait_time(TF)	temporisation pendant le temps TA, TE, TD ou TF

Précisions concernant les fonctions wait_time() :

Pendant les attentes des fonctions wait_time(), les événements affectant la liaison série ne sont pas traités, mais sont mémorisés. Ils sont traités (ou rejetés) lors des traitements ultérieurs. Dans une approche « multi-threads », il est possible d'implémenter ces attentes sous la forme d'un « blocage » de l'automate.

Précisions concernant la variable VELOC :

Cette variable a été définie dans la spécification initiale et conservée pour des raisons historiques. Le rôle de cette variable VELOC est de verrouiller l'accès au buffer de réception de la couche physique lorsque l'état est incompatible avec la réception de caractères :

- la porteuse n'est pas présente (DCD absent),
- le nombre de caractères reçus est supérieur à la capacité du buffer (il y a des parasites, ou un problème de fonctionnement important, ...),
- un caractère a fait l'objet d'une signalisation d'erreur (ou le driver série).

Cette variable est mise à l'état "FAUX" lors de la détection de porteuse et du passage en réception. Il convient de rester ensuite en réception jusqu'à la détection de la chute de cette porteuse, et VELOC peut alors être mis à l'état "VRAI", si une erreur est signalée par l'UART ou si le nombre maximum de caractères a été reçu. Les caractères ne sont mémorisés que si VELOC est à l'état "VRAI" (et que l'on est dans l'état "RECEIVING").

Remarque : il est possible, pour éliminer cette variable, d'introduire un état "REC_ERR", dans lequel l'automate passe en cas de détection d'une anomalie.

Signification des « états finaux ».

Etat	Signification
<i>Init</i>	état d'initialisation
Stopped	état de démarrage commun à l'Appelant et à l'Appelé
W.CTS (Wait for Clear To Send)	état de l'Emetteur en attente de la montée du circuit 106 après la montée du circuit 105, sous le contrôle du réveil TR1
W.DCD (Wait for Data Carrier Detect)	état d'attente de la montée du circuit 109, sous le contrôle du réveil TR
W.DSR (Wait for Data Set Ready)	état d'attente de la descente du circuit 107 sous le contrôle du réveil TR1
<i>M.Send (Must Send)</i>	état transitoire de l'Emetteur, pour tester s'il y a une trame à envoyer
Sending	état récurrent de l'Emetteur : émission d'un octet à la fois
Receiving	état du Récepteur. Pour rester dans cet état, le circuit 109 doit être monté en permanence
M.RecC (Must Receive but DCD Cut)	descente du circuit 109 en cours de réception

3.2.8. Répertoire et traitement des erreurs

Les erreurs sont répertoriées par le codage **EP-RNF** ou **EP-ENF** portant les significations suivantes :

EP	erreur de la couche Physique
-R	erreur en réception
-E	erreur en émission
N	numéro de l'erreur
F	erreur fatale

Tableau récapitulatif des erreurs

EP-R1	Erreur matériel détectée au niveau de l'USART
	Cette erreur conduit la couche Physique à en informer la couche Liaison et à ne pas mémoriser les octets suivants de la trame en cours
EP-R4F	Time-out bavard : durée de réception trop longue
	Cette erreur conduit à réinitialiser la couche Physique après avoir informé la couche Liaison
EP-R5F	7 émissions successives sans réponse (7 jetons ou 1 de données et 6 jetons)
	Cette erreur conduit à réinitialiser la couche Physique après avoir informé la couche Liaison
EP-R6F	Blocage du modem constaté après l'échéance du réveil TR1
	Cette erreur conduit à réinitialiser la couche Physique après avoir informé la couche Liaison
EP-R7F	Echéance du réveil TOCM
	Cette erreur conduit à réinitialiser la couche Physique après avoir informé la couche Liaison

Les erreurs non fatales sont remontées localement grâce à la primitive DL_DATA.ind. Toute occurrence de l'une des erreurs fatales est remontée localement grâce à la primitive de service Phy_ABORT.ind. La liste complète des numéros d'erreur fatale est fournie au chapitre 3.5.

3.3. Couche Liaison

3.3.1. Généralités sur la Couche Liaison

Le protocole de la couche **Liaison** est conçu pour supporter une transmission bidirectionnelle des données. Comme cette couche assure le contrôle du transfert correct des données, son fonctionnement en 2 modes (en mode ECHO et en mode normal) est spécifié dans les sous-chapitres suivants.

3.3.2. Couche Liaison en mode normal

3.3.2.1. Protocole Liaison en mode normal

En mode normal, le protocole est strictement identique pour l'Appelant (le Maître) et pour l'Appelé (l'esclave).

La couche **Liaison** en mode normal est chargée de transformer le canal physique exploité par la couche **Physique** en canal logique apte à transmettre l'information avec fiabilité. Ses fonctions principales sont :

- synchroniser les trames en émission et en réception,
- assurer une protection efficace contre les erreurs de transmission.

La couche **Liaison** doit régler tous les problèmes courants causés par les défauts du canal physique à l'exception de la coupure matérielle de la liaison.

La communication entre stations peut être divisée en trois phases:

- l'établissement du circuit réalise la connexion physique des stations au réseau téléphonique public commuté. Cette fonction est assurée par Physique1, non décrit dans ce document ;
- le transfert d'information est la phase utile de la transmission avec échange de trames d'information et d'acquiescement ;
- la terminaison du circuit libère la ligne téléphonique. Cette fonction est également assurée par Physique1.

On notera que ne sont pas évoquées ici, les phases d'ouverture de la liaison (reconnaissance des stations et négociation des conditions de l'échange) et de fermeture de la liaison (libération des stations). Le protocole **Liaison** est donc **orienté sans connexion**.

Deux variables globales, TestS et TestM, dont les valeurs sont gérées en dehors du protocole, permettent de définir un traitement des trames différent lorsque le mode test est opérationnel. Sur le Maître, TestM est à VRAI si l'équipement est en mode test, à FAUX sinon ; TestS est toujours à FAUX. Sur l'esclave, TestS est à VRAI si l'équipement est en mode test, à FAUX sinon ; TestM est toujours à faux.

Les traitements à effectuer en mode test sont les suivants :

- sur la station Esclave, l'octet Errpli est ajouté aux données, dans le champ Test, avant le calcul de CRC et l'envoi de la trame ;
- sur la station Maître, le champ Test est extrait et placé dans la variable CodeTest.

3.3.2.2. Procédure "envoyer et attendre"

Le traitement des défauts de ligne est effectué suivant la procédure "envoyer et attendre". Voici le mode de fonctionnement de cette procédure :

- à toute trame devant être émise, sont ajoutés deux octets de CRC (Cyclical Redundancy Check) ;
- toute trame reçue fait l'objet d'un contrôle de CRC. Si la trame est valide, un acquiescement positif est envoyé. Dans le cas contraire, c'est un acquiescement négatif qui est émis ;
- un réveil est armé à chaque envoi. Son échéance sans réception d'acquiescement est équivalente à la réception d'un acquiescement négatif ;
- pour éviter de possibles duplications suite à des retransmissions, le principe de la numérotation des trames est retenu.

3.3.2.3. Services et primitives de service de Liaison

L'utilisateur du protocole **Liaison** dispose des services et primitives de service suivants :

Service	Primitive
DL_DATA	DL_DATA.req(DSDU) DL_EndDATA.ind() DL_DATA.ind(DSDU)
DL_ABORT	DL_ABORT.req() DL_ABORT.ind(ErrorNb)

Voici le rôle attribué à chaque primitive :

DL_DATA.req(DSDU) permet à la couche **Session** de demander à la couche **Liaison** le transfert d'un paquet de données DSDU ;

DL_EndDATA.ind() permet à la couche **Liaison** d'informer la couche **Session** du bon envoi du dernier paquet de données ;

DL_DATA.ind(DSDU) permet à la couche **Liaison** d'informer la couche **Session** de l'arrivée d'un paquet de données DSDU ;

DL_ABORT.req() permet à la couche **Session** de demander à la couche **Liaison** de mettre fin à son activité ;

DL_ABORT.ind(ErrorNb) permet à la couche **Liaison** d'informer la couche **Session** de l'occurrence d'une erreur fatale repérée par le numéro ErrorNb.

3.3.2.4. Description des trames

Le protocole **Liaison** est une procédure de ligne orientée bit c'est-à-dire qu'il dispose l'information dans des trames dont les champs contiennent le texte (à savoir les données en provenance de la couche **session**) ainsi que les bits de contrôle associés. Les trames contiennent un nombre entier d'octets inférieur ou égal à 127. Comme tous les champs ont une taille fixe sauf le champ **Text**, il suffit de disposer d'un champ **Size** pour définir la longueur réelle de chaque trame.

Dans le protocole **Liaison**, il n'y a qu'un seul type de trame défini par les six champs suivants :

Size	: 1 octet
FrameType	: 4 bits
NSEQ	: 4 bits
Text (DSDU)	: 0 à 122 octets
Test	: 1 octet si présent (en mode test)
BCC	: 2 octets

1 octet	4 bits	4 bits	0 à 123 octets	2 octets	
Size	Type	NSEQ	Text	Test	BCC

Le champ **Size** correspond au nombre total d'octets dans la trame.

Le champ **Type** est codé "0000" pour une trame de données, "0110" pour un acquittement positif, "1011" pour un acquittement négatif.

Le champ **NSEQ** référence le numéro de la séquence en cours ; une séquence correspond à l'échange d'une trame de données et de l'acquittement en réponse. Il varie de 0 à 15.

Le champ **Text** n'est présent que dans les trames de données.

Le champ **Test** n'est présent qu'en mode test, dans le sens Esclave-Maître.

Enfin, le champ **BCC** (Block Check Characters) correspond aux 16 bits de redondance du CRC dont le principe est exposé au chapitre 3.6.

3.3.2.5. Gestion des échanges

Le protocole **Liaison** est symétrique, à part le traitement du mode test. Les systèmes Appelant et Appelé sont tour à tour Emetteur et Récepteur (cette dernière terminologie est la plus représentative de la gestion des échanges).

Après l'envoi d'une trame de donnée, la couche **Liaison** du côté Emetteur attend toujours une trame de contrôle venant de la couche **Liaison** du Récepteur. Cette attente est contrôlée par le réveil **TL**. L'échéance de TL est interprétée comme la réception d'un acquittement négatif.

Après l'envoi d'une trame de donnée et la réception d'un acquittement négatif de la trame précédemment envoyée, il y a ré-émission de la trame courante. Le nombre de ré-émissions est limité à MaxErrl. Au-delà de ce nombre, il y a arrêt de la communication au niveau **Liaison** et la couche **Session** en est informée.

A chaque réception d'une trame de donnée numérotée N par un des systèmes, il y a émission immédiate d'une trame de contrôle en réponse, portant le même numéro N.

Après l'échange d'une trame de données numérotée N, acquittée positivement, la prochaine trame de données portera le numéro N+1, quel que soit le sens de son échange.

L'alternance du sens d'échange des trames de données s'opère de façon quelconque.

3.3.2.6. Paramètres de Liaison

Le temps d'attente maximum TL, par l'Emetteur, de la trame de contrôle en retour du Récepteur est calculé sur la base de 6 pertes de jeton.

$$TL > 5(TR + TE) + TR = 2700ms \text{ d'où } TL \text{ fixé à } 3 \text{ s.}$$

La valeur du nombre MaxErrl d'envois d'une même trame provoquant la déconnexion est fixée à 8 ; le huitième envoi n'est pas effectué.

3.3.2.7. Transitions d'état

La machine d'état de l'Appelant est strictement identique à celle de l'Appelé. Les deux systèmes sont tour à tour Emetteur et Récepteur de DSDU. A tout instant, il n'existe qu'une seule occurrence de cet automate dans chaque équipement. Les temps sont exprimés en millisecondes.

Transitions d'état de Liaison

Etat initial	Cond. de déclenchement	Ensemble d'actions	Etat final
<i>Init</i>	\$true()	MaxErrl = 8 Errl = 1 Vseq = 1 Ack_expected= FALSE Errpli = 0 CodeTest = 0 TL = 3300	Stopped
Stopped	DL_DATA.req(DSDU) &	Type="0000"B Text=DSDU Size=size(Text) + 4 Fr=concat(Size, Type, VSeq, Text,_) Phy_DATA.req(concat_Fr(Fr, crc(Fr))) Ack_expected=TRUE Errl = Errl + 1	M.RecPE

Stopped	Phy_DATA.ind(Frame, ErrpliP) & not(check_frame(Frame, ErrpliP))	Errpli = Errpli + ErrpliP + EL_R1 Type = "1011"B	M.SendCr
Stopped	Phy_DATA.ind(Frame, ErrpliP) & check_frame(Frame, ErrpliP) & is_text(Frame) & (Nseq = Vseq Nseq = Vseq - 1)	\$none()	T.NbD
M.RecPE	Phy_EndDATA.ind() & Ack_expected=TRUE	Errpli = 0 Init_timer(TL)	M.RecCr
M.RecPE	Phy_EndDATA.ind() & Ack_expected=FALSE & Type = "0110" & not(check_EL2(Errpli))	DL_DATA.ind(extract_text (Frame) Errpli = 0 Vseq = Vseq + 1	Idle
M.RecPE	Phy_EndDATA.ind() & Ack_expected=FALSE & Type = "0110"& check_EL2(Errpli)	Errpli = 0 Vseq = Vseq + 1	Idle
M.RecPE	Phy_EndDATA.ind() & Ack_expected=FALSE & Type = "1011"	Errpli = 0	Idle
M.RecPE	DL_ABORT.req()	Errl = 1 Vseq = 1 Ack_expected= FALSE Phy_ABORT.req()	Stopped
M.RecPE	Phy_ABORT.ind(ErrorNb)	Errl = 1 Vseq = 1 Ack_expected= FALSE DL_ABORT.ind(ErrorNb)	Stopped
M.RecCr	Phy_DATA.ind(Frame, ErrpliP) & check_frame(Frame, ErrpliP) & not(is_nack(Frame))	stop_timer(TL) extract_test(Frame, TestM, CodeTest)	T.NbCr
M.RecCr	Phy_DATA.ind(Frame, ErrpliP) & check_frame(Frame, ErrpliP) & is_nack(Frame) & Errl < MaxErrl	stop_timer(TL) extract_test(Frame, TestM, CodeTest) Phy_DATA.req(Fr) Errl=Errl +1	M.RecPE
M.RecCr	Phy_DATA.ind(Frame, ErrpliP) & check_frame(Frame, ErrpliP) & is_nack(Frame) & Errl >= MaxErrl	Errl = 1 Vseq = 1 Ack_expected= FALSE stop_timer(TL) extract_test(Frame, TestM, CodeTest) DL_ABORT.ind(EL-E4F) Phy_ABORT.req()	Stopped
M.RecCr	Phy_DATA.ind(Frame, ErrpliP) & not(check_frame(Frame, ErrpliP)) & Errl < MaxErrl	Errpli = Errpli + EL_R1 stop_timer(TL) Phy_DATA.req(Fr) Errl=Errl +1	M.RecPE

M.RecCr	Phy_DATA.ind(Frame, ErrpliP) & not(check_frame(Frame, ErrpliP)) & Errl >= MaxErrl	Errl = 1 Vseq = 1 Ack_expected= FALSE stop_timer(TL) DL_ABORT.ind(EL-E4F) Phy_ABORT.req()	Stopped
M.RecCr	time_out(TL) & Errl < MaxErrl	Errpli = Errpli + EL_E1 Phy_DATA.req(Fr) Errl=Errl +1	M.RecPE
M.RecCr	time_out(TL) & Errl >= MaxErrl	Errl = 1 Vseq = 1 Ack_expected= FALSE DL_ABORT.ind(EL-E4F) Phy_ABORT.req()	Stopped
M.RecCr	DL_ABORT.req()	Errl = 1 Vseq = 1 Ack_expected= FALSE stop_timer(TL) Phy_ABORT.req()	Stopped
M.RecCr	Phy_ABORT.ind(ErrorNb)	Errl = 1 Vseq = 1 Ack_expected= FALSE stop_timer(TL) DL_ABORT.ind(ErrorNb)	Stopped
Idle	DL_DATA.req(DSDU) & not(TestS)	Type="0000"B Text=DSDU Size=size(Text) + 4 Fr=concat(Size, Type, VSeq, Text,_) Phy_DATA.req(concat_Fr(Fr, crc(Fr))) Ack_expected=TRUE Errl = Errl + 1	M.RecPE
Idle	DL_DATA.req(DSDU) &TestS	Type="0000"B Text=DSDU Size=size(Text) + 5 Fr=concat(Size, Type, VSeq, Text, Errpli) Phy_DATA.req(concat_Fr(Fr, crc(Fr))) Ack_expected=TRUE Errl = Errl + 1	M.RecPE
Idle	Phy_DATA.ind(Frame, ErrpliP) & not(check_frame(Frame, ErrpliP))	Errpli = Errpli + ErrpliP + EL_R1 Type="1011"B	M.SendCr
Idle	Phy_DATA.ind(Frame, ErrpliP) & check_frame(Frame, ErrpliP) & is_text(Frame) & (Nseq = Vseq Nseq= Vseq - 1)	extract_test(Frame, TestM, CodeTest)	T.NbD
Idle	Phy_DATA.ind(Frame, ErrpliP) & check_frame(Frame, ErrpliP) & (not(is_text(Frame)) (is_text(Frame) & (Nseq<>Vseq) & (Nseq<> Vseq - 1)))	Errl = 1 Vseq = 1 Ack_expected= FALSE extract_test(Frame, TestM, CodeTest) DL_ABORT.ind(EL- R3F) Phy_ABORT.req()	Stopped

Idle	DL_ABORT.req()	Errl = 1 Vseq = 1 Ack_expected= FALSE Phy_ABORT.req()	Stopped
Idle	Phy_ABORT.ind(ErrorNb)	Errl = 1 Vseq = 1 Ack_expected= FALSE DL_ABORT.ind(ErrorNb)	Stopped
M.SendCr	not(TestS)	Size= 4 Fr=concat(Size, Type, VSeq, _,_) Phy_DATA.req(concat_Fr(Fr, crc(Fr)))	M.RecPE
M.SendCr	TestS	Size= 5 Fr=concat(Size, Type, VSeq, _, Errpli) Phy_DATA.req(concat_Fr(Fr, crc(Fr)))	M.RecPE
T.NbCr	is_ack(Frame) & (Nseq = Vseq)	Errl = 1 Vseq = Vseq + 1 DL_EndDATA.ind() Ack_expected= FALSE	Idle
T.NbCr	is_ack(Frame) & Nseq <> Vseq	Errl = 1 Vseq = 1 Ack_expected= FALSE DL_ABORT.ind(EL-E5F) Phy_ABORT.req()	Stopped
T.NbCr	is_text(Frame) & (Nseq = Vseq - 1) & Errl < MaxErrl	Phy_DATA.req(Fr) Errl=Errl +1 Errpli = Errpli + EL_E3	M.RecPE
T.NbCr	is_text(Frame) & (Nseq = Vseq - 1) & Errl = MaxErrl	Errl = 1 Vseq = 1 Ack_expected= FALSE DL_ABORT.ind(EL-E4F) Phy_ABORT.req()	Stopped
T.NbCr	is_text(Frame) & (Nseq = Vseq + 1)	Errpli = Errpli + EL_E2 Errl = 1 Vseq = Vseq + 1 DL_EndDATA.ind() Ack_expected= FALSE	Idle
T.NbCr	is_text(Frame) & (Nseq <> Vseq + 1) & (Nseq <> Vseq - 1)	Errl = 1 Vseq = 1 Ack_expected= FALSE DL_ABORT.ind(EL- E5F) Phy_ABORT.req()	Stopped
T.NbD	Nseq = Vseq	\$none()	M.SendCr
T.NbD	Nseq = Vseq - 1	Errpli = Errpli + EL_R2 Vseq = Vseq - 1	M.SendCr

Signification des états mentionnés dans le tableau précédent

Etat	Signification
<i>Init</i>	état d'initialisation
Stopped	état de démarrage commun à l'Appelant et à l'Appelé
M.RecPE (Must Receive Phy_Enddata.ind)	état de l'Emetteur : attente de l'événement généré par le protocole physique signalant la fin de l'envoi de la dernière trame
M.RecCr (Must Receive frame ContRol)	état du Récepteur : sous le contrôle du réveil TL, attente d'une trame de contrôle en provenance de l'Emetteur
Idle	état opérationnel d'attente de réception d'une trame ou d'une demande d'envoi de données
<i>M.SendCr</i> (Must Send frame ContRol)	état de l'Emetteur : envoi d'une trame de contrôle
<i>T.NbCr</i> (Test sequence NumBer of a frame ContRol)	vérification du numéro de séquence d'une trame de contrôle
<i>T.NbD</i> (Test sequence NumBer of a Data frame)	vérification du numéro de séquence d'une trame de données

Définition des procédures et des fonctions classées dans l'ordre alphabétique

Procédure ou fonction	Définition
check_frame(Frame, ErrpliP)	vérification que la trame Frame reçue est correcte : <ul style="list-style-type: none"> ■ CRC correct, ■ type de trame correct, ■ nombre d'octets compatible avec le champ Size, ■ ErrpliP = 0.
check_EL2(Errpli)	vérification que le bit d'erreur EL_R2 a été mis dans Errpli
Fr = concat(Size, Type, Vseq, Text, Errpli)	construction d'une chaîne binaire Fr par concaténation de plusieurs paramètres
concat(Fr, crc(Fr))	construction d'une chaîne binaire par concaténation du premier paramètre avec le deuxième paramètre ;
crc(OctetString)	calcul du CRC de la chaîne d'octets OctetString
exist_dl_data_req(DL_DATA.req(DSDU))	consommation d'un événement de type DL_DATA.req(DSDU)
extract_test(Frame, TestM, CodeTest)	extraction du champ Test de la trame Frame et recopie dans CodeTest, si TestM est à VRAI
extract_text(Frame)	extraction du champ Text de la trame Frame
init_timer(TL)	armement du réveil TL
Is_ack(Frame)	vérification que la trame Frame reçue est un acquittement positif
Is_nack(Frame)	vérification que la trame Frame reçue est un acquittement négatif
Is_text(Frame)	vérification que la trame Frame reçue est une trame de données
size(Text)	calcul du nombre d'octets du champ Text
Stop_timer(TL)	désarmement du réveil TL
Time_out(TL)	déclenchement du réveil TL

Précisions concernant la variable NSEQ :

Pour des raisons historiques, la description de l'automate fait apparaître une variable Nseq qui n'est jamais affectée. Cette variable Nseq fait référence au champ NSEQ de la trame reçue et qui est en cours d'analyse. Il est possible d'utiliser une « vraie » variable, mise à jour (implicitement) par la fonction check_frame, ou une fonction « contextuelle » de la forme « extract_Nseq(Frame) ».

3.3.3. Couche Liaison en mode Echo

La station Esclave peut fonctionner en mode Echo ; dans ce cas, la couche Liaison en mode ECHO remplace à la fois la couche Liaison en mode normal et la couche Session. Cette couche se contente de ré-émettre toute trame reçue. La machine d'état n'existe que sur une station Esclave (le système appelé).

Transitions d'état de Liaison en mode ECHO

Etat initial	Cond. de déclenchement	Ensemble d'actions	Etat final
<u>Stopped</u>	Phy_DATA.ind(Frame, ErrpliP)	Fr = Frame Phy_DATA.req(Fr)	M.RPEE
M.RPEE	Phy_EndDATA.ind()	\$none ()	M.RecFE
M.RPEE	Phy_ABORT.ind(_)	\$none ()	<u>Stopped</u>
M.RecFE	Phy_DATA.ind(Frame, ErrpliP)	Fr = Frame Phy_DATA.req(Fr)	M.RPEE
M.RecFE	Phy_ABORT.ind(_)	\$none ()	<u>Stopped</u>

Signification des états mentionnés dans le tableau précédent

Etat	Signification
<u>Stopped</u>	état de démarrage
M.RPEE (Must Receive Phy_Enddata.ind and Echo)	état de l'émetteur : attente de l'événement généré par le protocole physique signalant la fin de l'envoi de la dernière trame
M.RecFE (Must RECeive Frame and Echo)	état du Récepteur : attente d'une trame en provenance de l'Emetteur

3.3.3.1. Répertoire et traitement des erreurs

Les erreurs sont répertoriées par le codage **EL-RNF** ou **EL-ENF** portant les significations suivantes :

EL	erreur de la couche Liaison
-R	erreur en réception
-E	erreur en émission
N	numéro de l'erreur
F	erreur fatale.

Tableau récapitulatif des erreurs

EL-R1	Trame incorrecte. Cette erreur peut avoir les origines suivantes : <ul style="list-style-type: none"> ■ erreur détectée au niveau physique, ■ erreur de CRC, ■ type de trame différent de "0000 ", "0110", "1011", ■ nombre d'octets non compatible avec le champ Size.
	Cette erreur conduit à renvoyer la trame de données en cours, ou un NACK (acquiescement négatif)
EL-R2	Réception d'une trame de données avec un numéro de séquence égal au numéro attendu moins 1 (trame de données déjà acquiescée positivement)
	Cette erreur conduit à décrémenter le numéro attendu de 1, puis à envoyer un acquiescement positif
EL-E1	Expiration du délai TL sans qu'aucune trame de contrôle n'ai été reçue
	Cette erreur conduit à renvoyer la trame de données en cours, l'expiration du délai TL étant interprété comme un acquiescement négatif
EL-E2	Réception d'une trame de données à la place d'une trame de contrôle, avec un numéro de séquence égal au numéro attendu plus 1
	Cette erreur conduit à incrémenter le numéro attendu de 1 ; la trame reçue est interprétée comme un acquiescement positif
EL-E3	Réception d'une trame de données à la place d'une trame de contrôle, avec un numéro de séquence égal au numéro attendu moins 1
	Cette erreur conduit à renvoyer la trame de données en cours, la trame reçue étant interprétée comme un acquiescement négatif
EL-R3F	Réception d'une trame de données avec un numéro de séquence différent du numéro attendu ou du numéro attendu moins 1 ; ou réception d'une trame de contrôle sans envoi d'une trame de données au préalable
	Cette erreur conduit à réinitialiser la couche Liaison après avoir informé la couche Session et fait avorter la couche Physique
EL-E4F	Une quantité égale à (MaxErrl moins 1) envois de la même trame effectuée sans qu'aucune trame d'acquiescement positif n'ait été reçue
	Cette erreur conduit à réinitialiser la couche Liaison après avoir informé la couche Session et fait avorter la couche Physique
EL-E5F	Réception d'une trame de contrôle avec un mauvais numéro de séquence ; ou réception à la place d'une trame de contrôle d'une trame de données, avec un numéro de séquence différent du numéro attendu moins 1 ou du numéro attendu plus un
	Cette erreur conduit à réinitialiser la couche Liaison après avoir informé la couche Session et fait avorter la couche Physique

Les erreurs non fatales sont signalées dans la variable ERRPLI. Toute occurrence de l'une des erreurs fatales est remontée localement grâce à la primitive de service DL_ABORT.ind. La liste complète des numéros d'erreur fatale est fournie au chapitre 3.5.

3.4. Couche Session

3.4.1. Généralités sur la Couche Session

La couche **Session** est responsable du dialogue entre les stations. Elle permet l'échange de données entre une station Maître et une station Esclave. La station Maître est toujours l'Appelant, la station Esclave est toujours l'Appelé.

La couche Session offre deux types de services :

- les services d'administration de la connexion : établissement, maintien et coupure entre les deux stations,
- les services d'échanges de données en lecture ou en écriture: composition du dialogue entre stations par échanges d'invitation à émettre, invitations à recevoir, segmentation des données à échanger.

Le protocole de la couche **Session** présente des spécificités selon que l'équipement fonctionne en Maître ou en Esclave.

Précisions sur la variable « Code »

Dans le présent chapitre, la variable « Code » est utilisée pour différents services ou primitives décrits au sous-chapitre 3.4.2 et pour des commandes SPDU décrites au sous-chapitre 3.4.3.

Le contenu de cette variable est le suivant :

- le 1^{er} octet du double-octet de la variable « Code » contient la valeur (codée en hexadécimal) du Code de télé-relevé du groupe de données à transférer ;
- le 2^{ème} octet du double-octet de la variable « Code » est variable en fonction du type de compteur et du groupe de données désiré. De manière générale, il peut contenir soit une référence identifiant l'application du compteur, soit un code complémentaire du code de télé-relevé précisant l'identité des données désirées.

Les valeurs exactes de la variable « Code » sont précisées au chapitre 2.2.

Précisions sur les variables « Slaveld » et « MasterId »

Dans le présent chapitre, les variables « Slaveld » et « MasterId » sont utilisées pour différents services ou primitives décrits au sous-chapitre 3.4.2 et pour des commandes SPDU décrites au sous-chapitre 3.4.3.

Le contenu de ces variables est le suivant :

- la variable « Slaveld » contient la valeur de la « clé client » de l'appareil concerné,
- la variable « MasterId » contient la valeur de la « clé maître » de l'appareil concerné.

Les valeurs exactes des variables « Slaveld » et « MasterId » sont précisées au chapitre 2.1.1.

3.4.2. Services et primitives de service de Session

L'utilisateur du protocole **Session** dispose des services et primitives de service suivants :

Service	Primitive de service
S_Connect	S_Connect.req(MasterId, Slaveld) S_Connect.cnf()
S_Read	S_Read.req(Code, Lg, SSDU) S_Read.cnf(SSDU) S_Read.ind(Code) S_Read.rsp(SSDU) S_EndRead.ind()
S_Write	S_Write.req(Code,SSDU) S_Write.cnf() S_Write.ind(Code) S_Write.rsp(Lg, SSDU) S_EndWrite.ind(SSDU)
S_Disconnect	S_Disconnect.req() S_Disconnect.cnf()
S_ABORT	S_ABORT.ind ()

Les contenus des variables « Code », « SlaveId » et « MasterId » sont décrits en début du chapitre 3.4.
Le rôle attribué à chaque primitive est décrit dans les 2 tableaux ci-après selon qu'elles existent du côté Maître ou du côté Esclave, les requêtes s'adressant dans tous les cas à la couche Session.

Tableau des rôles de primitive existant du côté Maître

Primitive	Rôle de la primitive
S_Connect.req(MasterId, SlaveId)	Permet de demander l'ouverture d'une connexion entre un Maître d'identité MasterId et un esclave d'identité SlaveID
S_Connect.cnf()	Permet de confirmer l'ouverture d'une connexion préalablement demandée
S_Read.req(Code, Lg, SSDU)	Permet de demander une lecture de données caractérisées par un code et une longueur (SSDU permet de récupérer les données lues)
S_Read.cnf(SSDU)	Permet de transmettre les données qui ont fait l'objet d'une demande préalable de lecture
S_Write.req(Code, SSDU)	Permet de demander une écriture des données, caractérisée par un code
S_Write.cnf()	Permet de signaler la bonne fin d'une écriture préalablement demandée
S_Disconnect.req()	Permet de demander la fermeture d'une connexion
S_Disconnect.cnf()	Permet de confirmer la fermeture d'une connexion préalablement demandée
S_ABORT.ind()	Permet à la couche Session d'informer d'une fermeture brutale de la connexion à la suite d'une erreur

Les contenus des variables « Code », « SlaveId » et « MasterId » sont décrits en début du chapitre 3.4.

Tableau des rôles de primitive existant du côté Esclave :

Primitive	Rôle de la primitive
S_Read.ind(Code)	Permet de prévenir de l'arrivée d'une demande de lecture caractérisée par un code
S_Read.rsp(SSDU)	Permet de transmettre les données qui ont fait l'objet d'une demande préalable de lecture
S_EndRead.ind()	Permet de prévenir de la fin de la lecture en cours
S_Write.ind(Code)	Permet de prévenir de l'arrivée d'une demande d'écriture caractérisée par un code
S_Write.rsp (Lg, SSDU)	Permet de transmettre la longueur des données qui ont fait l'objet d'une demande préalable d'écriture, et un buffer SSDU pour recevoir ces données
S_EndWrite.ind(SSDU)	Permet de transmettre les données qui viennent d'être écrites, à la fin de l'écriture en cours
S_ABORT.ind()	permet à la couche Session d'informer d'une fermeture brutale de la connexion à la suite d'une erreur

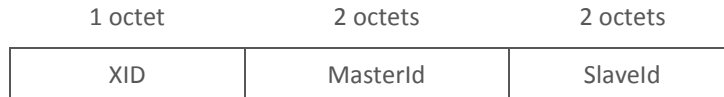
Les contenus des variables « Code », « SlaveId » et « MasterId » sont décrits en début du chapitre 3.4.

3.4.3. Description des SPDU

Les messages échangés entre entités de session sont acheminés dans des SPDU qui contiennent chacune un nombre entier d'octets inférieur ou égal à 122. Les SPDU possibles et leurs enchaînements sont décrits ci-après :

SPDU de type **XID** :

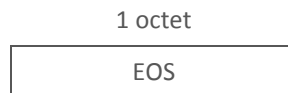
SPDUType (XID) : 0F Hexadécimal
 MasterId : 0 si la session est en lecture seule ou Identité du Maître sinon
 Slaveld : Identité de l'Esclave



Ce SPDU est envoyé à l'ouverture de connexion par le Maître ; l'Esclave répond par le même SPDU. Les contenus des variables « Slaveld » et « MasterId » sont décrits en début du chapitre 3.4.

SPDU de type **EOS** :

SPDUType (EOS) : 01 Hexadécimal



Ce SPDU est envoyé à la fermeture de connexion par le Maître ; l'Esclave ne répond pas.

SPDU de type **ENQ** :

SPDUType (ENQ) : 09 H
 Code : 2 octets



Ce SPDU est envoyé par le Maître pour initier une lecture caractérisée par le champ Code. Il sera suivi de l'envoi par l'Esclave des données demandées dans un ou plusieurs SPDU de type DAT. La connexion doit être ouverte. Le contenu de la variable « Code » est décrit en début du chapitre 3.4.

SPDU de type **REC** :

SPDUType (REC) : 06 H
 Code : 2 octets



Ce SPDU est envoyé par le Maître pour initier une écriture caractérisée par le champ Code. Il sera suivi de l'envoi par le Maître des données à écrire dans un ou plusieurs SPDU de type DAT. La connexion doit être ouverte. Le contenu de la variable « Code » est décrit en début du chapitre 3.4.

SPDU de type DAT :

SPDUType (DAT) : 0C H
Data : Données, 1 à 121 octets



Ce SPDU sert à transférer des données après une demande de lecture ou d'écriture ; il est envoyé par le Maître dans le cas d'une écriture, par l'Esclave dans le cas d'une lecture ; le nombre de SPDU de type DAT envoyés successivement dépend de la taille totale des données à écrire ou à lire.

SPDU de type EOD :

SPDUType (EOD) : 03 Hexadécimal



Ce SPDU est envoyé pour indiquer que toutes les données ont été transmises, à la fin d'une lecture ou d'une écriture ; il est envoyé par le Maître dans le cas d'une écriture, par l'Esclave dans le cas d'une lecture ; il suit donc toujours un SPDU de type DAT.

SPDU de type WTM :

SPDUType (WTM) : 0A H



Ce SPDU est envoyé par le Maître pour garder la connexion ouverte, alors qu'il n'a pas de commande à transmettre ; l'Esclave répond par un SPDU de même type.

3.4.4. Paramètres de la couche Session

La couche Session gère trois délais :

TSE correspondant à la durée maximale d'attente d'une SPDU de l'autre station, et imposé par le type d'échange en cours ; il est calculé sur la base de $6(TM+ TL)+TM$ max, avec une valeur de TM correspondant à une trame de 6 octets ; TSE est fixé à 22 s ;

TPA est le délai maximal d'attente d'une commande application du coté Maître ; son écoulement provoque le passage en mode attente si la session n'était pas déjà dans ce mode et l'envoi d'un SPDU de type WTM ; il est calculé sur la base de $TSE/3$; TPA est fixé à 6 s ;

TMA correspond à la durée maximale en mode attente ; il est fixé à 300 s.

La taille maximale des données véhiculées dans un SPDU de type DAT est égale à MaxPktSize ; elle est fixée à 121.



3.4.5. Transitions d'état

La machine d'état de l'Appelant diffère de celle de l'Appelé. Les deux systèmes jouent tour à tour le rôle d'Emetteur et de Récepteur de SSDU. Le Maître est toujours appelant, l'esclave toujours appelé. Les temps sont exprimés en millisecondes.

Transitions d'état de Session - Maître

Etat initial	Cond. de déclenchement	Ensemble d'actions	Etat final
<i>Init</i>	\$true()	MaxPktSize = 121 TPA = 6000 TSE = 22000 TMA = 300000	Stopped
Stopped	S_Connect.req(MasterId, SlaveId)	EndTMA = FALSE SlId = SlaveId MstId = MasterId Comm = XID store_error(_, Comm) DL_DATA.req(concat(XID, MstId, SlId))	M.RecDE
M.RecDE	DL_EndDATA.ind() & (Comm= XID Comm = ENQ Comm = WTM)	init_timer(TSE)	M.RData
M.RecDE	DL_EndDATA.ind() & Comm = EOD	S_Write.cnf () init_timer(TPA)	W.Comm
M.RecDE	DL_EndDATA.ind() & Comm = EOS & EndTMA	DL_ABORT.req() S_ABORT.ind()	Stopped
M.RecDE	DL_EndDATA.ind() & Comm = EOS & not(EndTMA)	stop_timer(TMA) DL_ABORT.req() S_Disconnect.cnf()	Stopped
M.RecDE	DL_EndDATA.ind() & Comm = DAT & LgSend > 0	store_error(_, Comm) substr_pack(SMsg, MaxPktSize, Packet) LgSend = LgSend - size(Packet) DL_DATA.req(concat(DAT, Packet))	M.RecDE
M.RecDE	DL_EndDATA.ind() & Comm = DAT & LgSend = 0	Comm = EOD store_error(_, Comm)	M.RecDE
M.RecDE	time_out(TMA)	EndTMA = TRUE	M.RecDE
M.RecDE	DL_ABORT.ind(ErrNb)	stop_timer(TMA) store_error(ErrNb, _) S_ABORT.ind()	Stopped
M.RData	DL_DATA.ind(SPDU) & check_type(extract_type(SPDU), Comm)	stop_timer(TSE) store_error(_, Type)	<i>T.Type</i>
M.RData	DL_DATA.ind(SPDU) & not(check_type(extract_type(SPDU), Comm))	store_error(ES.R2F, _) stop_timer(TMA) DL_ABORT.req () S_ABORT.ind()	Stopped

M.RData	time_out(TSE)	store_error(ES.R3F,_) stop_timer(TMA) DL_ABORT.req() S_ABORT.ind()	Stopped
M.RData	time_out(TMA)	EndTMA =TRUE	M.RData
M.RData	DL_ABORT.ind(ErrNb)	store_error(ErrNb,_) stop_timer(TSE) stop_timer(TMA) S_ABORT.ind()	Stopped
T.Type	Type = XID & check_id(SPDU)	S_Connect.cnf() init_timer(TPA)	W.Comm
T.Type	Type = XID & not(check_id(SPDU))	store_error(ES.R1F,_) stop_timer(TMA) DL_ABORT.req() S_ABORT.ind()	Stopped
T.Type	Type = DAT	Packet = extract_pkt(SPDU) LgPacket = size(Packet) Comm = DAT store_error(_Comm)	T.size
T.Type	Type = EOD	Comm = EOD store_error(_Comm)	T.size
T.Type	Type = WTM ¬(EndTMA)	init_timer(TPA)	W.Comm
T.Type	Type = WTM &EndTMA	Comm = EOS store_error(_Comm) DL_DATA.req(EOS)	M.RecDE
T.Size	Type = DAT & LgPacket <= LgReceive	Rmsg = concat(Rmsg, Packet) LgReceive = LgReceive - LgPacket	M.RData
T.Size	(Type = DAT & (LgPacket > LgReceive) (Type = EOD & LgReceive <> 0)	store_error(ES.R5F,_) stop_timer(TMA) DL_ABORT.req() S_ABORT.ind()	Stopped
T.Size	Type = EOD & LgReceive = 0	S_Read.conf(SSDU) init_timer(TPA)	W.Comm
W.Comm	time_out(TPA) & Type <> WTM	Comm =WTM store_error(_Comm) DL_DATA.req(WTM) init_timer(TMA)	M.RecDE
W.Comm	time_out(TPA) & Type = WTM	DL_DATA.req(WTM)	M.RecDE

W.Comm	S_Read.req(Code, Lg, SSDU)	stop_timer(TPA) EndTMA = FALSE stop_timer(TMA) Rmsg = SSDU LgReceive = Lg Comm = ENQ store_error(_Comm) DL_DATA.req(concat(ENQ, Code))	M.RecDE
W.Comm	S_Write.req(Code,SSDU) & MstId <> 0	stop_timer(TPA) EndTMA = FALSE stop_timer(TMA) Smsg =SSDU LgSend =size(SMsg) store_error(_REC) DL_DATA.req(concat(REC, Code)) Comm = DAT	M.RecDE
W.Comm	S_Write.req(Code,SSDU) & MstId = 0	S_Write.cnf(Err_access)	W.Comm
W.Comm	S_Disconnect.req()	stop_timer(TPA) EndTMA = FALSE stop_timer(TMA) Comm = EOS store_error(_Comm) DL_DATA.req(EOS)	M.RecDE
W.Comm	time_out(TMA)	EndTMA =TRUE	W.Comm
W.Comm	DL_Abort.ind (Errnb)	store_error(ErrNb,_) stop_timer(TPA) stop_timer(TMA) S_ABORT.ind()	Stopped

Les contenus des variables « Code », « SlaveId » et « MasterId » sont décrits en début du chapitre 3.4.

Transitions d'état de Session - Esclave

Etat initial	Cond. de déclenchement	Ensemble d'actions	Etat final
<i>Init</i>	\$true()	TSE = 220 MxPktSize = 121	Stopped
Stopped	DL_DATA.ind(SPDU) & check_type(extract_type(SPDU), Comm)	Comm = EOS store_error(_,Type)	<i>T.Type</i>
M.RecDE	DL_EndDATA.ind() & (Comm= XID I Comm = WTM)	init_timer(TSE)	M.RData
M.RecDE	DL_EndDATA.ind() & Comm = DAT & LgSend >0	substr_pack(SMsg, MaxPktSize, Packet) LgSend = LgSend -size(Packet) DL_DATA.req(concat(DAT, Packet))	M.RecDE
M.RecDE	DL_EndDATA.ind() & Comm = DAT & LgSend = 0	Comm = EOD store_error(_, Comm) DL_DATA.req(EOD)	M.RecDE
M.RecDE	DL_EndDATA.ind() & Comm = EOD	S_EndRead.ind() Init_timer(TSE)	M.RData
M.RecDE	DL_ABORT.ind(ErrNb)	store_error(ErrNb, _) S_ABORT.ind()	Stopped
M.RData	DL_DATA.ind(SPDU) & check_type(extract_type(SPDU), Comm)	stop_timer(TSE) Type = extract_type(SPDU)	<i>T.Type</i>
M.RData	DL_DATA.ind(SPDU) & not(check_type(extract_type (SPDU), Comm))	stop_timer(TSE) store_error(ES.R2F, _) S_ABORT.ind() DL_ABORT.req()	Stopped
M.RData	time_out(TSE)	store_error(ES.R3F, _) S_ABORT.ind() DL_ABORT.req()	Stopped
M.RData	DL_ABORT.ind(ErrNb)	stop_timer(TSE) store_error(ErrNb, _) S_ABORT.ind()	Stopped
<i>T.Type</i>	Type = XID	MstID = extract_Mst(SPDU) SIID = extract_SI(SPDU) Comm = XID store_error(_, Comm)	<i>T.id</i>
<i>T.Type</i>	Type = ENQ	Comm = ENQ store_error(_, Comm) Code = extract_code(SPDU) S_Read.ind(Code)	W.Rrsp

<i>T.Type</i>	Type = REC	Comm = REC store_error(, Comm) Code = extract_code(SPDU) S_Write.ind(Code)	W.Wrsp
<i>T.Type</i>	Type = DAT	Comm = DAT store_error(, Comm) Packet = extract_pkt(SPDU) LgPacket = size(Packet)	<i>T.size</i>
<i>T.Type</i>	Type = EOD	Comm = EOD store_error(, Comm)	<i>T.size</i>
<i>T.Type</i>	Type = WTM	Comm = WTM store_error(, Comm) DL_DATA.req(WTM)	M.RecDE
<i>T.Type</i>	Type = EOS	store_error(, EOS) S_ABORT.ind() DL_ABORT.req()	Stopped
<i>T.id</i>	check_id(SPDU)	store_error(, XID) DL_DATA.req(concat(XID, MstId, SId))	M.RecDE
<i>T.id</i>	not(check_id(SPDU))	store_error(ES.R1F, XID) S_ABORT.ind() DL_ABORT.req()	Stopped
<i>T.Size</i>	LgPacket <= LgReceive & Type = DAT	Rmsg = concat(Rmsg, Packet) LgReceive = LgReceive - LgPacket init_timer(TSE)	M.RData
<i>T.Size</i>	(Type = DAT & LgPacket > LgReceive) (LgReceive <> 0 & Type = EOD)	store_error(ES.R5F,_) S_ABORT.ind() DL_ABORT.req ()	Stopped
<i>T.Size</i>	LgReceive = 0 & Type = EOD	S_EndWrite.ind(Rmsg) init_timer(TSE)	M.RData
W.Rrsp	S_Read.rsp(SSDU)	Comm = DAT store_error(, Comm) Smsg =SSDU LgSend =size(SMsg) substr_pack(SMsg, MaxPktSize, Packet) Lgsend = Lgsend - size(Packet) DL_DATA.req(concat(DAT, Packet))	M.RecDE
W.Rrsp	DL_ABORT.ind(Errnb)	store_error(ErrNb,_) S_ABORT.ind()	Stopped
W.Wrsp	S_Write.rsp(Lg, SSDU) & MstId <> 0	LgReceive = Lg Rmsg = SSDU init_timer(TSE)	M.RData
W.Wrsp	S_Write.rsp(Lg, SSDU) & MstId = 0	store_error(ES.R4F,_) S_ABORT.ind() DL_ABORT.req()	Stopped
W.Wrsp	DL_ABORT.ind(Errnb)	store_error(ErrNb,_) S_ABORT.ind()	Stopped

Les contenus des variables « Code », « Slaveld » et « MasterId » sont décrits en début du chapitre 3.4.

Signification des états mentionnés dans les tableaux précédents

Etat	Signification
<i>Init</i>	état d'initialisation
Stopped	état de démarrage
M.RecDE (Must RECeive DI_Enddata.ind)	état d'attente de l'événement généré par le protocole liaison signalant l'envoi réussi des dernières données (données acquittées positivement)
M.RData (MustReceive dI_DATA.ind)	état d'attente de l'événement généré par le protocole liaison signalant la réception de données
W.Rrsp (Wait s_Read.RSP)	état de l'Esclave : attente du retour de l'application donnant les données à envoyer, lors d'une demande de lecture
W.Wrsp (Wait s_Write.RSP)	état de l'Esclave: attente du retour de l'application donnant la longueur des données qui seront reçues, lors d'une demande d'écriture
W.Comm (Wait Command)	état du Maître : attente d'une commande en provenance de l'application sous contrôle du délai TPA
<i>T.Id (Test Identity)</i>	état de l'Esclave : vérification des identités reçues dans un SPDU de type XID
<i>T.Type (Test Type)</i>	vérification du type du dernier SPDU reçu
<i>T.Size (Test Size)</i>	état de vérification de la taille des données reçues dans un SPDU de type DAT

Définition des procédures et des fonctions classées dans l'ordre alphabétique

Procédure ou fonction	Définition
check_id (SPDU)	vérification que les identifiants Maître et Esclave reçu dans un SPDU de type XID sont corrects
check_type(Type, Comm)	vérification que le type du SPDU reçu est correct, ainsi que son enchaînement : <ul style="list-style-type: none"> ■ pour le Maître, à vrai si : <ul style="list-style-type: none"> • type = XID et Comm = XID, • type = DAT et Comm = ENQ ou DAT, • type = EOD et Comm = DAT, • type = WTM et Comm = WTM, ■ pour l'Esclave, à vrai si : <ul style="list-style-type: none"> • type = XID et Comm = EOS, • type = ENQ ou REC et Comm = XID ou EOD ou WTM, • type = DAT et Comm = REC ou DAT, • type = EOD et Comm = DAT, • type = WTM et Comm = WTM ou XID ou EOD, • type = EOS et Comm = EOD ou WTM.
concat(Type, Packet) ou concat(Type, Code) concat(Type, MstId, SlId)	construction d'une chaîne binaire par concaténation de plusieurs paramètres
extract_code(SPDU)	extraction du champ code d'une SPDU de type REC ou ENC (Esclave)
extract_Mst(SPDU)	extraction du champ identité Maître d'une SPDU de type XID (Esclave)
extract_pkt(SPDU)	extraction du champ de données d'une SPDU de type DAT
extract_SI(SPDU)	extraction du champ identité Esclave d'une SPDU de type XID (Esclave)
extract_type(SPDU)	extraction du type de la SPDU reçue
init_timer(TSE), init_timer(TPA) ou init_timer(TMA)	armement du réveil TSE, TPA ou TMA
size(SMsg)	calcul de la taille en octets du message SMsg
stop_timer(TSE), stop_timer(TPA) ou stop_timer(TMA)	désarmement du réveil TSE, TPA ou TMA
store_error(Errnb, Comm)	recopie de la dernière commande Comm dans les 4 bits de poids fort de ERRSES et du numéro de la dernière erreur fatale dans ERRFAT ou ERRSES
substr_pack(SMsg, MaxPktSize, packet)	extraction de MaxPktSize octets du message SMsg et recopie dans Packet ; si Smsg contient moins d'octets que MaxPktSize, recopie de Smsg dans Packet
time_out(TSE), time_out(TPA) ou time_out(TMA)	déclenchement du réveil TSE, TPA ou TMA

3.4.6. Répertoire et traitement des erreurs de la couche *Session*

Les erreurs sont répertoriées par le codage **ES-RNF** ou **ES-ENF** portant les significations suivantes :

ES	erreur de la couche <i>Session</i>
-R	erreur en réception
-E	erreur en émission
N	numéro de l'erreur
F	erreur fatale

Tableau récapitulatif des erreurs

ES-R1F	Réception d'un mauvais identificateur Maître ou Esclave
	Cette erreur conduit à réinitialiser la couche <i>Session</i> , faire avorter la couche <i>Liaison</i> , et à en informer l'application si l'équipement est Maître
ES-R2F	Réception d'une SPDU non répertoriée
	Cette erreur conduit à réinitialiser la couche <i>Session</i> , faire avorter la couche <i>Liaison</i> , et à en informer l'application si l'équipement est Maître
ES-R3F	Ecoulement du délai TSE sans réception d'une SPDU
	Cette erreur conduit à réinitialiser la couche <i>Session</i> , faire avorter la couche <i>Liaison</i> , et à en informer l'application si l'équipement est Maître
ES-R4F	Réception d'une demande d'écriture dans une session lecture seule (Esclave)
	Cette erreur conduit à réinitialiser la couche <i>Session</i> et à avorter la couche <i>Liaison</i>
ES-R5F	Réception d'un nombre de données non prévu
	Cette erreur conduit à réinitialiser la couche <i>Session</i> après en avoir informé l'application et fait avorter la couche <i>Liaison</i>

Toute occurrence de l'une des erreurs fatales est signalée dans ERRFAT et ERRSES (cf. chapitre 3.5).

3.5. Liste des erreurs

Les erreurs sont consignées dans les variables ERRPLI (pour les couches Physique et Liaison), ERRFAT (pour les erreurs fatales toutes couches confondues) et ERRSES (erreurs de la couche SESSion).

Codage de ERRPLI

Numéro du bit	0	1	2	3	4	5	6	7
Erreur	EP-R1			EL-R1	EL-R2	EL-E1	EL-E2	EL-E3

Codage de ERRFAT

Numéro du bit	0	1	2	3	4	5	6	7
Erreur	EP-R4F	EP-R5F	EL-R3F	EL-E4F	EL-E5F	ES-R1F	ES-R2F	ES-R3F

Codage de ERRSES

Numéro du bit	0	1	2	3	4	5	6	7
Erreur	ES-R4F	ES-R5F	EP-R6F	EP-R7F	dernière commande correcte			

Les erreurs fatales sont mémorisées par la couche Session dans les variables ERRFAT et ERSES (Errses mémorise également la dernière commande correcte avant l'erreur fatale).

La couche Liaison mémorise dans la variable ERRPLI les dernières erreurs non fatales détectées par les couches Physique et Liaison ; ERRPLI est remis à 0 à chaque fois qu'une trame est envoyée sans erreur.

Quelle que soit la couche incriminée, l'occurrence d'une erreur fatale conduit à :

- éventuellement, faire s'arrêter la couche inférieure au moyen d'une primitive de service abort.req ;
- éventuellement, informer la couche supérieure au moyen d'une primitive abort.ind avec comme paramètre le numéro de l'erreur fatale ;
- effectuer une réinitialisation totale de l'occurrence d'automate correspondante.

3.6. Principe du CRC

3.6.1. Généralités sur le CRC

Le contrôle de la transmission correcte des bits se fait globalement sur un bloc d'information exprimé en octets (cf. le chapitre 3.3 décrivant la couche Liaison). La clef de contrôle est un ensemble de bits appelé **champ de contrôle**. Son calcul repose sur la théorie des codes cycliques qui fait appel à la **division des polynômes** et aux propriétés algébriques des **restes de la division** des polynômes.

3.6.2. Opérations sur les polynômes

Soit $A = (a_1, a_2, \dots, a_n)$ la suite de bits sur laquelle doit être calculée la clef de contrôle. Cette suite peut être vue comme un polynôme de degré $n-1$:

$$A(X) = a_1X^{n-1} + \dots + a_{n-1}X + a_n$$

Supposons choisi un polynôme diviseur $D(X)$ de degré m . La division polynomiale de $A(X)*X^m$ par $D(X)$ donne la relation suivante :

$$A(X)*X^m = D(X)*Q(X) + R(X)$$

où $R(X)$ est un polynôme de degré inférieur ou égal à $m-1$ représentant la suite de bits $R = (r_1, r_2, \dots, r_m)$

Compte tenu des propriétés de l'arithmétique booléenne, cette relation s'écrit aussi :

$$A(X)*X^m + R(X) = A(X)*X^m - R(X) = D(X)*Q(X)$$

ce qui représente la suite de bits $(a_1, a_2, \dots, a_n, r_1, r_2, \dots, r_m)$

3.6.3. Procédure de contrôle

Le champ de contrôle du CRC (Cyclical Redundancy Check) est représenté par la suite de bits $R = (r_1, r_2, \dots, r_m)$. Son calcul pratique courant est à base de registres de décalage et de cumul permettant d'évaluer la clef au fur et à mesure que les bits de données se présentent. L'algorithme correspondant n'est pas décrit dans le présent document.

Comme l'indique la théorie, l'Emetteur doit évaluer la suite de bits $R = (r_1, r_2, \dots, r_m)$, concaténer cette suite à la séquence des bits à protéger $A = (a_1, a_2, \dots, a_n)$ et, transmettre au Récepteur la suite de bits résultante $(a_1, a_2, \dots, a_n, r_1, r_2, \dots, r_m)$.

Le Récepteur considère qu'il n'y a pas d'erreur de transmission dès lors que la suite de bits reçues correspond à un polynôme de degré $m+n-1$ divisible exactement par $D(X)$.

3.6.4. Paramètres de fonctionnement

m	16
D(X)	$X^{16} + X^{15} + X^2 + 1$