

AGRÉGATION

SESSION 2006

CONCOURS INTERNE

Section : GÉNIE ÉLECTRIQUE

Option A : ÉLECTRONIQUE ET INFORMATIQUE INDUSTRIELLE

ÉTUDE D'UN SYSTÈME INDUSTRIEL

DURÉE : 8 HEURES, COEFFICIENT : 1

Aucun document n'est autorisé.

Calculatrice autorisée (conformément à la circulaire n° 99-186 du 16 novembre 1999)
Convertisseur en Euro autorisé

ROBOT MOBILE D'INTERVENTION ET DE SÉCURITÉ

Ce sujet comporte 4 dossiers distincts :

📁 Dossier de présentation	16 pages
📁 Texte du sujet avec le travail demandé	19 pages
📁 Documents réponses DR1 à DR7	3 pages
📁 Documents annexes	19 pages
📁 Schémas structurels au format A3	2 pages

Ce sujet comporte trois parties indépendantes :

Partie **A** : Modélisation et analyse du fonctionnement.

Partie **B** : Analyse de la transmission des données.

Partie **C** : Analyse et implémentation de l'action locomotion.

Une lecture préalable et complète du sujet est indispensable.

Il sera tenu compte de la cohérence avec laquelle les candidats traiteront chaque partie, le jury préférant une réflexion d'ensemble de la partie abordée à un éparpillement des réponses.

! Les questionnements portant sur les développements pédagogiques se verront affectés de 60 % des points de l'épreuve. Il est donc conseillé d'y consacrer 60 % du temps de l'épreuve.

Les candidats sont invités à numéroter chaque page de leur copie et à indiquer clairement le numéro de la question traitée.

Les candidats sont priés de rédiger les différentes parties du problème sur feuilles séparées et clairement repérées. Chaque question est identifiée par une police italique et repérées par un numéro.

Il leur est rappelé qu'ils doivent utiliser les notations propres au sujet, présenter clairement les calculs et dégager ou encadrer tous les résultats.

Tout résultat incorrectement exprimé ne sera pas pris en compte. En outre les correcteurs leur sauront gré d'écrire lisiblement et de soigner la qualité de leur copie.

Il sera tenu compte de la qualité de rédaction, en particulier pour les réponses aux questions ne nécessitant pas de calcul. Le correcteur attend des phrases complètes respectant la syntaxe de la langue française.

Pour la présentation des applications numériques, il est rappelé que lors du passage d'une forme littérale à son application numérique, il est recommandé aux candidats de procéder comme suit :

- après avoir rappelé la relation littérale, chaque grandeur est remplacée par sa valeur numérique en respectant la position qu'elle avait dans la relation puis le résultat numérique est donné sans calculs intermédiaires et sans omettre son unité.

Si le texte du sujet, de ses questions ou de ses annexes, vous conduit à formuler une ou plusieurs hypothèses, il vous est demandé de la (ou les) mentionner explicitement dans votre copie.

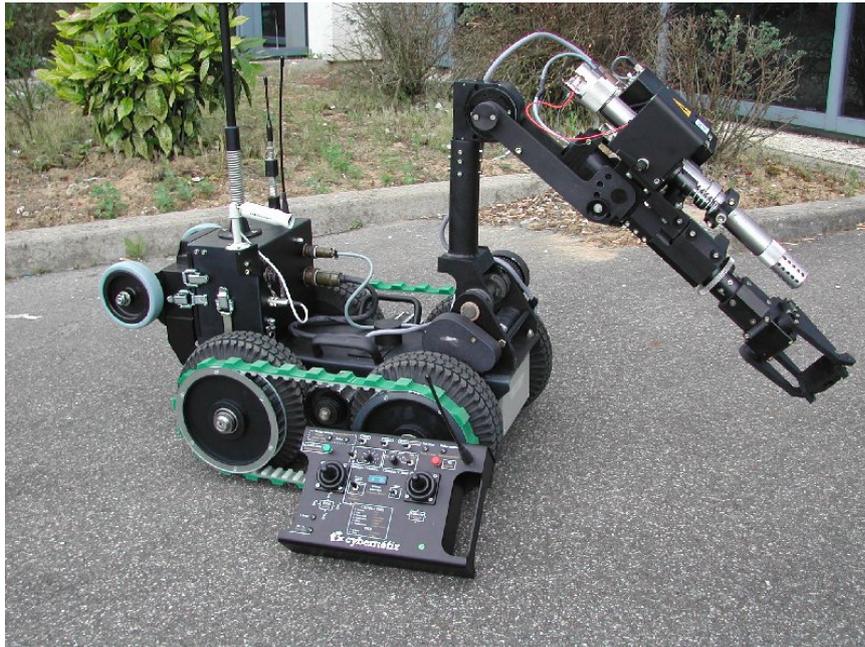
1 PRÉSENTATION DU SYSTÈME

1.1 PRÉSENTATION GÉNÉRALE

L'unité mobile de télé opération que nous nous proposons d'étudier ici est développée et commercialisée par la société CYBERNETIX (robot d'intervention télé opéré CASTOR), spécialisée dans les domaines de la robotique mobile, civile et militaire.

Ce robot est conçu pour effectuer toutes sortes d'opérations d'inspection et de surveillance en milieu hostile et dangereux pour l'homme : milieux chimique ou nucléaire, champs de mines, lieux publics soumis à des risques d'attentats, etc.

Son faible encombrement et sa légèreté le rendent particulièrement intéressant pour des missions urgentes et des approches rapides d'objets suspects.



Le robot peut-être équipé de différentes fonctionnalités dont la modularité permet d'optimiser l'efficacité du système au regard de la nature de la mission à effectuer (modules missions).

L'étude portera sur l'architecture générale du robot et sur certaines fonctions, tant du point de vue matériel que logiciel, l'approche se vaudra donc pluridisciplinaire (électronique, informatique industrielle, automatique).

1.2 PRÉSENTATION DU MATÉRIEL

Le système est composé de deux unités distinctes :

Une valise incluant un pupitre permettant de piloter l'ensemble des actionneurs du Robot, de visualiser les états du système ainsi que les images acquises par les caméras vidéo.



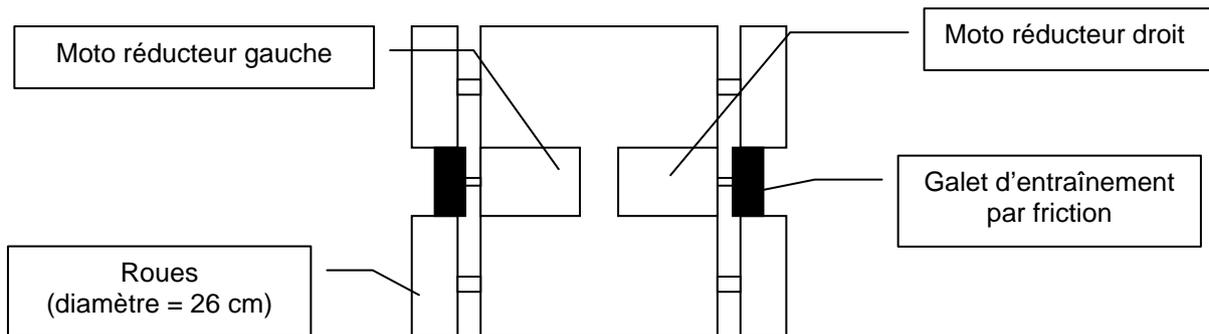
Une unité mobile équipée de modules pour des missions adéquats



L'unité mobile (robot) communique avec l'unité de télécommande (valise) par voie hertzienne, et dispose donc d'un émetteur vidéo, avec un sélecteur de canal pour la transmission des images caméras et d'une liaison MODEM radiofréquence half duplex pour la transmission des données numériques.

L'unité mobile (robot) est constituée de :

- Un bras articulé permettant de supporter les différents modules missions :
 - ✓ Un canon pour la neutralisation des objets piégés,
 - ✓ Une pince articulée pour la manipulation d'objets,
 - ✓ Un projecteur,
 - ✓ Un télémètre ultrasonore avec un pointeur laser pour la visée,
 - ✓ Une caméra,
 - ✓ Etc.
- Un système de caméra qui permet d'accueillir jusqu'à 4 caméras dont une montée sur tourelle motorisée.
- Un châssis mobile monté sur chenilles ou sur roues, propulsé par 2 moto réducteurs mus par des moteurs à courant continu, positionnés de chaque côté du robot.
Chaque roue est entraînée par friction d'un galet solidaire de l'axe du moto réducteur.



Ainsi, le sens de déplacement du Robot et sa direction ne dépendent que de la vitesse respective de chacun des moteurs. Une différence entre les vitesses de rotation de chaque moteur entraîne un virage de l'unité mobile.

La valise de commande contient :

- Un rack portable incluant toutes les commandes du robot,
- Un écran de contrôle à affichage LCD.

Notes sur les systèmes de transmission de données :

Il existe deux versions du système de communication des données par voies radiofréquences (MODEM), l'une exploitant les fréquences militaires et l'autre utilisant les bandes civiles. Pour des raisons de confidentialité, nous n'étudierons que la version exploitant la bande des **433 MHz**.

Dans tous les cas de figure, la transmission des données numériques par voie hertziennne s'effectue en utilisant le mode **NRZ-FSK**.

1.3 DÉLIMITATION DE LA PRÉSENTATION

Pour aborder un tel système il est important de délimiter les domaines que nous allons traiter dans cette étude.

Nous allons dans un premier temps limiter notre analyse au logiciel qui est embarqué dans l'unité robot.

Le rôle dévolu à cet élément est de dialoguer d'une part avec la valise et d'autre part avec chacune des unités spécialisées de commande des organes du robot.

Dans un second temps sera esquissée l'étude du logiciel embarqué dans la valise.

Pour terminer, nous vous proposons de vous intéresser plus en détail aux unités spécialisées.

Pour les besoins du sujet, la modélisation n'est que partielle.

2 MODÉLISATION

Nous proposons deux modélisations, l'une basée sur le modèle UML, l'autre s'appuyant sur une approche fonctionnelle.

Le modèle UML permet de décrire le comportement du robot vis-à-vis de son unité de commande et de l'utilisateur. Cette approche volontairement orientée logiciel est supportée par une architecture matérielle modulaire et extensible. Cette modularité est rendue possible grâce à l'utilisation d'un bus de terrain mono maître interne qui autorise l'ajout de modules repérés par une adresse unique.

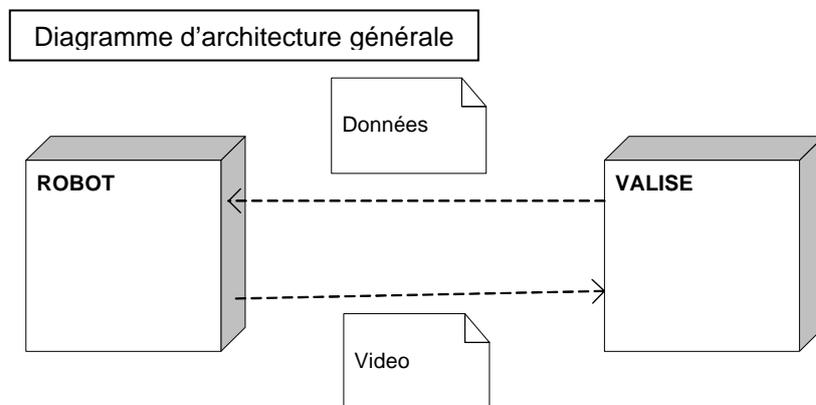
L'approche fonctionnelle a été réduite aux schémas de niveau 1 pour un robot et une valise de commande dans une configuration « standard ». Seuls les schémas électriques des modules étudiés (module de pilotage moteur et module de gestion des échanges côté robot) sont fournis.

2.1 MODÈLE UML

L'objectif de la démarche consiste à étudier les architectures logicielles respectives des deux principaux composants que sont le robot et la valise. Ces deux composants communiquent par deux liaisons hertziennes, la première étant dédiée à la vidéo (4 canaux simultanés, simplex) et la seconde aux données (télécommandes (TC) et télémessures (TM), half-duplex). Nous passerons sous silence la liaison filaire développée par le fabricant.

Nota: Le circuit vidéo étant fonctionnellement indépendant et ne nécessitant - hormis la commutation de circuit - aucune ressource logicielle sur les systèmes robot et valise, il sera mentionné mais ne sera pas étudié dans la suite du sujet.

Afin d'offrir un niveau de sécurité suffisant dans la transmission des données par voie hertzienne, celles-ci sont encapsulées dans des trames de réseau de terrain "Modbus". Le maître est embarqué dans la valise et l'esclave dans le robot.



2.1.1 Description des données échangées entre la valise et le robot :

La valise émet une trame (TC) d'écriture des données vers le robot toutes les 100 ms et ce dernier l'acquitte. Conformément aux spécifications Modbus, la valise interroge le robot qui émet en réponse une trame d'état (TM). Nous allons détailler dans cette partie les trames qui sont échangées entre la valise et le robot.

a) Trame émise de la valise vers le robot (Tram-TCvr)

Cette trame est composée de 6 octets (3 mots Modbus) :

- 4 octets pour les entrées analogiques (côté valise),
- 2 octets pour les entrées TOR (côté valise),

En-tête	Octet 1	Octet 2	Octet 3	Octet 3	Octet 4	Octet 5	Octet 6	CRC
---------	---------	---------	---------	---------	---------	---------	---------	-----

ENTREES ANALOGIQUES

ENTREES TOUT OU RIEN

Octet N°1 : TCvr_EANA4 (consigne avant bras) contient la valeur analogique sur 8 bits du joystick
avant bras : montée /descente

Octet N°2 : TCvr_EANA3 (consigne bras) contient la valeur analogique sur 8 bits du joystick
bras : montée/ descente

Octet N°3 : TCvr_EANA1 (consigne loco av/ar) contient la valeur analogique sur 8 bits du joystick
locomotion : sens avant/arrière

Octet N°4 : TCvr_EANA2 (consigne loco Dr/Ga) contient la valeur analogique sur 8 bits du joystick
locomotion : sens droite /gauche

Octet N°5 : TCvr_TOR 1

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Freins	Bit0 P/T/Z (activ.)	Bit1 P/T/Z (sens)	Bit2 P/T/Z (select1)	Bit3 P/T/Z (select2)	Bit4 P/T/Z (select 3)	Bit0 (LSB) Sel_cam	Bit1 (MSB) Sel_cam

Bits 1 et 0 :

Ce codage est fait à partir des 4 informations binaires : PA0, PA1, PA2, PA3 (sélection caméra) valise

0 0 = caméra n°1 (avant).

1 0 = caméra n°2 (droite).

0 1 = caméra n°3 (arrière).

1 1 = caméra n°4 (gauche).

Bits 4, 3, 2 : sélection moteur

x x x = moteurs Pan / Tilt / Zoom1 / Zoom2 / pince / orientation pince.

Bits 6, 5 : activation / Sens

Bits 7

0 = Freins serrés.

1 = Freins desserrés.

Gestion des freins : l'acteur locomotion pilote automatiquement les freins si une consigne de déplacement est envoyée (sans tenir compte de ce bit de forçage).

Octet N°6 : TCvr_TOR 2

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Libre	Laser	Enrouleur	Activ. RX	Projecteur	Validation Tir	TIR	Commut. Câble/RF

Bit 0 :

0 = Mode RF.

1 = Mode câble.

L'état de ce bit est la recopie de PC0 de la valise

Bit 1 :

0 = Pas de tir.

1 = Tir demandé.

L'état de ce bit est la recopie de PA7 de la valise

Bit 2 :

0 = pas de validation de tir.

1 = validation du tir.

L'état de ce bit est la recopie de PA6 de la valise

Bit 3 :

0 = arrêt projecteurs.

1 = allumage projecteurs.

L'état de ce bit est la recopie de PA4 de la valise

Bit 4 :

- 0 = arrêt activation RX.
- 1 = activation RX.

L'état de ce bit est la recopie de PC1 de la valise

Bit 5 :

- 0 = arrêt enrouleur.
- 1 = activation enrouleur.

L'état de ce bit est la recopie de PA5 de la valise

Bit 6 :

- 0 = arrêt laser.
- 1 = activation laser.

L'état de ce bit est la recopie de PC2 de la valise

b) Trame émise du robot vers la valise (Tram-TMrv)

En-tête	Octet 1	Octet 2	Octet 3	Octet 3	Octet 4	Octet 5	CRC
---------	---------	---------	---------	---------	---------	---------	-----

ENTREES ANALOGIQUES

ENTREES TOR

Cette trame est composée de 5 octets :
 4 octets pour les entrées analogiques (côté robot),
 1 octets TOR (entrées côté robot),

- Octet N°1** : TMrv_EANA (8bits LSB) EANA1 sur robot
distance du télémètre (8 bits poids faible des 12 utilisés)
- Octet N°2** : TMrv_EANA (4 bits MSB) EANA1 sur robot
Bit de 0 à 3 = distance télémètre laser (4 bits poids fort des 12 utilisés)
Bit de 4 à 7 = 0
- Octet N°3** : TMrv_EANA (8 bits MSB) EANA2 sur robot
- Octet N°4** : TMrv_EANA (8 bits MSB) EANA3 sur robot
- Octet N°5** : TMrv_TOR1 : Entrées TOR du robot

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Défaut batterie robot	ETOR1	Canon prêt à tirer	0	0	0	0	0

Bit 0 à 4 :

= 0

Bit 5 :

- = 0 si le bit2 de l'octet TOR2 de la trame reçu par le robot est à 0 ou à 1 pendant un temps inférieur à 3 s.
- = 1 si le bit2 de l'octet TOR2 de la trame reçu par le robot est à 1 pendant un temps supérieur à 3 s.

Bit 6 :

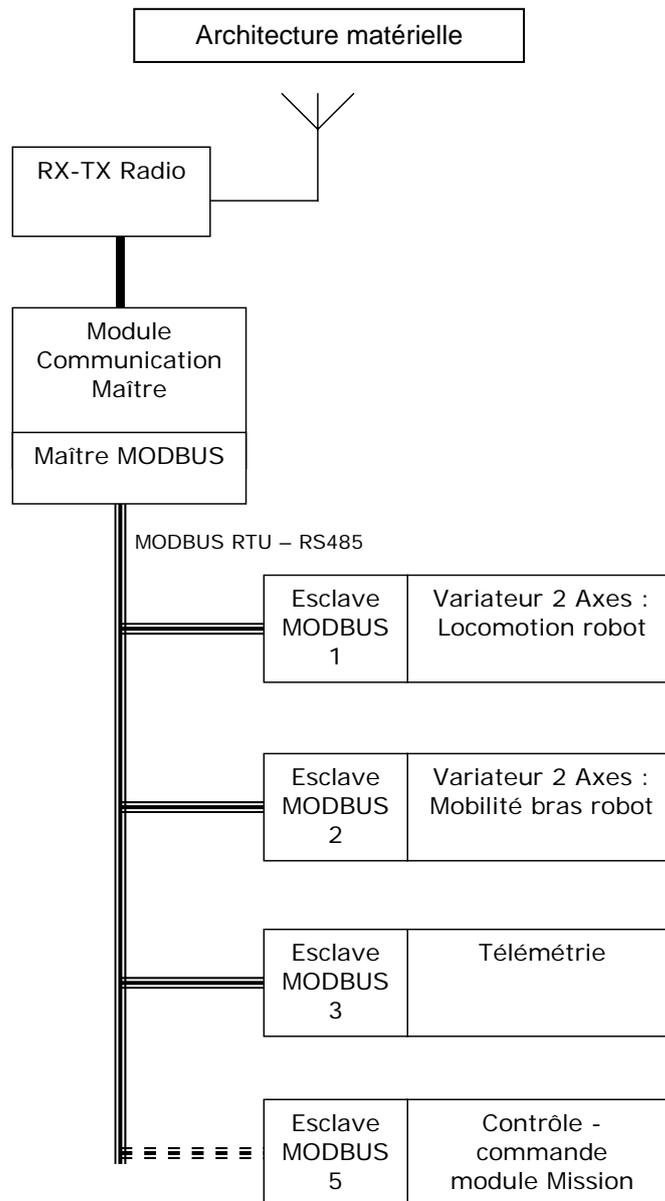
= état du bit PC0 du robot

Bit 7 :

= mise à 1 de ce bit si le seuil de l'entrée ANA0 (niveau batterie) est inférieur à 0AA9H pendant un temps supérieur à 3 s.

2.1.2 Architecture du robot

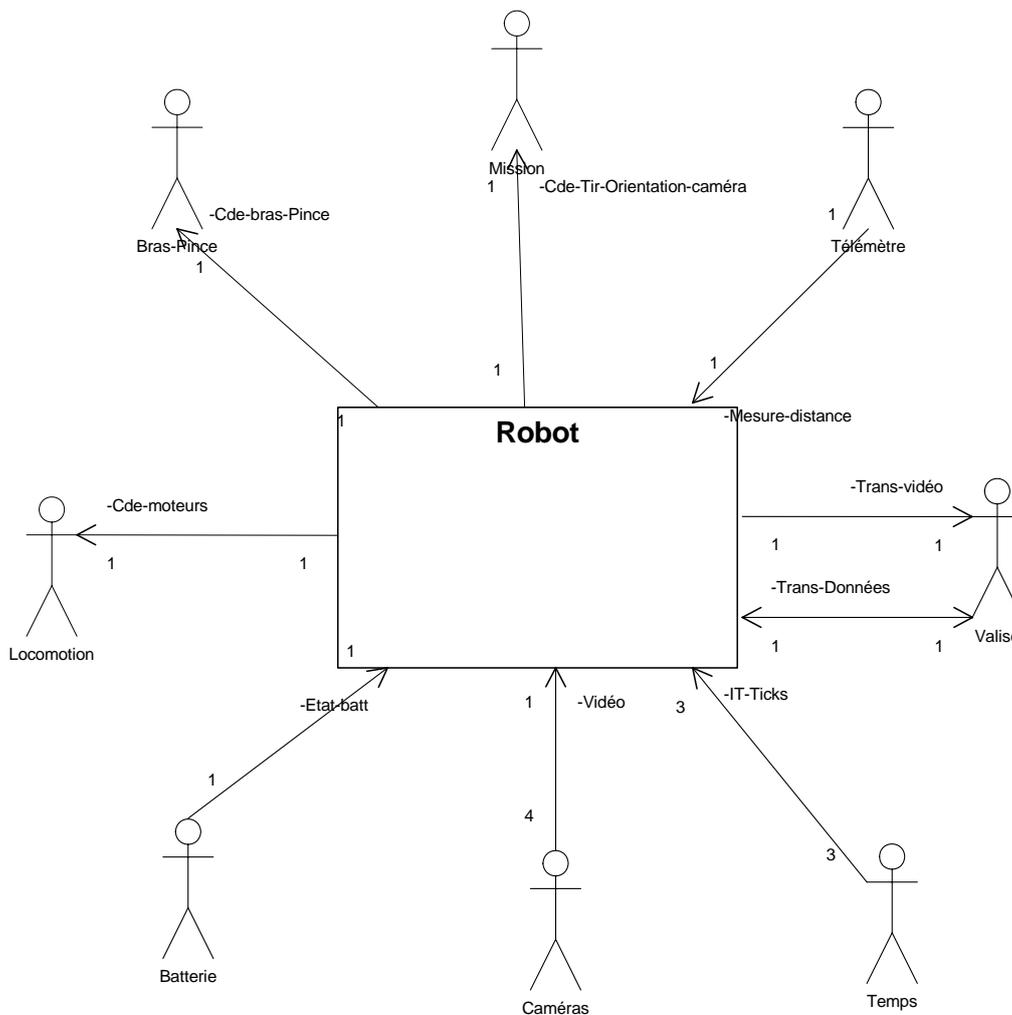
L'architecture matérielle de la partie commande du robot est composée d'une série d'unités de traitement organisées autour d'un BUS de communication MODBUS RTU.



La fiabilité du MODBUS RTU, de par son caractère déterministe, sa robustesse, son câblage très simple (3 fils) permet au module maître d'interroger ou de transmettre des ordres vers l'ensemble des modules. Pour ce faire, on utilise une table d'échange propre à chaque module. Un extrait est fourni en annexe E.

2.1.3 Modélisation de la partie supervision du robot

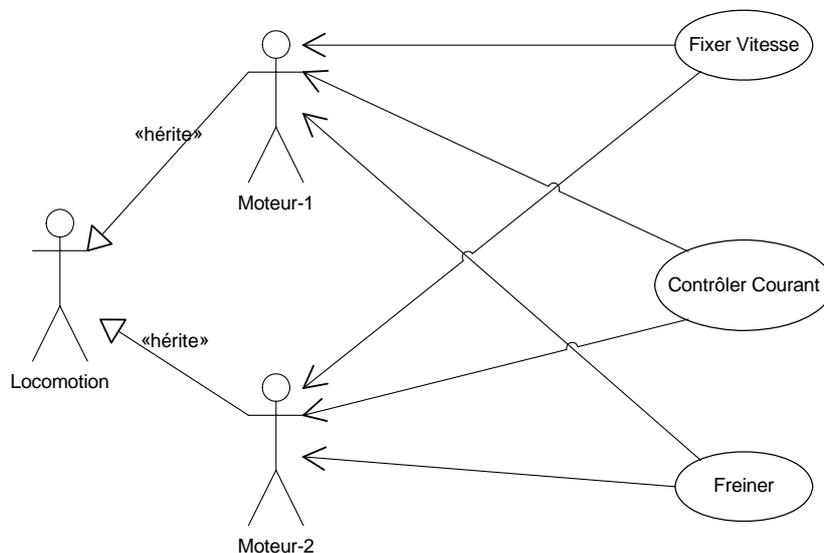
Repérons dans un premier temps les interactions entre la classe générique du système et son environnement. Chaque élément identifié de l'environnement est un acteur. Le diagramme ci-après est celui de contexte du système.



Dans le cadre du pilotage du robot nous sommes amenés, pour des raisons de lisibilité, à modéliser séparément l'interaction entre chaque acteur et le système sous forme de diagrammes de cas d'utilisation.

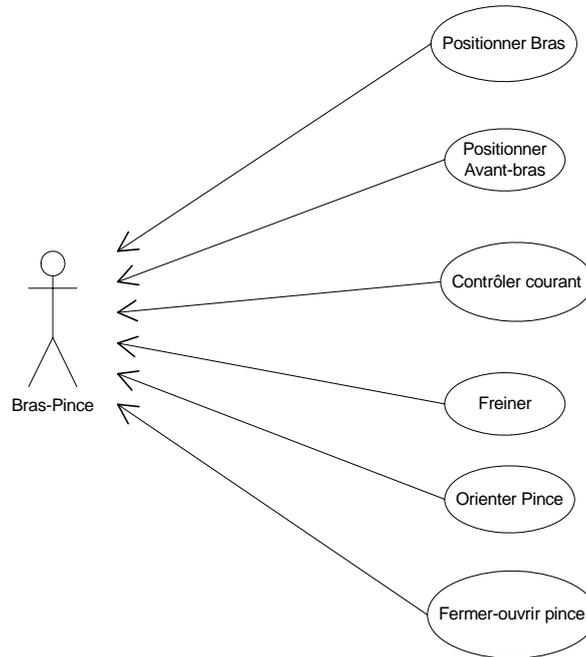
2.1.4 L'acteur locomotion

L'acteur locomotion assure le pilotage simultané de deux moteurs indépendants auxquels peut être envoyé une consigne de vitesse, elle fait intervenir une unité de traitement variateur deux axes.



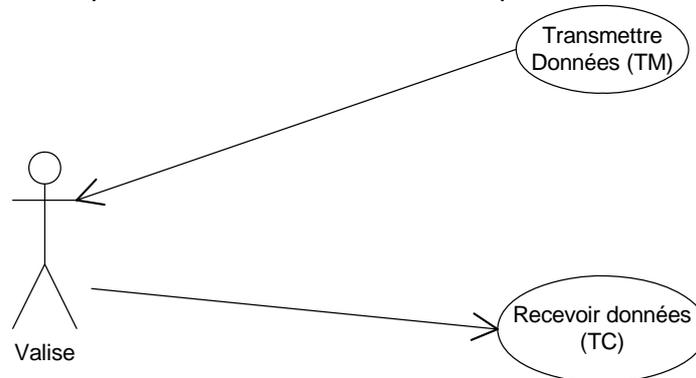
2.1.5 L'acteur bras/pince

Le bras, la pince et le tir font l'objet de pilotages indépendants, totalement contrôlés par l'opérateur, faisant intervenir deux unités de traitement variateur deux axes (Bras-Avant bras + Rotation-fermeture Pince).



2.1.6 L'acteur valise

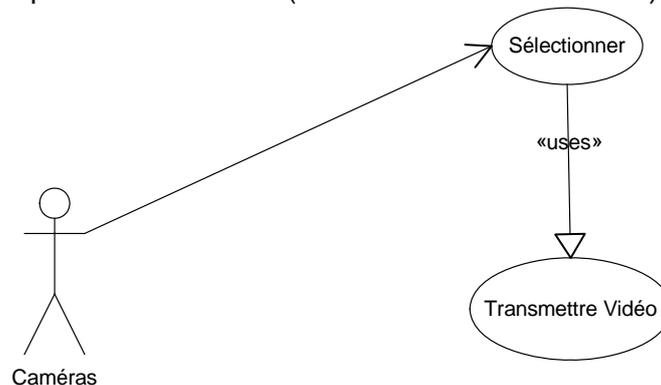
La valise interagit avec le robot par le circuit de données défini précédemment (sur canal hertzien).



2.1.7 L'acteur caméras

Une à quatre caméras peuvent être embarquées. Parmi les caméras, l'une d'entre elles peut être sélectionnée, l'une d'entre elles peut être pilotée (cf. mission).

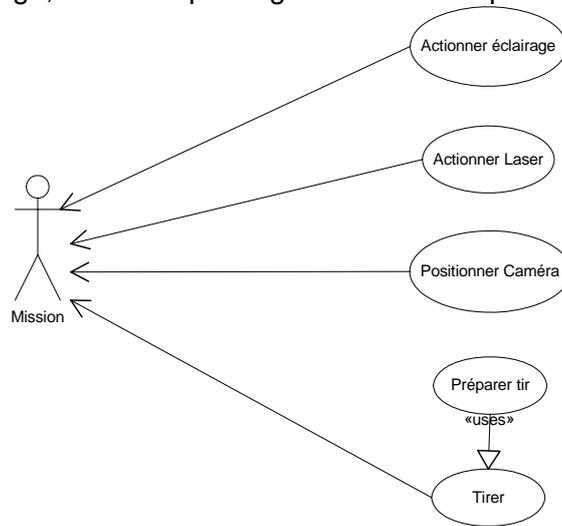
Le retour des images se fait par un circuit vidéo (sur canal filaire ou hertzien) qui est hors étude.



2.1.8 L'acteur mission

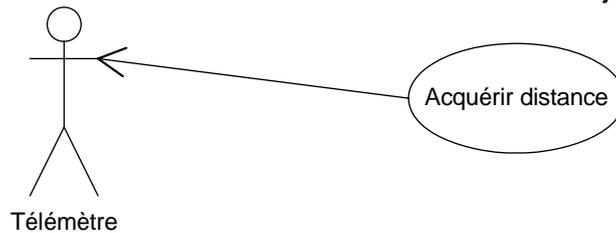
La mission a pour objet l'étude approfondie de l'objet suspect et la mise en œuvre des moyens nécessaires à sa destruction. Pour ce faire il est nécessaire de piloter la position de la caméra en azimut et en site par une unité de traitement variateur deux axes.

De plus des systèmes d'éclairage, d'aide au pointage laser et de tir peuvent être activés.



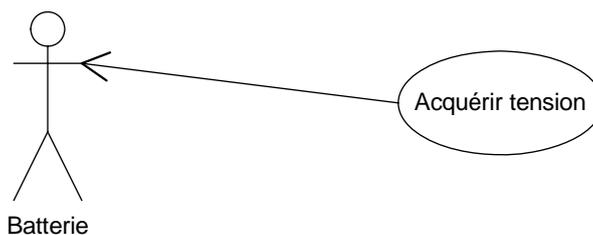
2.1.9 L'acteur télémètre

Le télémètre a pour rôle de mesurer la distance entre le robot et l'objet suspect.



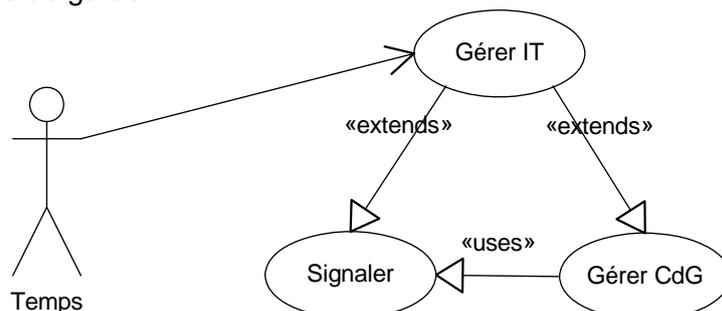
2.1.10 L'acteur batterie

La surveillance de la tension batterie est assurée en permanence et l'alarme est fournie en cas de besoin.



2.1.11 L'acteur temps

Deux bases de temps (3 s et 100 ms) sont nécessaires au fonctionnement du superviseur ainsi que la mise en œuvre de chiens de garde.



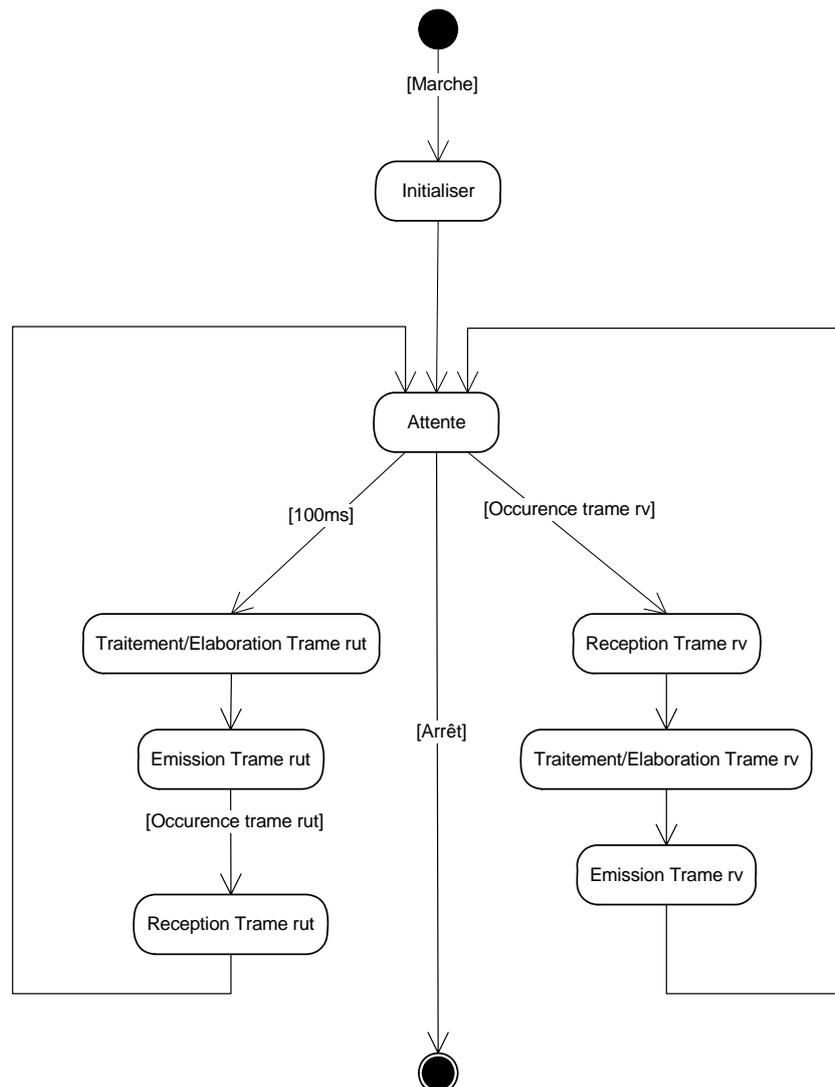
2.1.12 Diagramme d'état transition de la classe robot

Il est maintenant nécessaire de comprendre le comportement de la classe robot au cours du temps.

Pour ce faire il est indispensable de le modéliser par un diagramme "états-transitions".

Convention de notation : afin de bien différencier les communications entre la valise et le robot et entre le robot et les unités de traitement qui utilisent le même protocole, une extension "rv" sera appliquée aux éléments concernant la relation valise/robot et une extension "rut" sera appliquée aux éléments concernant la relation robot/unités de traitements.

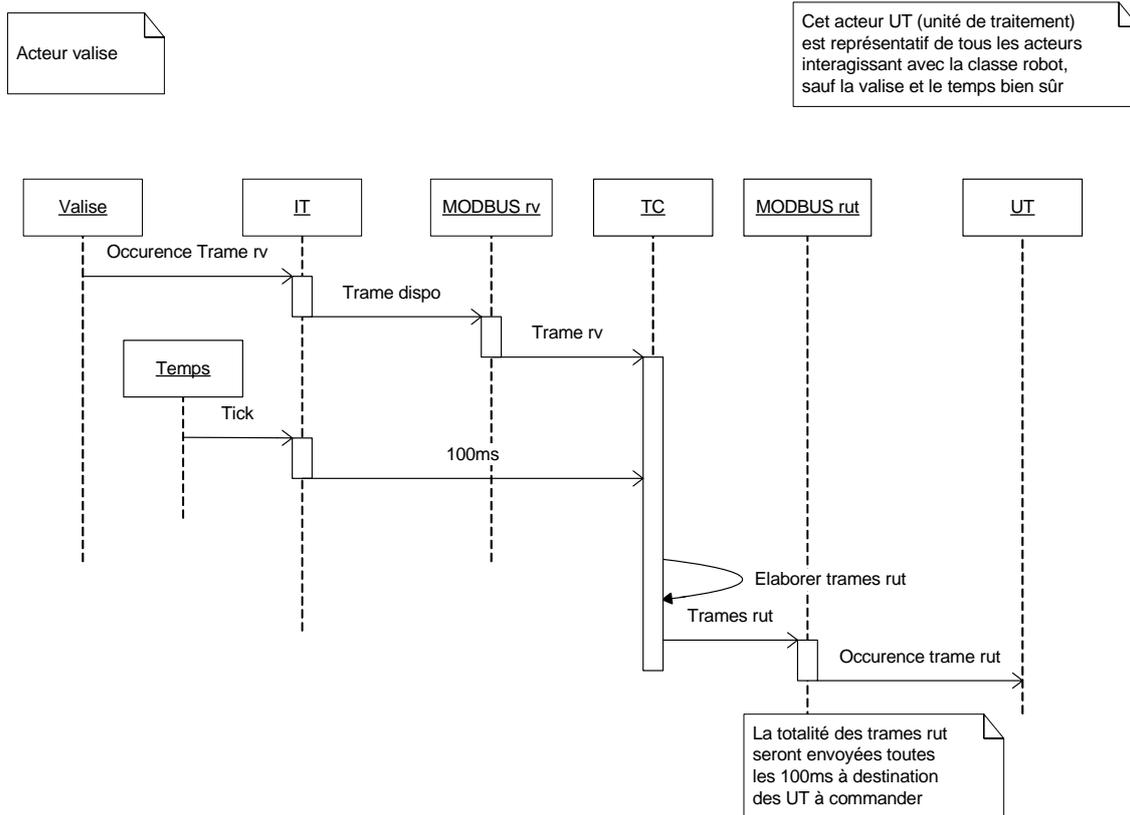
Vous noterez que "Trame rut" ou "Trame rv", conformément au protocole MODBUS, peuvent être, soit une TC, soit une TM, soit une demande de service, soit un acquittement. La différenciation n'a pas été faite à ce niveau pour des raisons de lisibilité.



Afin de bien comprendre la nature du travail attendu par chaque élément logiciel intervenant dans la réalisation d'un cas d'utilisation, il est nécessaire que chaque cas d'utilisation soit décrit par un diagramme de séquence mettant en évidence les contraintes temporelles associées.

Nous ne développerons pas la totalité des diagrammes de séquence mais nous allons décrire quelques cas d'utilisation permettant de mettre en évidence les classes qui vont participer à la réalisation de la supervision du robot.

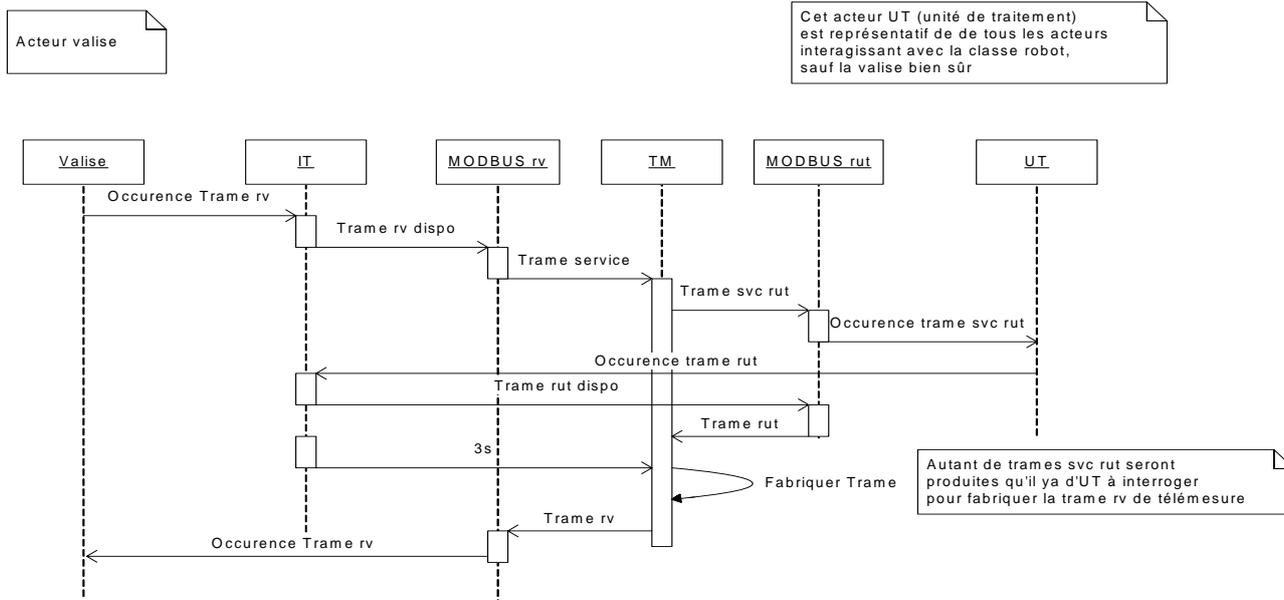
2.1.13 Diagramme de séquence du cas d'utilisation "recevoir données"



Nous constatons dans la description de ce cas d'utilisation le fait que l'objet MODBUS rv se comporte en esclave. L'acquittement de la trame rv n'apparaît pas dans ce modèle sachant qu'il est implicite dans le cadre du protocole utilisé.

Il est à noter que l'émission de la trame rut peut à la fois être cadencée par une interruption - c'est ce qui est modélisé - mais aussi l'être par l'occurrence de la trame rv.

2.1.14 Diagramme de séquence du cas d'utilisation "transmettre données"



Dans ce cas d'utilisation nous constatons que la production d'une trame par un esclave MODBUS ne peut être réalisée que si elle est explicitement demandée par le maître au travers d'une trame de service (svc).

Chaque cas d'utilisation participe à l'élaboration d'une trame rut à partir d'une restriction spécifique à l'unité de traitement de la structure de donnée TCrv reçu de la valise dans la TramTCrv.

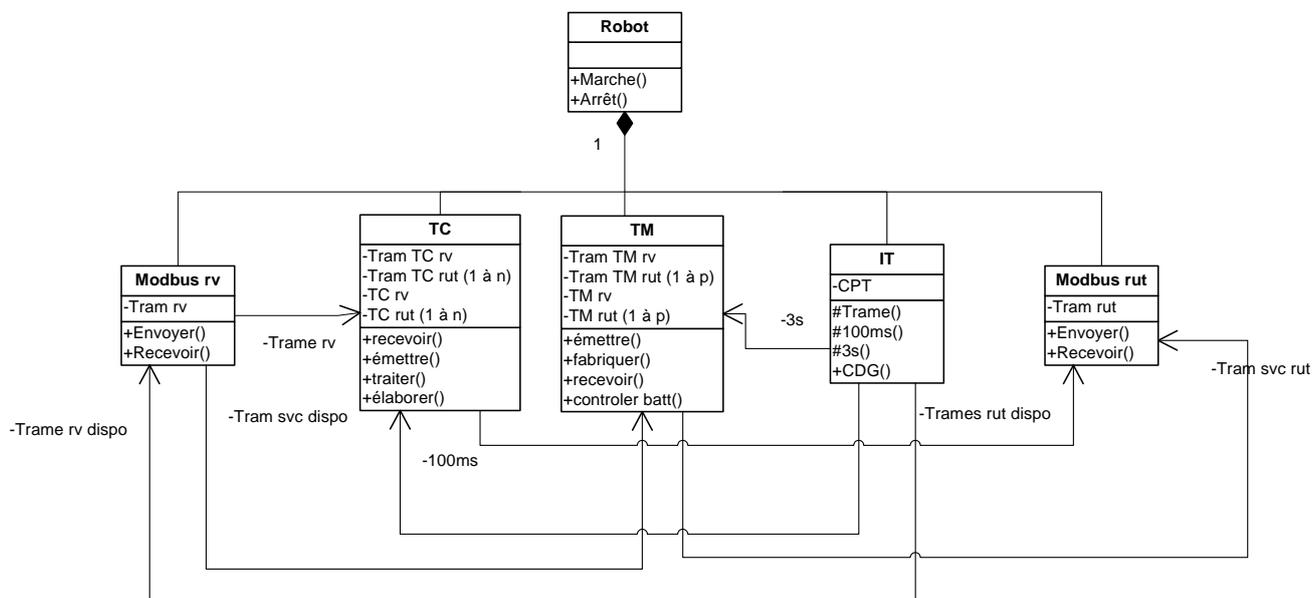
De même chaque cas d'utilisation participe à l'élaboration d'une trame Rv (TramTMrv) en la construisant à partir des informations reçues de chaque unité de traitement.

La structure principale est totalement décrite dans le chapitre 2.1.1 "Description des données échangées entre la valise et le robot". Ces structures de données sont définies comme des attributs des objets TC et TM ; les cas d'utilisation comme des compétences (méthodes) de ces deux objets.

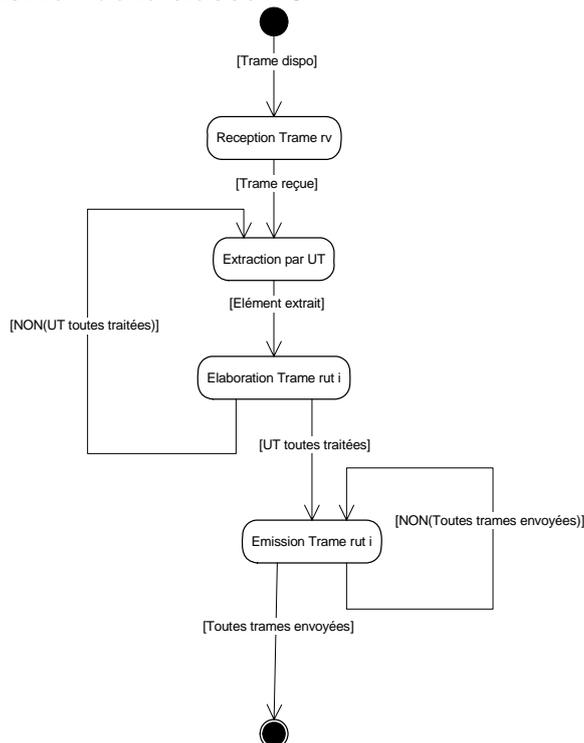
Pour ne pas surcharger les diagrammes suivants, la totalité des méthodes n'apparaîtra pas de manière explicite, ce qui n'ôtera rien à la compréhension globale de l'architecture du logiciel.

2.1.15 Diagramme des classes

Nous déduisons de notre étude les classes et les interactions entre classes nécessaires à la mise en œuvre du pilotage du robot, ainsi que la description du comportement de ces classes au cours du temps (nous limiterons ici à la classe TC).



2.1.16 Diagrammes état transition de la classe TC



2.1.17 Conception détaillée des classes Modbus

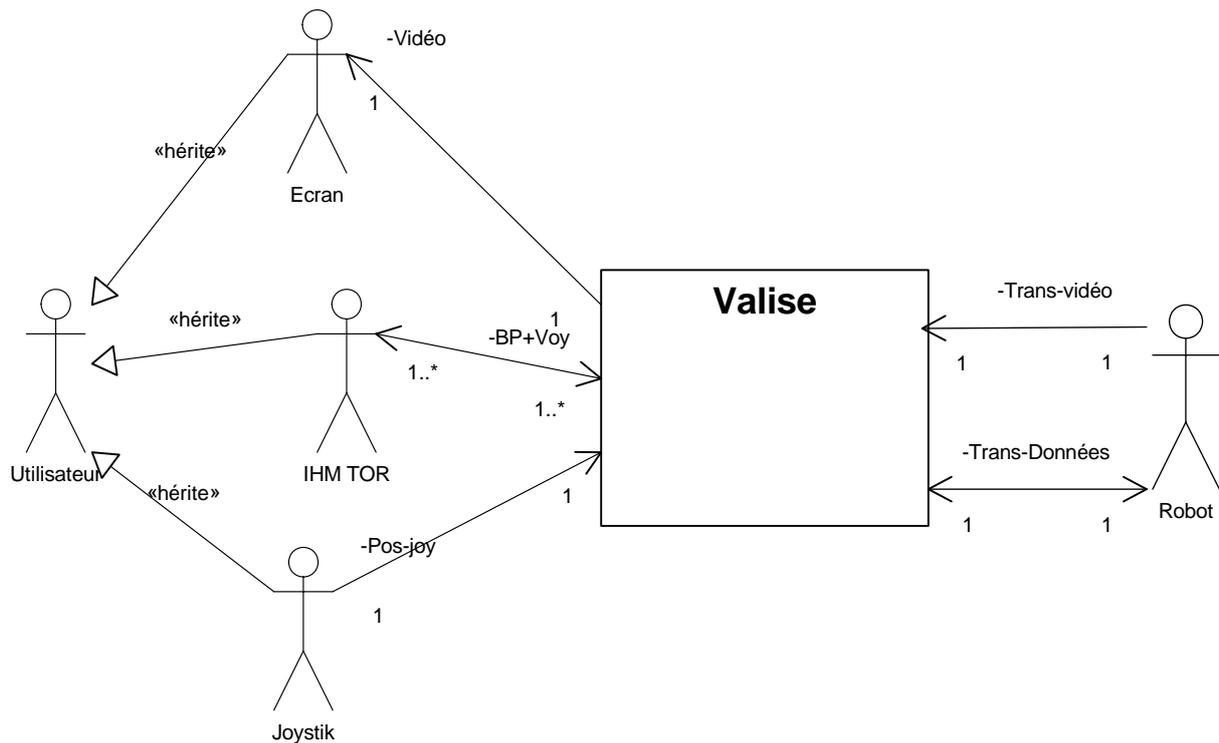
Les classes Modbus permettent d'assurer la construction de la trame (entête, corps, contrôle de trame), sa transmission sur le media, la vérification de son intégrité et son acquittement. La différence entre la classe Modbus rv et la classe Modbus rut est liée à son rôle de maître ou d'esclave et ne s'adresse pas au même média.

Nous nous intéressons aux états de chacune de ces classes ainsi qu'à l'élaboration du contrôle de CRC fourni en annexe dans la partie questionnement.

2.1.18 Modélisation de la partie commande de la valise

Ce début de modélisation vous est fourni comme base du travail pédagogique qui vous est demandé.

Vous pourrez comme pour le Robot mettre sous silence le circuit vidéo.



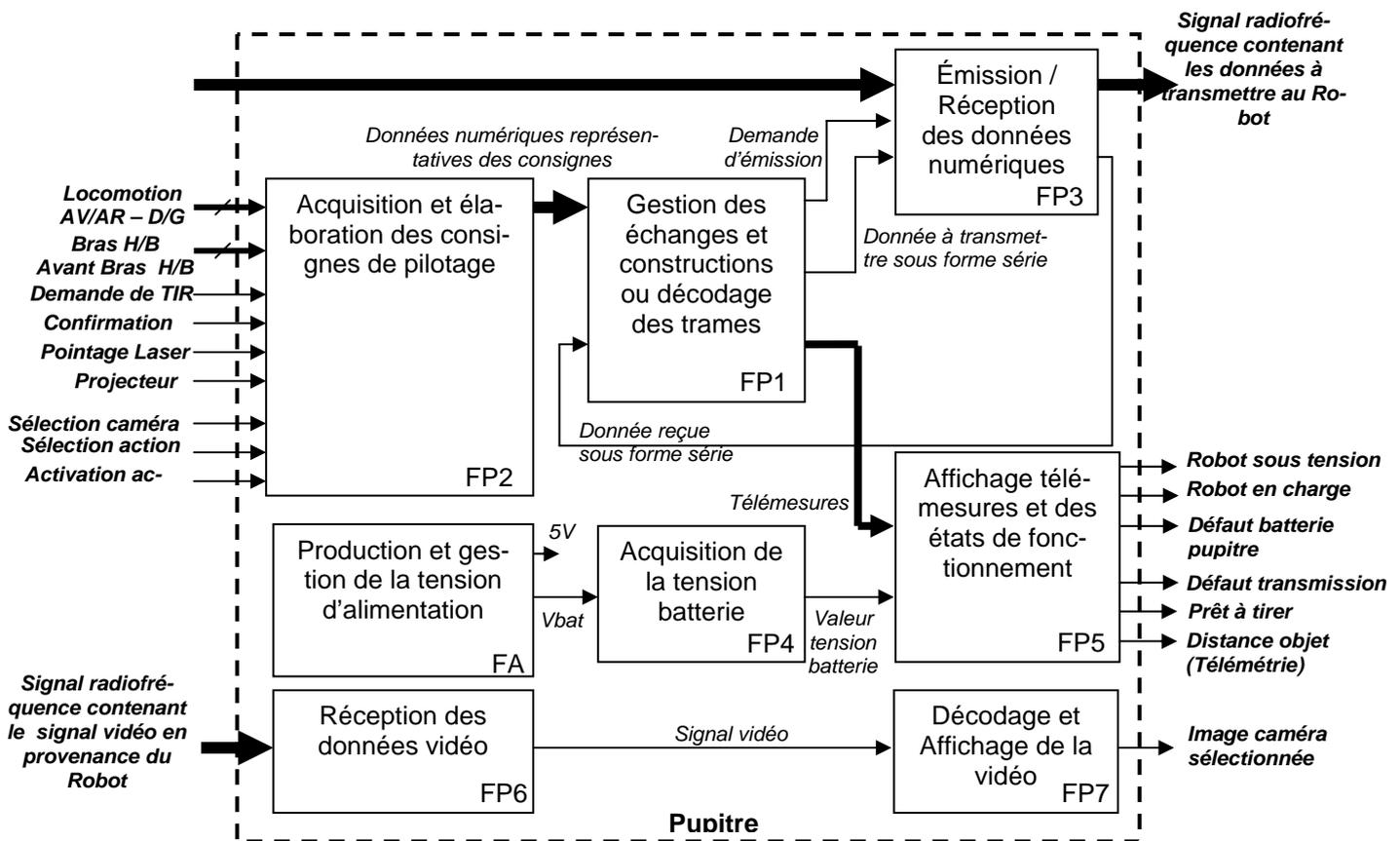
2.2 APPROCHE FONCTIONNELLE

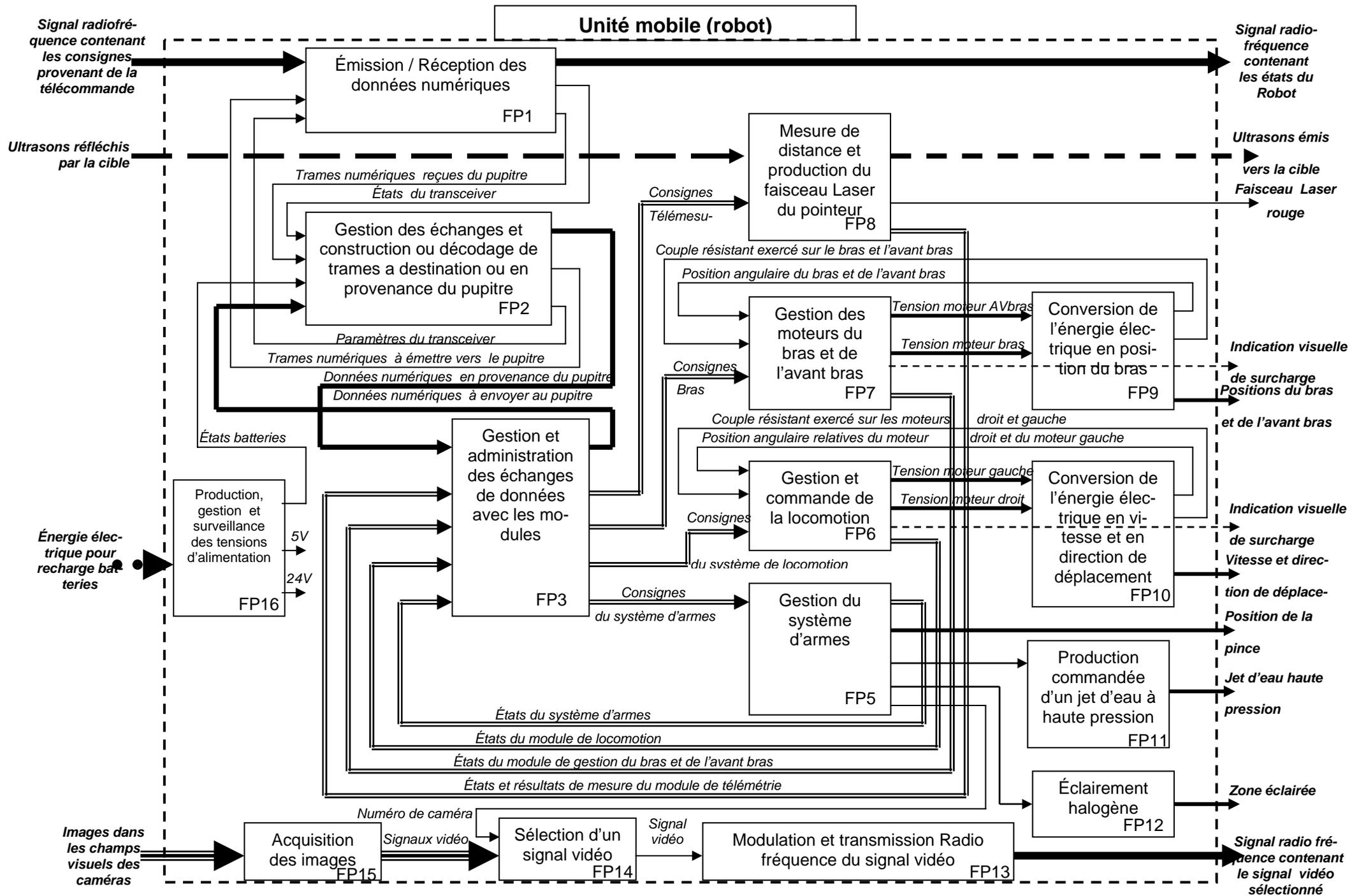
Afin de faciliter l'analyse de l'architecture de l'ensemble du système par l'électronicien, seuls les schémas fonctionnels de niveau 1 de la valise et du robot sont fournis ci après.

Il appartient au candidat d'associer les signaux ou informations d'entrées sorties des schémas fonctionnels aux acteurs définis dans le modèle UML.

Il est rappelé que l'architecture fonctionnelle ne permet pas de distinguer le traitement logiciel du traitement matériel associé à chaque fonction.

D'autre part, l'approche fonctionnelle étant orientée « flot de données », les architectures matérielles des BUS ne sont pas déductibles des représentations.





Sommaire du sujet

SUJET	27
A MODELISATION ET ANALYSE DU FONCTIONNEMENT	27
A.1 Analyse du système.....	27
A.2 Le sous système robot	27
A.3 Le sous-système Valise.....	28
A.4 Un contexte temps réel.....	28
A.5 Exploitation pédagogique	29
B ANALYSE DE LA TRANSMISSION DES DONNEES	30
B.1 Etude du principe de modulation du MODEM.....	30
B.2 Etude du modulateur intégré	32
B.3 Etude du récepteur intégré.....	35
B.4 Exploitation pédagogique	36
C ANALYSE ET IMPLÉMENTATION DE L'ACTEUR LOCOMOTION	37
C.1 Etude de la structure matérielle de la carte.....	37
C.1.1 Etude de l'alimentation des moteurs.....	37
C.1.2 Mesure et conversion du courant moteur	38
C.1.3 Présentation de l'asservissement	38
C.2 Etude de la structure logicielle de la carte.....	40
C.2.1 Mesure de la vitesse	40
C.2.2 Mesure du courant	40
C.2.3 Implémentation des correcteurs.....	40
C.3 Exploitation pédagogique	41
C.3.1 Présentation de la commande MLI en STI génie électronique.....	41
C.3.2 Implémentation et test d'une carte de variation de vitesse en STS Systèmes Electroniques	41

SUJET

A MODELISATION ET ANALYSE DU FONCTIONNEMENT

A.1 Analyse du système

L'approche de l'architecture logicielle du robot fait apparaître des diagrammes modèles écrits en langage UML (Unified Modeling Language) qui sont utilisés en particulier dans la description de projets sur la base d'une programmation objet. Certains modèles décrivent la structure statique du système étudié, d'autres la structure dynamique.

Q1. Généralités sur la modélisation

- a) *Citez les différents diagrammes utilisés dans le langage UML en précisant à quelle catégorie ils appartiennent et quel est leur rôle respectif.*

L'enrichissement du modèle se fait par une approche progressive de la solution tout en maintenant une structure logicielle opérationnelle.

- b) *Dans ces conditions comment qualifiez vous la méthode à mettre en œuvre pour réaliser complètement le projet ?*
 c) *Quelle stratégie de test allez-vous développer pour vérifier la conformité de votre système ?*

A.2 Le sous système robot



On souhaite compléter le modèle du robot tel qu'il est décrit dans la présentation afin de préciser certaines fonctionnalités.

Un détour s'impose par la classe IT dont le rôle est d'envoyer un événement toutes les 100 ms et les 3 s et de gérer des chiens de garde. Pour ce faire elle reçoit un événement appelé Tick, fourni par un composant spécialisé.

Q2. Description des méthodes de gestion du temps

- a) *Donnez les algorithmes des méthodes 100ms() et 3s() permettant d'envoyer les événements correspondants.*
 b) *Comment doit être évaluée la valeur du tick ?*
 c) *Décrivez par un algorithme la gestion d'un chien de garde, en précisant son rôle, ses paramètres et les signaux nécessaires à son utilisation.*
 d) *Décrivez la méthode trame().*
 e) *Quel est le rôle de l'attribut CPT ? Définissez son format.*

Nous avons vu que la classe TC a pour rôle de gérer la trame TramTCrv reçue de la valise.

Cette méthode permet d'extraire les données TCrv qui vont être partagées en autant de structures TCrut. Ces données, encapsulées dans des trames TramTCrut, permettent d'assurer le pilotage de chacune des unités de traitement.

Q3. Description des attributs de la classe TC

- a) *Décrivez et donnez les déclarations avant algorithme des structures de données des différents attributs de la classe TC.*
 b) *Vous donnerez très précisément le nombre et le contenu de chacun des attributs TCrut.*

Concentrons nous maintenant sur la méthode Modbus. A partir des documents fournis en annexe, vous vous intéresserez à la notion de fonctionnement en maître esclave et au rôle du contrôle de CRC des trames.

Q4. Le protocole Modbus

- a) *Donnez le principe du fonctionnement d'une communication réseau en Maître/Esclave.*
- b) *Quels sont les intérêts et les inconvénients de ce type de fonctionnement ?*
- c) *Qu'apporterait l'utilisation d'un protocole probabiliste dans notre cas et quel en serait son inconvénient principal ?*

Q5. Le contrôle de CRC

- a) *A quoi sert un contrôle de CRC dans une trame réseau ?*
- b) *Précisez le principe de son utilisation.*
- c) *Donnez en langage formel de type algorithmique la méthode permettant d'évaluer le CRC d'une trame Modbus dont le principe est fourni en annexe.*

A.3 Le sous-système Valise



Q6. Analyse de la valise

Pour ce sous-système dont le diagramme de contexte est fourni dans la présentation et en vous aidant de la description du sous système robot :

- a) *Etablir le diagramme état/transition spécifiant le comportement de la classe générique valise.*
- b) *Etablir le diagramme des classes du logiciel de gestion de la Valise.*

A.4 Un contexte temps réel

L'analyse du logiciel du robot fait apparaître dans le diagramme état/transition qu'il peut y avoir parallélisme de fonctionnement donc concurrence entre deux séquences initiées d'une part par l'état "Traitement/Elaboration Trame rut" et d'autre part par l'état "Réception Trame rv".

Q7. Nous nous trouvons face à un problème de gestion multitâche concurrente qui ne peut être traité de manière triviale.

- a) *Comment envisagez-vous de traiter un tel problème ? Indiquer les critères de choix permettant de gérer ce problème.*
- b) *Déterminer quels vont être les états qui pourront être mis en séquence dans une même unité d'exécution (Tâche ou Thread).*
- c) *Déterminer les priorités respectives de chacune des unités d'exécution.*
- d) *Donner le chronogramme d'exécution de chacune de ces unités.*
- e) *Qu'attendez-vous du mécanisme de séquençement des unités d'exécution ?*

A.5 Exploitation pédagogique

Elaboration des documents de présentation d'un système

Il est envisagé d'exploiter ce système afin d'illustrer les concepts mis en jeu dans les unités d'entrées sorties d'une part, dans la transmission des données d'autre part. Dans cette partie, il est demandé au candidat de préparer l'analyse du sous-système Valise et de prévoir les documents de présentation associés. Le public sera choisi dans des sections de STS IRIS. Ces documents pourraient être utilisés lors de différentes séquences d'enseignement.

Q8. A partir du diagramme de contexte fourni dans la présentation, faire la mise en situation générale (c'est-à-dire restreindre et délimiter le système étudié pour illustrer un ensemble de concepts, définir le système dans son contexte).

Q9. Élaborer les principaux schémas de présentation associés (schémas fonctionnels, schéma d'architecture matérielle, diagrammes UML, algorithmes, etc.).

Q10. Expliciter les caractéristiques principales retenues pour encadrer le travail des élèves (caractéristiques fonctionnelles, matérielles ou comportementales, liées au système étudié).

Attentes du jury

Pour les documents de présentation réclamés, on attend une production maximale de 4 pages format A4.

Critères d'évaluation du candidat :

- L'exactitude technique des documents présentés et le respect du système proposé,
- La pertinence pédagogique de la mise en situation choisie et des caractéristiques principales retenues,
- La cohérence et la pertinence pédagogique des représentations utilisées au niveau choisi,
- La pertinence des choix des travaux proposés et des résultats attendus des élèves,
- La qualité des descriptions et des explications proposées (exactitude technique, précision et concision, expression écrite, adéquation par rapport au contexte).

B ANALYSE DE LA TRANSMISSION DES DONNEES

B.1 Etude du principe de modulation du MODEM

On souhaite étudier le principe de modulation et de codage des données binaires afin de caractériser les contraintes à prendre en compte pour la reconstruction du message par le récepteur.

On rappelle que les échanges de données numériques entre la Valise et le Robot se font en half-duplex.
On note $m(t)$ le signal binaire modulant (signal informatif à transmettre).

Hypothèses simplificatrices :

- Dans la suite, sauf indication contraire, on suppose les transitions de fréquence instantanées lors d'un changement d'état logique du signal modulant (on néglige notamment le temps de réponse et les régimes transitoires du synthétiseur de fréquence).
- On considère que l'amplitude du signal $m(t)$ a la valeur unité ($M_{\max} = 1$ V) lorsque le bit à transmettre est un 1 logique et une amplitude nulle lorsque le bit à transmettre est un 0 logique.

Q11. Généralités

- a) Identifier les fonctions principales de l'unité mobile impliquées directement dans l'échange de données numériques par voie hertzienne.
- b) Rappeler les contraintes temporelles associées à une transmission HALF-DUPLEX.
- c) Identifier le modulateur / démodulateur intégré chargé de la transmission HF.
- d) Compléter le tableau du document réponse DR1, en indiquant pour chaque signal de l'interface numérique du CI concerné, son rôle, sa direction, puis indiquer la fonction des différents bus associés.

Q12. Expression du signal modulé

L'expression de la porteuse est : $V_p(t) = V_{\max} \cdot \sin(2\pi \cdot f_0 \cdot t + \varphi_0)$.

- a) Donner l'expression générale du signal $V_m(t)$ modulé en fréquence par un signal $m(t)$ faisant apparaître le terme Δf où,
 $\Delta f = f_1 - f_0$ où f_1 est la fréquence associée à $m(t)=1$ et f_0 , la fréquence associée à $m(t)=0$.
- b) Proposer une méthode de mesure permettant de relever le spectre COMPLET du signal modulé. Proposer un schéma en faisant apparaître les appareils utilisés et leur branchement sur la carte de l'émetteur. Quelle doit être la contrainte sur $m(t)$ pour visualiser ce spectre ?

Dans la suite $\varphi_0=0$.

Q13. Etude du codage de l'élément binaire utilisé pour le signal modulant

On note T_b le temps de transmission d'un bit fixant ainsi le rythme binaire du signal modulant.

Hypothèse simplificatrice : On suppose équiprobable l'apparition d'un 1 ou d'un 0 sur le signal modulant noté $m(t)$ (considéré comme un signal aléatoire à nombre d'états finis) pendant T_b , de plus, on suppose $m(t)$ stationnaire et ergodique d'ordre 2.

- a) **Quelle est alors la relation liant la densité spectrale bilatérale d'énergie normalisée du signal $m(t)$, noté $G(f)$ à sa fonction d'auto-corrélation (ici égal à sa covariance) noté $\Phi_{mm}(\tau)$ (où τ représente le retard de corrélation $(-\infty < \tau < +\infty)$). Donner le nom de ce théorème.**
- b) **Donner la définition de la fonction d'auto corrélation de $\Phi_{mm}(\tau)$ en fonction de $m(t)$ et τ .**

On définit $P1(\tau)$ la probabilité que $m(t).m(t - \tau) = 1$ dans l'intervalle de temps $[t_0, t_0 + \tau]$, où t_0 représente l'origine de temps des réalisations de $m(t)$ (t_0 correspond donc toujours au début de la transmission d'un caractère).

On rappelle que la probabilité ($P0(\tau)$) d'avoir $m(t).m(t - \tau) = 0$ dans le même intervalle de temps est alors égale à $(1 - P1(\tau))$.

On peut montrer que la relation définissant la fonction d'auto corrélation $\Phi_{mm}(\tau)$ de $m(t)$ peut se mettre sous la forme $\Phi_{mm}(\tau) = P1(\tau)$ si l'on considère qu'aucune puissance n'est contenue dans le signal $m(t)$ lorsque $m(t)=0$.

- c) **Compléter le tableau du document réponse DR2 en explicitant la valeur de $P1(\tau)$ et de $\Phi_{mm}(\tau)$ pour les intervalles de temps indiqués.**
- d) **Montrer que $\Phi_{mm}(\tau)$ peut se mettre sous la forme**

$$\Phi_{mm}(\tau) = \frac{1}{4} + \frac{Tri_{T_b}(\tau)}{4} = \frac{1}{4} + \frac{1}{4} \left(2 - \frac{\tau}{T_b} \right)$$

en fonction de τ et T_b pour toute la plage de variation de τ

$(-\infty < \tau < +\infty)$ puis tracer $\Phi_{mm}(\tau)$.

- e) **Exprimer $G(f)$ puis tracer cette fonction sur le document réponse DR3.**

Q14. Analyse des résultats et conséquences de ce type de codage

On se place en sortie du démodulateur, où l'on se propose d'étudier les contraintes de décodage du signal $m(t)$.

- a) **A partir de la représentation graphique de $G(f)$, indiquer si il est possible d'isoler la fréquence associée au rythme binaire T_b par un simple filtrage fréquentiel. A partir de l'expression de $G(f)$, justifier ce résultat. Expliquer l'intérêt que peut présenter ce type d'opération.**
- b) **Déduire le paramètre commun, dépendant de T_b à fixer obligatoirement sur le récepteur et sur l'émetteur pour rendre la transmission de donnée cohérente.**
- c) **Montrer que le résultat précédent peut être également appliqué aux liaisons RS485 utilisés pour le réseau MODBUS RTU du robot.**
- d) **Quel avantage offre le codage Manchester au regard du codage exploité dans ce système ? Quelle application directe de cet avantage est utilisée dans les réseaux Ethernet ? Justifier et donner un cas concret.**

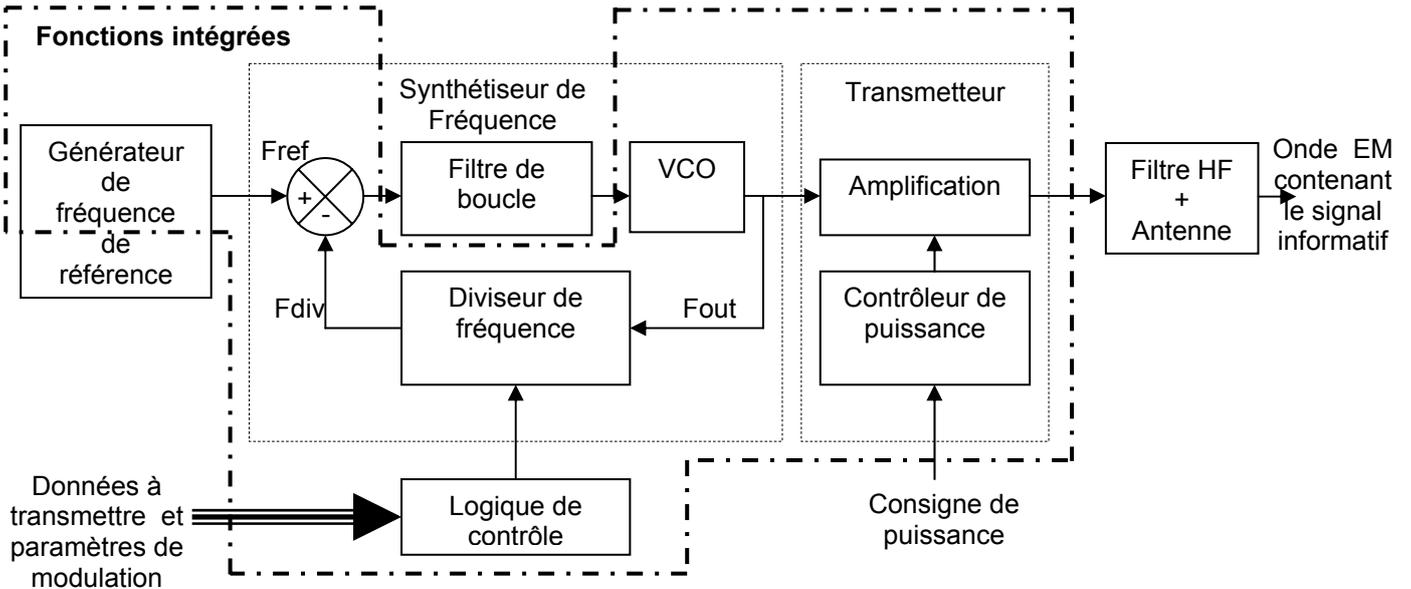
B.2 Etude du modulateur intégré

On s'intéresse au modulateur de fréquence intégré au circuit identifié précédemment.

Hypothèse simplificatrice : Sauf indication contraire, l'étude des différents éléments se fera autour du point de fonctionnement nominal, boucle verrouillée.

Q15. Etude de la structure du modulateur

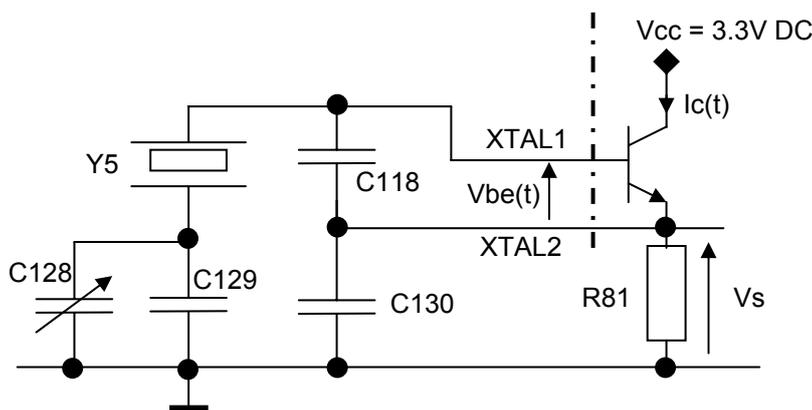
Schéma bloc du modulateur de fréquence



- Quel est le rôle du synthétiseur de fréquence ? Rappeler le nom de la structure utilisée pour le réaliser, puis expliciter brièvement le rôle de chacune des fonctions qui le compose.
- Pour quelle raison la donnée à transmettre a-t-elle une action sur le diviseur de fréquence ? Expliquer simplement le fonctionnement du modulateur de fréquence en faisant intervenir le signal modulant $m(t)$.
- Donner et expliquer la structure d'une trame destinée au paramétrage du Circuit intégré étudié ici en provenance du microcontrôleur. Expliciter le principe de transmission entre le microcontrôleur et le circuit étudié.

Q16. Étude du générateur de fréquence de référence

On donne le schéma simplifié du générateur de fréquence de référence (les composants internes au CI utiles à la polarisation du transistor n'apparaissent pas) :



Texte du sujet

Pour le point de polarisation choisi qui impose V_{be0} et I_s (courant de saturation du transistor), lorsque les oscillations existent à la fréquence f_0 on admet que la tension $V_{be}(t)$ peut s'écrire sous la forme :

$$V_{be}(t) = V_{be0} + V_1 \cdot \cos(\omega_0 \cdot t)$$

- a) **Montrer que la relation liant le courant collecteur (noté $I_c(t)$) à la tension $V_{be}(t)$ peut s'écrire sous la forme suivante en explicitant les différents termes. Préciser l'ordre de grandeur de V_T pour une température de 300 K. Justifier.**

$$I_c(t) = I_s \cdot \exp\left(\frac{V_{be0}}{V_T}\right) \cdot \exp\left(\frac{V_1 \cdot \cos(\omega_0 t)}{V_T}\right) = I_s \cdot \exp\left(\frac{V_{be0}}{V_T}\right) \cdot \sum_n a_n \cdot \cos(n\omega_0 t)$$

- b) **Etablir le schéma équivalent en petits signaux du transistor bipolaire (sans faire apparaître les impédances parasites) et montrer qualitativement que ce modèle peut alors être considéré comme un amplificateur à transconductance de facteur d'amplification noté g_m dans certaines conditions qu'il est demandé d'expliciter.**
- c) **Donner le nom du type d'oscillateur utilisé ici. En utilisant le modèle du transistor qui vient d'être établi, donner le schéma équivalent de ce montage pour les petits signaux. Faire clairement apparaître le modèle équivalent du quartz).**
- d) **Mettre en évidence la chaîne d'amplification ainsi que le réseau de contre-réaction.**

Dans la suite, on négligera les éléments modélisant les pertes du quartz.

On donne les valeurs numériques des éléments de la branche série du quartz :

$L_y = 0,00804$ H, $C_y = 30$ fF.

Pour la branche parallèle : $C_{yp} = 8$ pF

- e) **Quel est obligatoirement le type d'impédance du quartz pour que les oscillations existent ? Justifier votre réponse en exploitant les modèles équivalents des questions précédentes.**
- f) **Exprimer littéralement la fréquence d'oscillation en fonction des éléments du montage. Montrer que cette fréquence est réglable. Calculer numériquement les extrema de réglage de cette fréquence.**
- g) **On souhaite régler manuellement la fréquence d'oscillation du montage à l'aide d'un oscilloscope. Proposer une méthode de réglage, faisant apparaître le mode opératoire, ainsi que l'ensemble des appareils de mesures utilisés. On veillera à exposer clairement les précautions à prendre pour effectuer cette opération. Proposer un schéma représentant le circuit de mesure.**

Q17. Étude du synthétiseur de fréquence

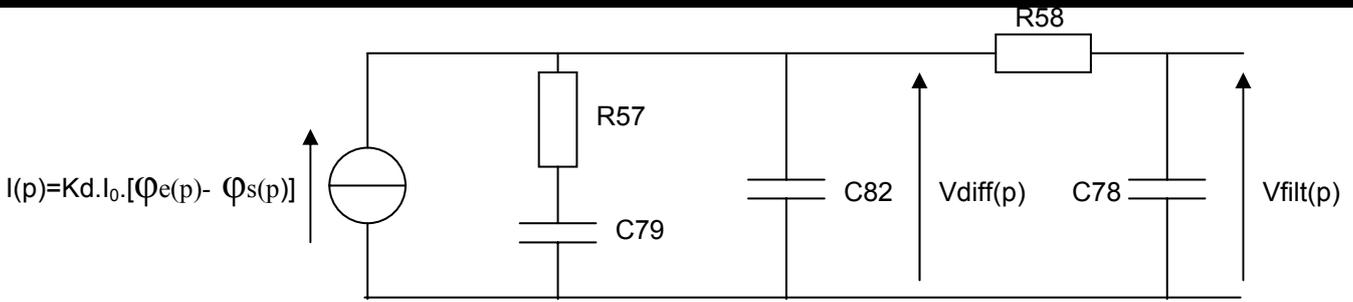
On cherche à évaluer le rythme binaire maximum du signal modulant en analysant la fonction de transfert du synthétiseur de fréquence.

Le comparateur de phase à pompe de charge est chargé par le filtre placé entre les broches 41 (FIL1) et 42 (VCON) (Par souci de clarté, le modèle du générateur de courant a été simplifié).

Le développement de Taylor de l'expression liant la fréquence en sortie du VCO (f_{OUTVCO}) à sa tension de commande (V_{evco}) autour du point de fonctionnement est donné ci-dessous :

$$f_{OUTVCO} = f_{VCO0} + K_{VCO} \cdot V_{evco} \quad \text{où } K_{VCO} = 2,67 \text{ MHz} \cdot \text{V}^{-1}, f_{VCO0} = 433 \text{ MHz.}$$

On modélise, dans l'espace de Laplace, l'ensemble comparateur de phase – filtre de boucle avec le schéma électrique équivalent suivant :



Avec :

$$Kd \cdot I_0 = 225 \cdot 10^{-6} \text{ A/rad,}$$

$\Phi_e(p)$: la phase instantanée du signal d'entrée,

$\Phi_s(p)$: la phase instantanée du signal de sortie de la boucle

$V_{filt}(p)$: la tension de sortie de l'ensemble comparateur de phase – filtre de boucle.

- a) **Exprimer littéralement $V_{filt}(p)$ en fonction de $\Phi_e(p) - \Phi_s(p) = \Delta\Phi(p)$ et des éléments du montage puis donner la fonction de transfert correspondante sous forme canonique et numérique.**

On montre que la fonction de transfert précédente peut se mettre sous la forme :

$$T_{filt}(p) = \frac{V_{filt}(p)}{\Phi_e(p) - \Phi_s(p)} = \frac{k}{p} \times \frac{(1 + \tau d \cdot p)}{(1 + \tau_1 \cdot p) \cdot (1 + \tau_2 \cdot p)}$$

On donne :

$$\tau d \approx 18,48 \mu\text{s}$$

$$\tau_1 \approx 1,722 \mu\text{s}$$

$$\tau_2 \approx 394 \text{ ns}$$

$$k \approx 36,437 \cdot 10^3 \text{ V/rad}$$

Sauf indication contraire, on prendra dans la suite le rapport $N/D = 0,02366$.

- b) **Compléter le schéma bloc du document réponse DR4, liant l'expression de la fréquence instantanée $F(p)$ à la phase instantanée $\Phi(p)$. Dans le cas particulier où $f(t) = \text{Constante} = f_c$, exprimer $\Phi(p)$ en fonction $F(p)$.**
- c) **Compléter le schéma – bloc du document réponse DR5 en faisant apparaître les fonctions de transfert.**
- d) **Tracer sur le document réponse DR6, les diagrammes de Bode asymptotiques et approchés de gain et de phase de la fonction de transfert en boucle ouverte $T_{BO}(f) = F_{div}(f) / \mathcal{E}(f)$, puis déterminer graphiquement les marges de phases et de gain ainsi que les fréquences associées à chacune de ces marges. Discuter de la stabilité de la boucle.**

On montre que la fonction de transfert simplifiée en boucle fermée $F_{out}(p)/F_{ref}(p)$ peut se mettre sous la forme suivante :

$$\frac{F_{out}(p)}{F_{ref}(p)} = T_{BF}(p) = \frac{D}{N} \frac{(1 + \tau d \cdot p)}{\left(1 + \frac{2m}{\omega_0} p + \frac{p^2}{\omega_0^2}\right) \cdot (1 + \tau b \cdot p) \cdot (1 + \tau c \cdot p)}$$

$$\tau d \approx 18,48 \mu\text{s}$$

$$\tau_b \approx 14,4 \mu\text{s}$$

$$\tau_c \approx 382 \text{ ns}$$

$$\omega_0 \approx 344,174 \cdot 10^3 \text{ rad/s}$$

$$m \approx 0,636$$

- e) Expliquer l'influence du polynôme numérateur sur les comportements dynamiques et harmoniques de la boucle. Quelle méthode de correction a été envisagée par le concepteur ? Justifier.
- f) Déterminer les erreurs statique, de traînage puis d'accélération du synthétiseur de fréquence. Conclure sur la précision de cet asservissement.
- g) On suppose le signal modulant de type binaire. Expliquer brièvement pourquoi le temps de réponse à 1% permet d'approximer le rythme binaire théorique maximum de $m(t)$.

On montre que la réponse temporelle indicielle de la PLL à un échelon de fréquence Δf lié à une variation instantanée (mais suffisamment faible) du rapport N/D peut être approximée par :

$$f_{out}(t) = \Delta f - \frac{\Delta f}{\sqrt{1-m^2}} \cdot e^{(-m \cdot \omega_0 \cdot t)} \cdot \cos(\omega_0 \cdot \sqrt{1-m^2} \cdot t + \varphi)$$

- h) Le concepteur a choisi un débit binaire de 19200 bits par secondes pour une excursion en fréquence de 5 KHz, justifier ce choix.
- i) Expliciter qualitativement les conséquences de l'analyse qui vient d'être faite sur le spectre du signal modulé. Mettre en évidence les avantages et les défauts de ce type de synthétiseur.
- j) Calculer le dépassement de fréquence maximum atteint lors du régime transitoire de la réponse indicielle précédente.

Q18. Étude de l'adaptation de puissance

- a) On souhaite obtenir une puissance de sortie du transmetteur de +14 dBm dans la bande des 433MHz, Donner l'ensemble des paramètres et composants à fixer en les identifiant puis en précisant leurs valeurs pour obtenir ce résultat. Exprimer en milliwatts la puissance de sortie correspondante.
- b) L'antenne présente une amplification de puissance de +2 dBi, à la fréquence nominale. Définir le dBi.
- c) Calculer la puissance du signal émis dans la bonne direction.

B.3 Etude du récepteur intégré

Q19. Principe de fonctionnement du récepteur

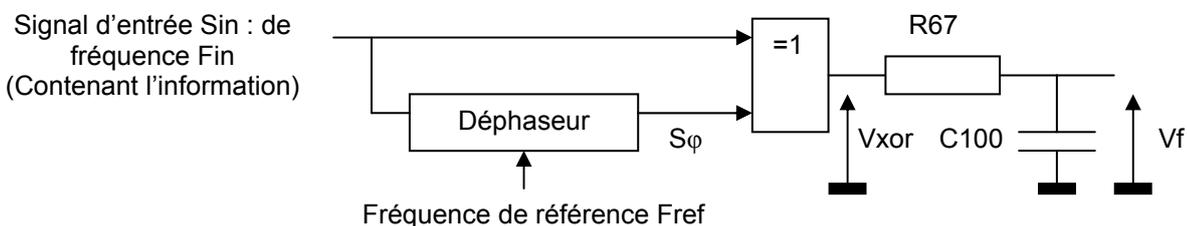
- a) Donner un schéma fonctionnel simplifié du récepteur intégré. Indiquer le type et le principe de fonctionnement de ce récepteur.

On donne la forme du signal reçu : $V_r(t) = A_r \cdot \cos[2\pi[f_r + m(t) \cdot \Delta f] \cdot t]$

- b) Donner l'équation du signal en sortie de chaque étage de changement de fréquence (après filtrage) et montrer l'intérêt de ce type de structure au regard des variations relatives de fréquences et des caractéristiques des composants utilisés dans ce système.
- c) Donner le principe de fonctionnement d'un filtre SAW. Quel est l'intérêt de ce type de filtre ?

Q20. Étude du discriminateur de fréquence

On donne le synoptique du discriminateur de fréquence intégré :



Sin est un signal périodique rectangulaire de fréquence F_{in} et d'amplitude E de 3 V.
 $V_{oh_{min}}$ de la porte XOR = 3 V = E.

Avec le signal S_{ϕ} identique à Sin mais déphasé de $\Phi_{S_{\phi}}$.

$\Phi_{S_{\phi}}$ est liée à F_{in} et F_{ref} par la relation (développement de Taylor limité à l'ordre 1 de la fonction de transfert du déphaseur pour de faibles variations de F_{in} autour de F_{ref}) :

$$\Phi_{S_{\phi}} = 2,73 \cdot ((F_{in}/F_{ref}) - 0,8).$$

On rappelle que $F_{in} = F_{ref} + m(t) \cdot \Delta f$

- a) **Expliciter le rôle de cette fonction.**
- b) **Exprimer le rapport cyclique α du signal V_{xor} en fonction de $m(t)$, puis en déduire l'expression de la valeur moyenne de $V_f(t)$.**
- c) **Donner la sensibilité (en V/kHz) de ce discriminateur de fréquence pour $F_{ref} = 455$ kHz, puis calculer $\langle V_f(t) \rangle$ pour une excursion en fréquence du signal reçu de 5 kHz.**

Q21. Principe du « conditionneur de données » (DataSlicer)

- a) **Expliciter le rôle de cette fonction et indiquer son principe de fonctionnement.**
- b) **Compte tenu du type de codage de l'information binaire utilisé pour $m(t)$, donner le mode de fonctionnement du « conditionneur de données », sachant que l'on n'exclut pas la transmission successive et prolongée de nombreux bits identiques (suite de 0 ou de 1).**

B.4 EXPLOITATION PÉDAGOGIQUE

On souhaite exploiter le sous-système qui vient d'être étudié afin d'introduire la transmission de données numériques en **STS Systèmes Electroniques**.

On s'intéresse au champ technologique TELECOMMUNICATION, TELEPHONIE ET RESEAUX TELEPHONIQUES pour développer le questionnement pédagogique demandé.

L'approche sera limitée à la transmission de données par voie hertzienne en modulation de fréquence et aux concepts abordés dans l'extrait de référentiel fourni en annexe.

Q22. Proposer une problématique et une mise en situation permettant d'introduire ce mode de transmission en prenant pour support ce système. Délimiter soigneusement l'analyse et l'étude proposée aux étudiants. Préciser les supports pédagogiques utilisés.

Q23. Élaborer les structures des séquences et progressions pédagogiques, en mettant en évidence les connaissances à mobiliser ou à développer pour atteindre les objectifs du référentiel. Détailler les déroulements et les objectifs intermédiaires des différentes activités.

Q24. Proposer une approche critique des concepts abordés dans votre développement pédagogique au regard de l'état de l'art et en faisant l'analogie avec des systèmes utilisés dans la vie quotidienne (télévision numérique, réseaux, etc.). Expliciter les contraintes et limites technologiques ou théoriques de votre approche vis-à-vis du niveau des étudiants de STS SE.

Attentes du jury :

On attend une production maximale de 2 à 4 pages format A4.

Critères d'évaluation du candidat :

- L'exactitude technique des documents présentés et le respect du système proposé,
- La pertinence pédagogique de la mise en situation choisie et des caractéristiques principales retenues,
- La justification des contraintes matérielles et temporelles,
- La validité et la réalité technologique des concepts abordés,
- Le respect des exigences du référentiel,
- La qualité des descriptions et des explications proposées (exactitude technique, précision et concision, expression écrite, adéquation par rapport au contexte).

C ANALYSE ET IMPLÉMENTATION DE L'ACTEUR LOCOMOTION

Dans cette partie nous allons nous placer dans la situation de l'électronicien qui doit implémenter une carte de pilotage de moteurs asservis en vitesse ou position à partir d'une consigne reçue par un canal de transmission Modbus. Sachant que la même carte est utilisée pour la propulsion et pour le positionnement d'accessoires motorisés, notre étude porte sur le pilotage en vitesse pour la locomotion du robot.

C.1 ETUDE DE LA STRUCTURE MATÉRIELLE DE LA CARTE

Indications techniques :

- Le schéma de la carte est fournie en annexe.
- La fréquence des signaux MLI est fixée à 613 Hz.
- La DEL D10 s'allume lorsque l'un des moteurs rencontre une surcharge en couple.
- Les vitesses de rotation sont exprimées en radian par seconde.
- Les moteurs sont de type S220F du constructeur PARVEX équipés de freins à manque de courant et d'un codeur incrémental solidaire de l'axe moteur de 1024 points par tour, délivrant deux signaux en quadrature de phase, disponibles sur des sorties à collecteur ouvert de transistors PNP.
- Lorsque les moteurs sont à l'arrêt (tension d'alimentation nulle), le frein doit impérativement être activé.
- L'axe 1 correspond au moteur gauche et l'axe 2 au moteur droit du robot.
- Si AXE2PHA = AXE1PHA = 1 et AXE2PHB = AXE1PHB = 0 le robot avance en ligne droite à vitesse maximale.
- Si le signal BRAKES = 1, alors, les deux freins moteurs à manque de courant sont actifs.

C.1.1 ETUDE DE L'ALIMENTATION DES MOTEURS

La carte variateur fonctionne en mode locomotion :

- le contenu de l'adresse 0x0011 de la table d'échange MODBUS représente la consigne de vitesse de déplacement du robot :
0x0011 > 0 : avance, 0x0011 = 0 : manœuvre sur place, 0x0011 < 0 marche arrière
- le contenu de l'adresse 0x0012 de la table d'échange MODBUS représente la consigne de direction de déplacement du robot :
0x0012 > 0 : virage à droite, 0x0012 = 0 : tout droit, 0x0012 < 0 virage à gauche.

L'axe 1 correspond au moteur gauche et l'axe 2 au moteur droit du robot.

Q25. Commande d'un pont en H

Sauf indication contraire, on s'intéresse à la commande de l'axe 2, le principe de commande de l'axe 1 étant similaire.

- a) **Donner la relation liant la tension moyenne aux bornes du moteur au rapport cyclique α des signaux AXEPH2A ou AXEPH2B, sachant que pour un sens de rotation donné, l'un des deux signaux est en permanence au 0 logique (on négligera la chute de tension dans la résistance shunt).**

Les signaux MLI permettant de piloter l'axe 2 sont produits par le Timer 0 de l'ATMEGA88 fonctionnant en mode « Phase Correct PWM » avec une résolution de 8 bits, le compteur évoluant entre 0 et 255.

- b) **Identifier les registres du Timer0 de l'ATMega88 mis en jeu et déterminer leur contenu pour obtenir le fonctionnement souhaité pour chaque sens de rotation. Préciser le facteur de pré division N à appliquer au signal d'horloge interne (f_{clk-IO}).**
- c) **Exprimer le rapport cyclique α en fonction des contenus des registres OCR0A et OCR0B, puis montrer que le sens de rotation du moteur peut être fixé à l'aide de ces deux registres.**
- d) **On souhaite asservir la vitesse de rotation. Montrer que suivant le sens de rotation souhaité, il faudra sélectionner le registre à piloter.**
- e) **Sachant que l'autre pont en H est piloté de manière similaire par les sorties OC1A et OC1B (cf. schéma électrique) correspondant respectivement aux registres OCR1A et OCR1B du Timer1 de l'ATMega88, compléter le tableau du document réponse DR7.**

C.1.2 MESURE ET CONVERSION DU COURANT MOTEUR

Sauf indication contraire, on s'intéresse à l'axe 2, le principe de mesure du courant de l'axe 1 étant similaire.

Q26. Etude de la chaîne analogique

- a) Donner la fonction de transfert liant le courant moteur à la tension IAXE2.
- b) Sachant que l'entrée du CAN intégré contient un suiveur, montrer que la tension IAXE2 représente approximativement la valeur moyenne sur une période MLI du courant moteur.
- c) Donner la relation liant le couple moteur au courant moteur, à l'aide des documents constructeur. Exprimer numériquement cette relation.
- d) Quelle peut être l'utilité du signal DGAXE2 ? Expliquer.

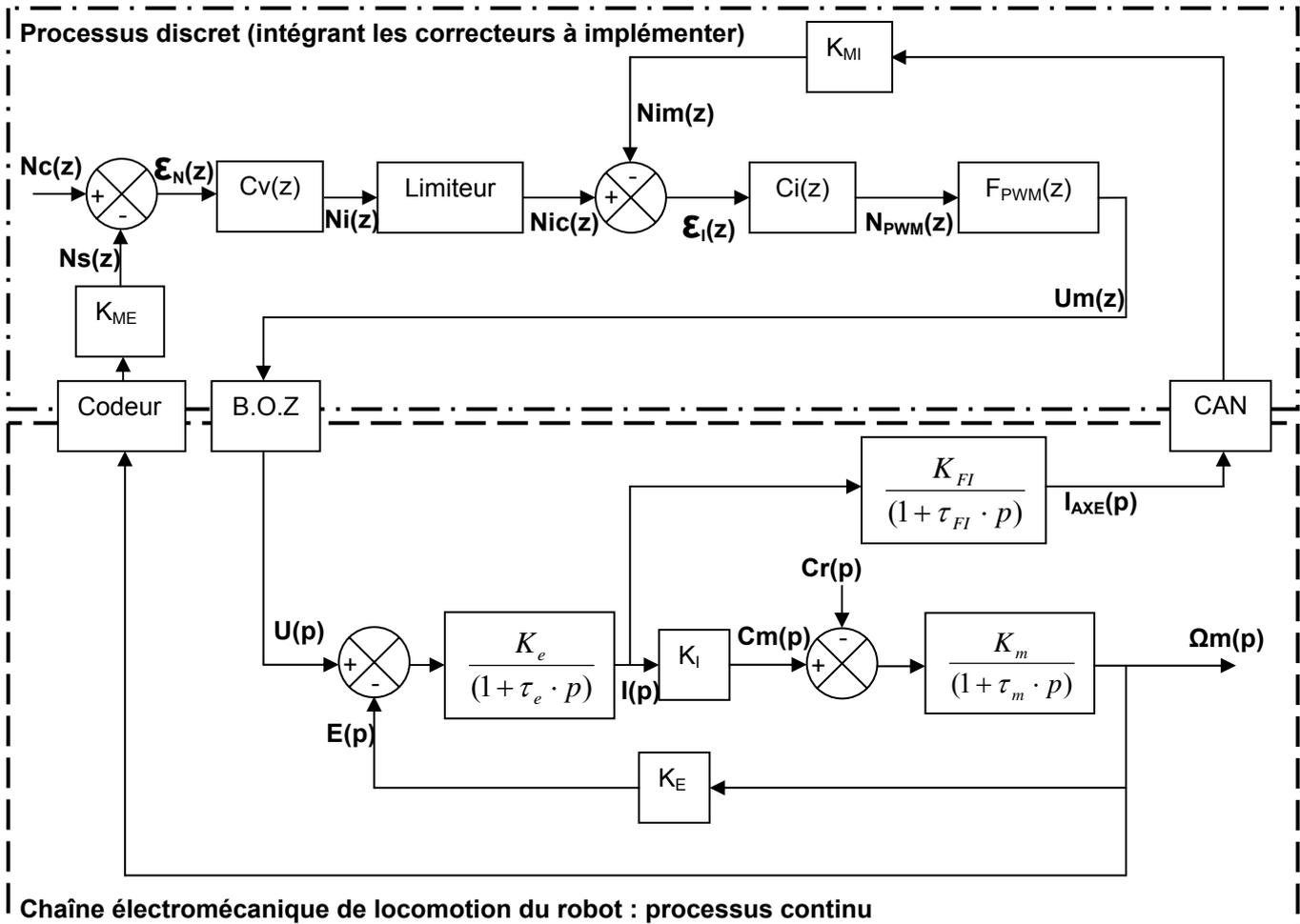
Q27. Etude de l'acquisition de l'information courant moteur par le microcontrôleur

- a) Donner la plage de variation du courant et de couple moteur mesurable par le microcontrôleur (la tension de référence du CAN est de 5V). Justifier.
- b) Donner la plus petite variation de couple mesurable par le microcontrôleur.
- c) Quelle est l'atténuation (en dB) du fondamental du signal MLI ? Quelle est l'influence de ce signal sur la précision de la mesure ? Proposer une méthode numérique permettant d'améliorer cette précision.
- d) Calculer la tension IAXE2 pour le couple nominal du moteur.

C.1.3 PRÉSENTATION DE L'ASSERVISSEMENT

L'objectif de cette partie est de modéliser la boucle d'asservissement de vitesse et de déterminer les équations de récurrence. Celle-ci permettent d'implémenter les algorithmes des correcteurs numériques équivalents à $Cv(z)$ et $Ci(z)$ (sans calculer leurs paramètres) dans le microcontrôleur.

Schéma bloc du modèle de l'asservissement de vitesse du moteur gauche :



- L'automaticien a choisi d'utiliser des correcteurs numériques PID ($Cv(z)$) et PI ($Ci(z)$) pour optimiser le pilotage du système et l'adaptabilité de l'algorithme de calcul.
- On notera T_e la période d'exécution des algorithmes de correction cherchés.
- $Nim(z)$ est un nombre entier qui représente le courant moteur avec une précision de 10 mA ($Nim = 1 \Rightarrow I = 10 \text{ mA}$).
- $Ns(z)$ est un nombre entier qui représente la vitesse de rotation du moteur avec une précision de 0,1 radian par seconde ($Ns = 1 \Rightarrow \Omega_m = 0,1 \text{ rad/s}$).

Q28. Modèle du processus continu et éléments d'analyse du comportement dynamique

On donne les paramètres mécaniques de la chaîne cinématique de propulsion du robot ramenés sur l'axe moteur :

Moment d'inertie $J_m = 2,9 \text{ kg.m}^{-1}$

Coefficient de frottement visqueux $K_{fv} = 1,33 \cdot 10^{-5} \text{ N.m.rad}^{-1} \cdot \text{s}$.

Les roues ont un diamètre de 26 cm et sont reliées à l'axe moteur par un réducteur équivalent à la chaîne cinématique complète (moto réducteur planétaire, galet de friction, etc.) de rapport $1/13^{\text{ième}}$.

- En vous aidant des résultats des questions précédentes et des documents constructeurs, exprimer littéralement puis numériquement l'ensemble des paramètres des modèles analogiques (moteur, chaîne de mesure du courant, etc.).**
- Que représente l'excitation externe $Cr(p)$? Donner un exemple concret provoquant une variation de $Cr(p)$.**

La grandeur $N_{PWM}(z)$ représente le contenu de l'un des registre OCRxx.

- Exprimer la relation liant N_{PWM} à la tension moyenne aux bornes du moteur pour un sens de rotation fixé et justifier la présence du bloqueur d'ordre zéro (B.O.Z.).**
- Expliquer brièvement l'intérêt, du point de vue des performances de l'asservissement, de la boucle imbriquée de régulation de courant.**
- Calculer le temps que met le robot départ arrêté pour atteindre la vitesse de 1 m/s sur terrain plat si le couple moteur reste égal à sa valeur nominale lors de la phase d'accélération (supposée constante).**

Q29. Extraction des équations de récurrence

- Donner les valeurs numériques des coefficients K_{MI} et K_{ME} .**

On donne la forme général des correcteurs $Cv(z)$ et $Ci(z)$:

$Cv(z) = K_{cv} \cdot \left(1 + \frac{T_e}{T_{iv}} \cdot \frac{1}{z-1} + \frac{z-1}{K_{dv} \cdot (z-z_0)} \right)$ <p>Où :</p> <p>T_{iv} est la constante d'intégration K_{dv} : paramètre de mise à l'échelle de l'action dérivée K_{cv} : gain statique du correcteur Z_0 : pôle d'adoucissement de l'action dérivée</p>	$Ci(z) = K_{ci} \cdot \left(1 + \frac{T_e}{T_{ii}} \cdot \frac{1}{z-1} \right)$ <p>Où :</p> <p>T_{ii} est la constante d'intégration K_{ci} : gain statique du correcteur</p>
--	---

- Exprimer la relation de récurrence liant $N_i(kT_e)$ à $\mathcal{E}_N(kT_e)$.**
- Exprimer la relation de récurrence liant $N_{PWM}(kT_e)$ à $\mathcal{E}_I(kT_e)$.**
- Justifier la présence du limiteur s'appliquant à la valeur de $Nic(z)$. Donner un exemple montrant l'intérêt de ce limiteur. Calculer la valeur limite de Nic , sachant que le courant moteur ne doit pas excéder 5,1 A.**
- Exprimer et calculer K_{ci} permettant d'harmoniser les formats numériques des nombres entiers $N_{PWM}(kT_e)$ (codé sur 8 bits) et $\mathcal{E}_I(kT_e)$.**
- Sachant que $Nc(z)$ est toujours positif, donner une relation liant $Ncg(z)$ du moteur gauche à $Ncd(z)$ du moteur droit en fonction des contenus des mots MODBUS 0x0011 et 0x0012. On veillera à mettre à l'échelle les consignes en s'aidant d'un facteur multiplicateur. Répertoire la destination des mots $N_{PWM}(kT_e)$ correspondant à chaque axe en fonction des sens de rotation imposés par la relation liant les mots MODBUS 0x0011 et 0x0012.**

C.2 ETUDE DE LA STRUCTURE LOGICIELLE DE LA CARTE

Le calculateur embarqué sur la carte est un ATMEGA 88 de la société Atmel.

C.2.1 MESURE DE LA VITESSE

Nous rappelons que l'acquisition de la vitesse se fait par l'intermédiaire d'un codeur incrémental, 1024 points par tour, sur l'axe moteur qui fournit deux signaux carrés en quadrature. Nous allons utiliser les deux signaux pour élaborer deux vitesses qui seront ensuite comparées pour éliminer l'aléa de mesure engendré par les vibrations du robot autour de sa position d'équilibre sur le front de détection du codeur.

Q30. On met en œuvre deux compteurs d'impulsions. Préciser la stratégie d'élaboration de la vitesse (en radian par seconde), celle-ci devant être stockée dans un mot de 16 bits et offrir une résolution de 0,1 radian par seconde (nota: la vitesse angulaire maximale du moteur est de 314 rad/s).

- a) *Deux stratégies d'acquisition sont possibles (scrutation ou interruption), justifier le fait que l'acquisition par interruption est la mieux adaptée dans le cas de ce système, en précisant les critères.*
- b) *En utilisant l'extrait de documentation du microcontrôleur, déterminer les valeurs d'initialisation des registres pour que les interruptions puissent être générées (il est rappelé que les deux codeurs sont branchés respectivement sur les lignes 27 (PCINT12) pour AXE1-XA, 28 (PCINT13) pour AXE1-XB, 15 (PCINT4) pour AXE2-XB, 16 (PCINT3) pour AXE2-XA)*
- c) *Produire sous forma algorithmique la routine de gestion des interruptions.*
- d) *Fournir l'algorithme du programme d'élaboration de la vitesse.*

C.2.2 MESURE DU COURANT

Les mesures de courant I_{AXE1} et I_{AXE2} des deux ponts sont câblées respectivement sur les entrées 23 (ADC 0) et 24 (ADC1) du microcontrôleur.

Les données "mesure de courant à fournir" au logiciel de correction doivent être codées sur 16 bits avec une précision de 10^{-2} A.

Q31.

- a) *Préciser le canal utilisé pour effectuer cette conversion à l'aide du schéma de la carte.*
- b) *En utilisant l'extrait de documentation du microcontrôleur, identifier les registres et proposer les valeurs d'initialisation à y placer pour effectuer la conversion.*
- c) *Déterminer l'instant de fin de conversion pour vérifier que le bit d'état est bien positionné.*
- d) *Fournir l'algorithme de traitement de cette acquisition.*

C.2.3 IMPLÉMENTATION DES CORRECTEURS

Q32. Suite à la présentation de l'asservissement développé précédemment

- a) *Définir les structures de données nécessaires à l'implémentation de chaque correcteur.*
- b) *Ecrire l'algorithme permettant d'implémenter sous forme de procédure chacun de ces correcteurs.*
- c) *Ecrire l'algorithme général permettant de piloter un axe moteur à partir de la consigne Nc.*
- d) *Vérifier par une méthode de traçage le bon fonctionnement de ces algorithmes.*

C.3 EXPLOITATION PÉDAGOGIQUE

C.3.1 PRÉSENTATION DE LA COMMANDE MLI EN STI GÉNIE ÉLECTRONIQUE

Il s'agit de proposer l'étude du principe de la commande MLI et la structure d'un pont en H pour la commande en vitesse d'un moteur à courant continu dans les deux sens de rotation. On cherche également à montrer l'utilité de la mesure du courant moteur afin de détecter et d'indiquer une surcharge en couple.

Q33. Définition des séquences et documents pédagogiques

- a) *Proposer une structure de séquence pédagogique permettant d'aborder les concepts mis en œuvre sur la carte variateur pour le pilotage de la vitesse moteur au niveau choisi (on ne s'intéresse pas à l'aspect asservissement).*
- b) *Produire un schéma fonctionnel de degré 1 de la carte variateur dans la configuration de l'acteur locomotion.*

Q34. Développement d'une activité pratique

- a) *Proposer une mise en œuvre matérielle destinée à mettre en évidence le principe de la commande MLI pour la commande en vitesse d'une MCC. Mettre en évidence les objectifs des mesures envisagées.*
- b) *Expliciter le positionnement de cette activité dans la séquence proposée précédemment ainsi que les objectifs pédagogiques visés.*

Attentes du jury :

On attend une production maximale de 4 pages format A4.

Critères d'évaluation du candidat :

- L'exactitude technique des documents présentés et le respect du système proposé,
- La pertinence pédagogique des objectifs définis,
- Le respect des exigences du référentiel,
- La cohérence et la pertinence pédagogique des représentations utilisées au niveau choisi,
- La qualité des descriptions et des explications proposées (exactitude technique, précision et concision, expression écrite, adéquation par rapport au contexte).

C.3.2 IMPLÉMENTATION ET TEST D'UNE CARTE DE VARIATION DE VITESSE EN STS SYSTÈMES ELECTRONIQUES

On envisage d'exploiter pédagogiquement le système et le sous-système associés à la carte de variation de vitesse dans une section de STS Systèmes Electroniques en développant le champ d'application « **Automatique et robotique** ». On cherche à amener les étudiants à tester et valider les performances des éléments logiciels et matériels de l'électronique de contrôle du module, à partir du bus de communication, jusqu'aux performances dynamiques de l'asservissement.

On cherche à développer les compétences terminales suivantes :

- T2 : établir les procédures de test sur une maquette.
- M3 : détecter le (ou les) composants ou le (ou les) constituants défectueux.

Q35. Définition de la configuration matérielle

- a) *Définir et proposer une configuration pédagogique matérielle permettant de tester et valider les performances de la carte.*
- b) *Définir la méthodologie envisagée pour réaliser les tests de l'asservissement de vitesse.*

Q36. Production des documents pédagogiques

- a) *Proposer un document pédagogique précisant aux étudiants les étapes de tests et les critères de validation du fonctionnement de la carte.*
- b) *Elaborer une stratégie pédagogique permettant d'amener les étudiants à proposer une configuration industrielle de test de la carte.*

Critères d'évaluation du candidat :

- La pertinence des solutions matérielles envisagées,
- La cohérence des stratégies pédagogiques développées,
- Le respect des exigences du référentiel,
- La qualité des descriptions et des explications proposées (exactitude technique, précision et concision, expression écrite, adéquation par rapport au contexte).

Document réponse DR 1 (Q11.d)

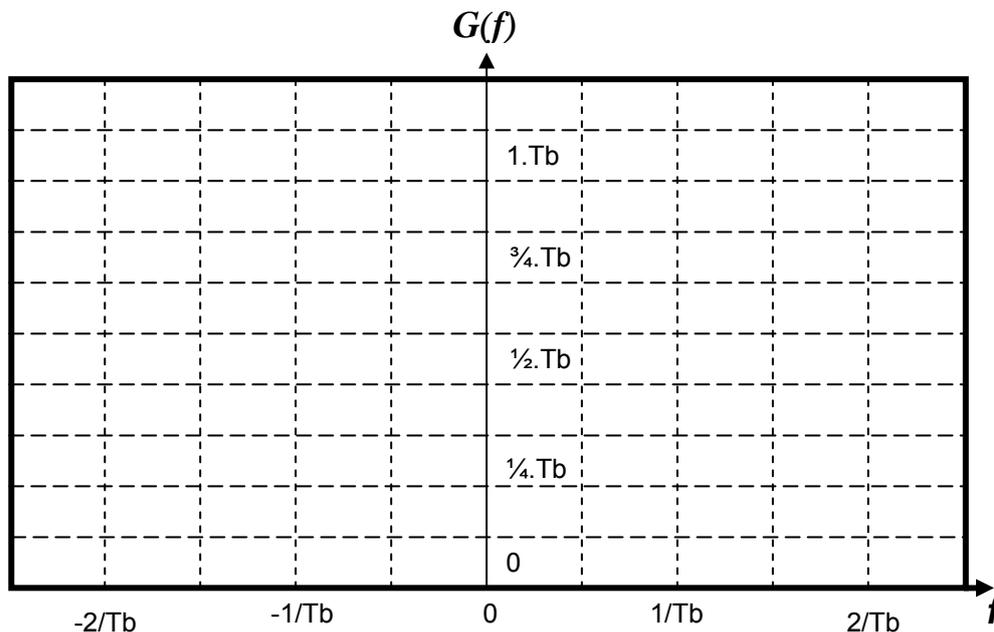
Nom du signal	Rôle du signal	Direction entrée/sortie	Rôle du bus associé
DATMSG			
DATACLK			
SLE			
SCK			
SDATA			

NB : On se place du point de vue du modulateur/démodulateur HF intégré.

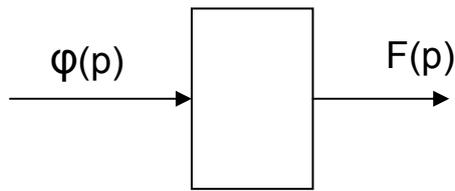
Document réponse DR 2 (Q13.c)

Contraintes temporelles	$P1(\tau) = \Phi_{mm}(\tau)$
$ \tau > T_b$	
$t \in [t_0, t_0 + \tau]$ avec $0 < \tau < T_b$	
$t \in [t_0 + \tau, t_0 + T_b]$ avec $0 < \tau < T_b$	

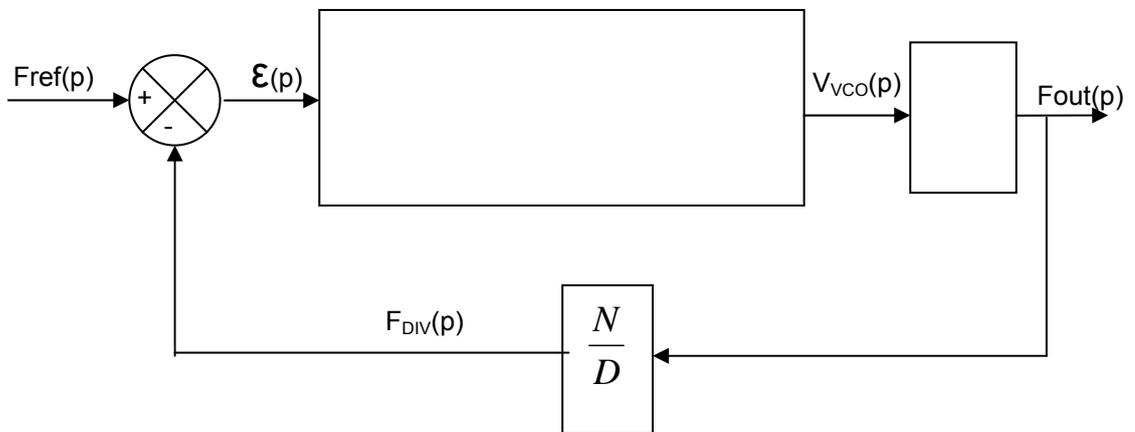
Document réponse DR 3 (Q13.e)



Document réponse DR 4 (Q17.b)



Document réponse DR 5 (Q17.c)



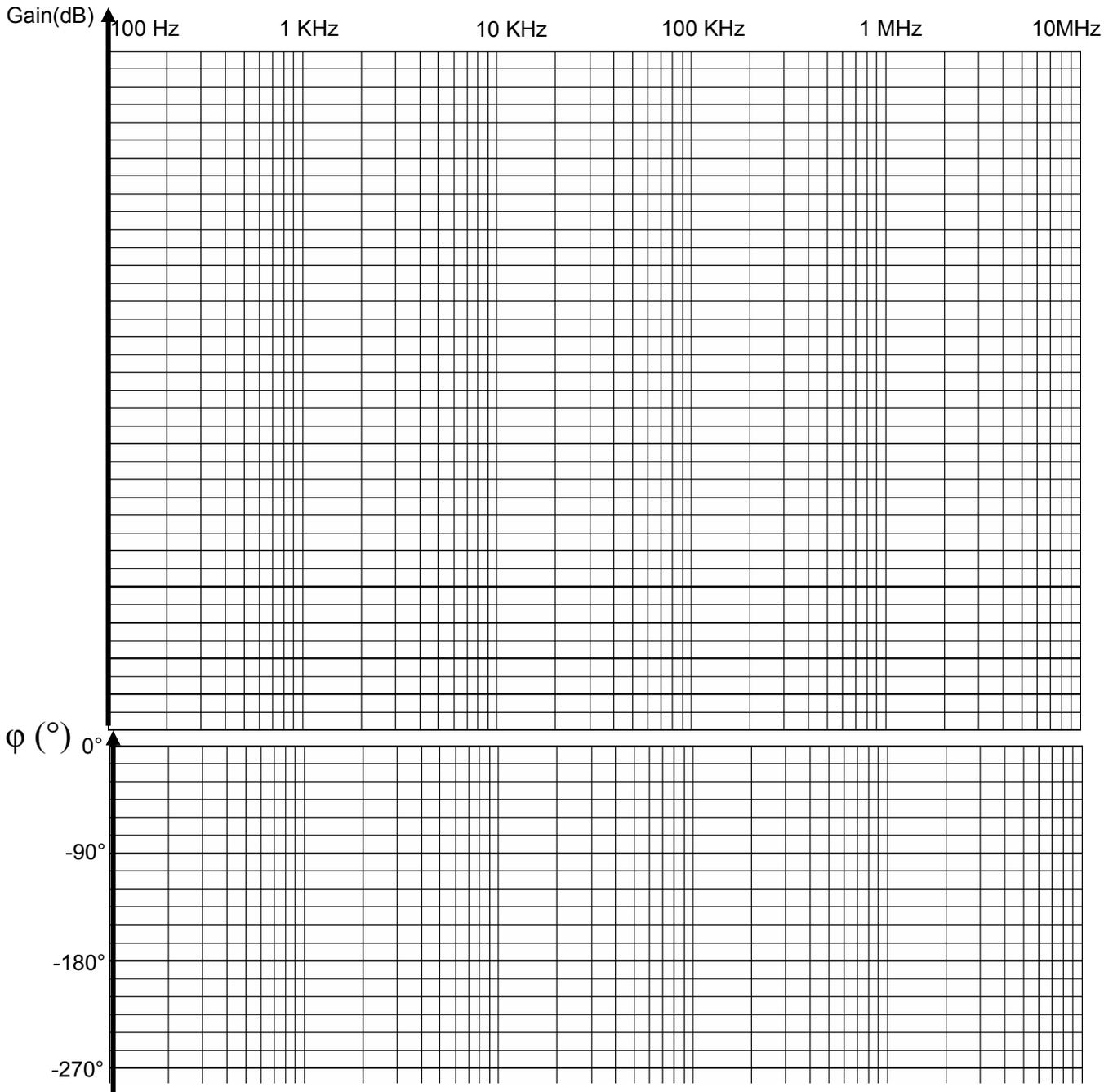
Document réponse DR 7 (Q25.e)

Indiquer entre parenthèse pour chaque registre à piloter le nom et la valeur du registre μC inutilisé.

Contenu du mot MODBUS 0x0011	Contenu du mot MODBUS 0x0012	Nom du Registre à piloter pour le moteur Gauche (Nom et valeur du Registre inutilisé)	Nom du Registre à piloter pour le moteur Droit (Nom et valeur du Registre inutilisé)	Etat logique du signal BRAKES
0x0400	0x0000			
0x0400	0x0200			
0x0000	0x0000			
0xFC00	0x0100			
0xFC00	0xFF00			

Document réponse DR 6 (Q17.d)

Fixer la graduation de l'axe de gain



ANNEXES

SOMMAIRE

A. Formulaire.....	49
A.1 – Définition de la fonction d’auto corrélation	49
A.2 – Définition de la transformée de Fourier	49
A.3 – Rappels concernant la définition de la fonction Triangle.....	49
B. Informations sur le protocole MODBUS et son implémentation	50
B.1 – Diagrammes d’états des maîtres ou esclaves MODBUS.....	50
B.2 – Calcul du CRC.....	50
C. Documents constructeurs.....	51
C.1 - Moteur à courant continu PARVEX RS220F	51
C.2 - IPS5451	52
C.3 - IPS0151	52
C.4 - Document constructeur du CI AT86RF211	52
C.5 - Extraits de document constructeur du microcontrôleur ATmega88	52
D. Extraits de référentiel de première et terminale STI Génie Electronique.....	53
D.2 – Extraits de référentiel du STS Systèmes Electroniques	55
D.3 – Extraits de référentiel du STS IRIS	61
E. Informations techniques générales.....	63
D.1 - Tables d’échanges MODBUS des différents modules.....	63

Note du rédacteur du rapport : pour éviter des problèmes de téléchargement, les extraits de documents constructeurs qui constituaient l’essentiel de cette annexe ont été supprimés (ils étaient sous forme d’images prenant un volume considérable). En revanche, les documents constructeurs en format *pdf* sont fournis, ainsi que les références aux parties utiles.

A. Formulaire

A.1 – Définition de la fonction d'auto corrélation

$$\phi_{xx}(\tau) = \frac{1}{T} \cdot \int_T x(t) \cdot x(t - \tau) \cdot dt$$

A.2 – Définition de la transformée de Fourier

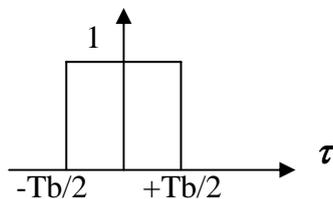
$$TF[x(t)] = X(f) = \int_{-\infty}^{+\infty} x(t) \cdot e^{-j2\pi \cdot f \cdot t} \cdot dt$$

A.3 – Rappels concernant la définition de la fonction Triangle $Tri_{Tb}(\tau)$

$$Tri_{Tb}(\tau) = \frac{1}{Tb} [Rect_{Tb/2}(\tau) * Rect_{Tb/2}(\tau)]$$

Où * représente le produit de convolution

et $[Rect_{Tb/2}(\tau)]$ est définis ci-dessous :



B. Informations sur le protocole MODBUS et son implémentation

La documentation complète du protocole MODBUS se trouve sur le site :
<http://www.modbus.org/specs.php>.

B.1 – Diagrammes d'états des maîtres ou esclaves MODBUS

Les extraits des documentations initialement dans cette partie ont été supprimés.
Elles figurent dans le fichier *Modbus_over_serial_line_v1.pdf* aux pages 9/44-11/44.

B.2 – Calcul du CRC

Les extraits des documentations initialement dans cette partie ont été supprimés.
Elles figurent dans le document *Modbus_over_serial_line_v1.pdf* aux pages 39/44-41/44.

C. Documents constructeurs

C.1 - Moteur à courant continu PARVEX RS220F

Le couple nominal du moteur est de 0,225 Nm

Extrait de documentation constructeur :

<i>Couple en rotation lente</i>	<i>0.225</i>	<i>N.m</i>	<i>M₀</i>
<i>Courant permanent rotation lente</i>	<i>4.1</i>	<i>A</i>	<i>I₀</i>
<i>Tension d'alimentation de définition</i>	<i>25.4</i>	<i>V</i>	<i>U</i>
<i>Vitesse de définition</i>	<i>3000</i>	<i>tr/mm</i>	<i>N</i>
<i>Tension maximale</i>	<i>40</i>	<i>V</i>	<i>U_{max}</i>
<i>Vitesse maximale</i>	<i>6600</i>	<i>tr/mm</i>	<i>N_{max}</i>
<i>Courant impulsionnel</i>	<i>12</i>	<i>A</i>	<i>I_{max}</i>
<i>Fem par 1000 tr/mm (25°C)</i>	<i>6</i>	<i>V</i>	<i>K_e</i>
<i>Coefficient de couple électromagnétique</i>	<i>0.057</i>	<i>N.m/A</i>	<i>K_t</i>
<i>Couple de frottement sec</i>	<i>1.2</i>	<i>N.cm</i>	<i>T_f</i>
<i>Coefficient de viscosité par 1000tr/mm</i>	<i>0.14</i>	<i>N.cm</i>	<i>K_d</i>
<i>Résistance du bobinage (25°C)</i>	<i>1.12</i>	<i>Ω</i>	<i>R_b</i>
<i>Inductance du bobinage</i>	<i>0.65</i>	<i>mH</i>	<i>L</i>
<i>Inertie rotor</i>	<i>0.0000195</i>	<i>kg.m²</i>	<i>J</i>
<i>Constante de temps thermique</i>	<i>6.9</i>	<i>min</i>	<i>T_{th}</i>
<i>Masse moteur</i>	<i>0.7</i>	<i>kg</i>	<i>M</i>

C.2 - IPS5451

La page 1 du document *IPS5451.pdf* constitue l'annexe afférente à ce composant.

C.3 - IPS0151

La page 1 du document *IPS0151.pdf* constitue l'annexe afférente à ce composant.

C.4 - Document constructeur du CI AT86RF211

La documentation technique de ce circuit est dans le fichier *AT86RF211.pdf*. On la trouve sur l'URL suivant :

http://www.atmel.com/dyn/resources/prod_documents/doc1942.pdf

C.5 - Extraits de document constructeur du microcontrôleur ATmega88

La documentation technique du circuit ATmega88 est dans le fichier *ATmega48.pdf*. On la trouve également sur l'URL :

http://www.atmel.com/dyn/resources/prod_documents/2545S.pdf

D. Extraits de référentiel de première et terminale STI Génie Electronique

Le candidat doit être capable de :

- définir le système technique ou le système mixte (A)
- analyser l'organisation fonctionnelle d'un objet technique (B)
- analyser le processus de fonctionnement du système (C)
- identification à une fonction (D)
- analyser l'organisation structurelle d'une fonction (E)
- rechercher l'adéquation entre les structures et les segments de programmes (F)
- produire une réorganisation structurelle (G)
- produire un dispositif de mesurages (H)
- produire une maquette (I)

CONVERSION DE GRANDEURS ÉLECTRIQUES EN GRANDEURS PHYSIQUES

L'ensemble des connaissances est réparti sur les années de premières et terminale et les diverses parties peuvent par l'approche fonctionnelle être abordées dès la première. Il convient que les élèves étudient des systèmes et objets techniques permettant une approche indépendante de la technologie utilisée et qu'ensuite ils procèdent à une étude structurelle spécifique.

Le professeur fait développer une méthodologie fondée sur une approche fonctionnelle offrant ainsi à l'élève un champ technologique vaste varié et évolutif dans le temps.

FONCTION	ÉLÉMENTS DE STRUCTURE MIS EN ŒUVRE	Durée	Commentaires
Conversion électrique / mécanique	Moteur à courants continu à aimants permanents Moteur pas à pas	20 h	On se limitera à une utilisation des données du constructeur. La commande des moteurs pas à pas sera réalisée par des composants spécialisés.
Contrôle de l'énergie	Redresseur, gradateur, hacheur série à transistors, régulateur	30 h	L'étude portera essentiellement sur la justification du dimensionnement des composants Au travers de relevés expérimentaux.

FONCTION	ÉLÉMENTS DE STRUCTURE MIS EN ŒUVRE	Durée	Commentaires
Amplification	amplificateur de tension, amplificateur de courant, amplificateur de puissance: - aspects technologiques - problèmes thermiques - rendement	60 h	<p>Les principes physiques mis en oeuvre dans l'amplification étant abordés en physique appliquée l'activité consistera surtout à travers les documents constructeurs à mettre en évidence les caractéristiques et limites d'emploi des composants. En ce qui concerne les amplificateurs de tension et de courant les composants intégrés seront de préférence utilisés. L'étude des différentes structures est abordée dans le contexte fonctionnel.</p> <p>L'étude de l'amplificateur de puissance sera l'occasion de mettre en oeuvre les solutions de dissipation thermique des composants de puissance. La notion de rendement étudiée en physique sera abordée en relation avec le contexte fonctionnel.</p>

Traitement programmé de l'information - Organisation matérielle : architecture fonctionnelle d'un dispositif utilisant la logique programmée - séquencement, - mémorisation, - échanges des informations entre les entités fonctionnelles. - Organisation logicielle :	structures de bus, interfaces. structure algorithmiques de base: - linéaire, - alternative, - itérative, - sous-programmes, - éléments de programmation permettant la réalisation de fonctions relatives à l'objet technique étudié.	50 h	Lors de l'analyse du fonctionnement d'un objet technique programme, il est important de montrer comment l'association des structures et du programme réalisé la (ou les) fonction requise par l'objet technique. Après étude des algorithmes et analyse du programme en langage assembleur ou en langage évolué, les élèves seront amenés à modifier certains segments de programme en relation avec l'évolution des contraintes fonctionnelles. A cet effet les élèves recevront les informations et documents strictement nécessaires. Les langages de programmation ne pourront faire l'objet d'une étude intrinsèque. La maîtrise des segments de programme relatifs à l'initialisation des périphériques ne pourra être exigée des élèves.
---	--	-------------	---

D.2 – Extraits de référentiel du STS Systèmes Electroniques

Capacité : INSTALLER et MAINTENIR un équipement ou un produit en fonctionnement

BTS systèmes électroniques

Compétence terminale M3 : détecter le (ou les) composants ou le (ou les) constituants défectueux

Savoir-faire Être capable de ..	Conditions de réalisation		Niveau de performance (on se limite à)
	On donne	On demande	
<p>Le candidat doit être capable de détecter le ou les composants défectueux. Cela implique qu'il doit :</p> <ul style="list-style-type: none"> analyser les résultats du rapport de test comparer le fonctionnement réel et le fonctionnement attendu, diagnostiquer une défaillance en vérifiant le fonctionnement des matériels et logiciels, détecter le ou les éléments défectueux en suivant le signal (ou l'information), établir un rapport d'intervention qui précise le coût et durée de celle-ci, préciser les mesures à réaliser après l'intervention qui permettront de vérifier que le fonctionnement sera correct après remplacement du composant défectueux. 	<p>Le candidat dispose :</p> <ul style="list-style-type: none"> du rapport de test sur le produit en fonctionnement, des dossiers d'installation de l'équipement, du plan architectural du site et du schéma de câblage du site (si nécessaire), de la configuration initiale du système sur le quel il intervient, des outils de tests et câblage, des spécifications du cahier des charges du sous-ensemble à remettre en conformité et de son cahier technique, des résultats des analyses fonctionnelles et structurelles de l'objet technique, d'un guide (complet ou partiel) décrivant les processus de test et de maintenance, des règlements d'hygiène et sécurité, des normes et consignes de sécurité. <p>Le candidat dispose également de l'équipement ou du produit défectueux ainsi que de l'ensemble des appareils de mesures spécifiques permettant d'effectuer les tests nécessaires.</p>	<p>Le candidat consigne les tests effectués dans un court rapport qui identifie le ou les composants défectueux à remplacer. Ce travail s'appuie sur le rapport de test de l'équipement ou du produit. Ce rapport d'investigation contient :</p> <ul style="list-style-type: none"> le diagnostic du dysfonctionnement établi en comparant le fonctionnement réel et le fonctionnement attendu, l'organisation de son poste de mesure, les tests et les essais effectués en précisant le type de mesures, les appareils utilisés, les conditions de la mesure, l'interprétation des tests effectués, le rapport d'intervention qui précise le coût et la durée celle-ci, la conclusion qui préconise le remplacement du composant défectueux. <p>Un résumé du rapport de tests qui doit conclure au remplacement du composant défectueux est écrit en Français et en Anglais.</p>	<p>Le candidat connaît le produit ou l'équipement avant la séance. Dans ces conditions, l'ensemble des activités demandées (y compris la rédaction du rapport) doit être réalisable en quatre heures.</p> <p>A partir de son rapport de test et après une courte période d'adaptation, le candidat doit être en mesure de reprendre à tout moment des mesures complémentaires demandées par le jury.</p>

Capacité : TESTER puis VALIDER un équipement ou un produit électronique

BTS systèmes électroniques

Compétence terminale T2 : établir les procédures de tests sur une maquette

Savoir-faire Être capable de ..	Conditions de réalisation		Niveau de performance (on se limite à)
	On donne	On demande	
<p>Le candidat doit établir les procédures de tests d'une maquette ou d'une fonction d'un produit. Cela implique qu'il doit être capable de :</p> <ul style="list-style-type: none"> déduire, à partir des spécifications du cahier des charges, le comportement d'une fonction, proposer des méthodes de mesure et/ou de test après analyse des différents schémas (fonctionnels, structurels,...), établir un plan d'organisation conduisant à la vérification des spécifications, en utilisant soit une simulation, soit des essais et des mesures sur la maquette en fonctionnement, déterminer les tests pertinents dont les résultats détermineront la validité de la maquette, choisir et placer les points de test permanents, définir une procédure de mise en conformité de la maquette et valider la procédure. 	<p>Le candidat dispose :</p> <ul style="list-style-type: none"> des dossiers d'études fonctionnelle et structurelle, du logiciel dans le cas d'un objet technique réalisé en partie en technologie programmée, des spécifications techniques extraites du cahier des charges. <p>Le candidat dispose également :</p> <ul style="list-style-type: none"> d'un poste d'expérimentation qui permet de générer les signaux d'entrée appliqués aux fonctions réalisées et mesurer les signaux électriques représentatifs, d'un poste de simulation informatique équipé d'un logiciel utilisant le modèle électrique des composants, d'un poste de développement pour le langage étudié, d'une maquette opérationnelle ou un élément de produit en état de bon fonctionnement. 	<p>Le candidat doit rédiger un dossier qui définit les procédures de tests qui permettent de valider le bon fonctionnement de la maquette. Ce dossier comprend :</p> <ul style="list-style-type: none"> la description du comportement mesurable de la fonction ou de l'ensemble de fonctions à partir des spécifications du cahier des charges, les stimuli à appliquer à la maquette et le mode de création de ces stimuli, éventuellement, les logiciels ou programmes à implanter dans les composants programmables de la maquette et les modes de transfert de ces programmes, le choix des appareils de mesures appropriés, les relevés des grandeurs caractéristiques de la maquette, l'organisation des procédures de tests, une synthèse des procédures de tests qui permet de conclure à la conformité ou la non-conformité de la maquette, éventuellement une proposition d'amélioration des procédures de test. 	

S04 : transmission et transport de l'information

Savoirs	On se limite à :
<ul style="list-style-type: none"> • transmission de signaux analogiques et numériques en bande de base. 	<ul style="list-style-type: none"> • déterminer la relation entre la grandeur d'entrée et la grandeur de sortie des modulations de type AM, FM, PM et FSK. Les autres types de modulation seront abordés expérimentalement en fonction du projet et des systèmes étudiés, • valider le type de modulation au regard du cahier des charges de la transmission en terme de bande passante, de débit ou de rapport signal sur bruit. Il est souhaitable que cette validation s'effectue expérimentalement.
<ul style="list-style-type: none"> • transmission de signaux analogiques et numériques en bande transposée (fréquence porteuse, ..) 	
<ul style="list-style-type: none"> • modulation et démodulation analogique (AM, FM, PM) et numériques (FSK, PSK....), • encombrement spectral. 	
<ul style="list-style-type: none"> • transmission numérique de l'information point à point sous forme série (RS232, RS 485,...) ou parallèle. 	<ul style="list-style-type: none"> • vérifier le type de connexion est conforme à la norme ou à la convention de connexion.
<ul style="list-style-type: none"> • architecture et organisation en couches des réseaux de transmissions et mode de gestion (protocoles) de ces réseaux (CAN, Ethernet...) bus de terrain...). 	<ul style="list-style-type: none"> • identifier le protocole utilisé par la liaison et vérifier que celui-ci répond au cahier des charges de l'équipement.

La colonne de gauche précise les exigences professionnelles liées aux domaines d'activités dans l'entreprise. La colonne du milieu indique les savoirs disciplinaires en liaison avec les contraintes développées. Les colonnes de droite déterminent les niveaux de compétence. Chacun de ces niveaux cumule les compétences des niveaux précédents.

- 1) **Niveau d'information** : "je sais de quoi je parle", est un niveau d'information, il correspond à l'appréhension d'une vue d'ensemble d'un sujet. Les problèmes sont abordés de manière globale.
- 2) **Niveau d'expression** : "je sais en parler", est un niveau de compréhension, il correspond à l'acquisition de moyens d'expression et de communication. Le technicien définit et utilise les termes des spécialistes du domaine.
- 3) **Niveau de maîtrise d'outils** : "je sais faire", est un niveau d'application, il correspond à la maîtrise de procédés et d'outils d'étude ou d'action. Le technicien sait utiliser, mettre en place des procédures en vue d'un résultat à atteindre.
- 4) **Niveau de la maîtrise méthodologique et technologique** : "je sais choisir", est un niveau de savoir et d'autonomie, avec une capacité d'analyse, de synthèse et d'évaluation, il correspond à la méthodologie de pose et de résolution de problèmes techniques. Le technicien maîtrise une démarche. Il est en mesure de choisir les équipements, d'encadrer une petite équipe afin de mener à terme un mini projet lié à une phase de cycle de vie du produit.

TELECOMMUNICATION, TELEPHONIE ET RESEAUX TELEPHONIQUES

Télécommunication, téléphonie et réseaux téléphoniques					
Champs technologiques	Connaissances associées aux champs technologiques	Niveau d'exigence			
		1	2	3	4
	<ul style="list-style-type: none"> Lecture des schémas fonctionnels et structurels des produits spécifiques (émetteurs et récepteurs radiofréquences analogiques et numériques) 			X	
	<ul style="list-style-type: none"> Organisation fonctionnelle des réseaux cellulaires 		X		
	<ul style="list-style-type: none"> Normes et recommandations ETSI, IEEE (CT0, CT2, DECT, GSM, ...) 		X		
	<ul style="list-style-type: none"> Modulations et démodulations analogiques (AM, FM, PM) et numériques (FSK, PSK, GMSK, QPSK, QAM, ...) 				X
	<ul style="list-style-type: none"> Traitements du signal analogique en radiofréquences (adaptation d'impédance, amplification, filtrage, production de signaux, transposition de fréquence ...) 				X
	<ul style="list-style-type: none"> Traitements spécifiques du signal numérique (multiplexages, compressions de données, ...) 		X		
	<ul style="list-style-type: none"> Boucles à verrouillage de phase 				X
	<ul style="list-style-type: none"> Etalement de spectre par sauts de fréquence et séquence pseudo-aléatoire 		X		
	<ul style="list-style-type: none"> Composants passifs et actifs du domaine radiofréquence 				X
	<ul style="list-style-type: none"> Technologies micro-strip, stripline, ... 		X		
	<ul style="list-style-type: none"> Instrumentation spécifique : analyseurs de spectre et de réseaux, générateurs RF, montés en réseau instrumental (GPIB, Ethernet) 				X
	<ul style="list-style-type: none"> Instruments spécialisés : testeurs de mobiles et de stations de base, de protocoles de communication, ... 		X		
	<ul style="list-style-type: none"> Caractéristiques d'un récepteur et d'un émetteur (y compris leurs mesures) 				X
	<ul style="list-style-type: none"> Caractéristiques des antennes 			X	
	<ul style="list-style-type: none"> Validations de maquettes, de prototypes, de composants RF, d'ASIC, ... 			X	
	<ul style="list-style-type: none"> Travaux de préparation aux agréments (CE ...) 			X	
	<ul style="list-style-type: none"> Contrôles de fabrication 			X	
	<ul style="list-style-type: none"> Identification de pannes en production et maintenance 			X	
	<ul style="list-style-type: none"> Installation, configuration et maintenance de stations relais 			X	
Télemaintenance	<ul style="list-style-type: none"> Mise à niveau matérielle du client 		X		
	<ul style="list-style-type: none"> Mise à niveau logicielle du client à distance 			X	

Informatique, télématique et bureautique

Champs technologiques	Connaissances associées aux champs technologiques	Niveau d'exigence			
		1	2	3	4
Acquisition d'images fixes ou animées	<ul style="list-style-type: none"> • Capteurs optiques et techniques de numérisation de l'image 			X	
Restitution d'images fixes ou animées	<ul style="list-style-type: none"> • Les systèmes de visualisation sur écran ou afficheurs. • Les systèmes d'impressions sur papier 			X	
Stockage de données	<ul style="list-style-type: none"> • Les systèmes de mémorisation électroniques, magnétiques et optiques 			X	
Traitement des informations (matériel et logiciel)	<ul style="list-style-type: none"> • Les systèmes de traitement de données et leurs systèmes d'exploitation 			X	
Interconnexions d'équipements	<ul style="list-style-type: none"> • Les connexions en mode parallèles ou série, les bus d'extension ou d'interconnexion 				X
Réseau téléphonique	<ul style="list-style-type: none"> • Structure du réseau téléphonique analogique ou numérique et des terminaux d'accès (fax, modems adaptateurs Numéris) 		X		
Réseaux informatiques de terrains, locaux & publics	<ul style="list-style-type: none"> • Compréhension de la structure en couche des réseaux de transmissions. • Spécificités et protocoles d'exploitation (TCP/IP) 			X	
Interconnexion des réseaux	<ul style="list-style-type: none"> • Gestion et mise en place des équipements d'interconnexion (modem, routeurs et adaptateurs divers) 			X	
Systèmes de câblage des réseaux	<ul style="list-style-type: none"> • Éléments passifs et actifs d'un réseau informatique 			X	
Systèmes de liaisons sans fils	<ul style="list-style-type: none"> • Dispositifs de connexion par liaison infrarouge ou ondes électromagnétiques 				X
Tests et mesures	<ul style="list-style-type: none"> • Utilisation d'équipements de test et de mesure spécialisés pour les champs ci-dessus 				X

AUTOMATIQUE ET ROBOTIQUE

Dans le cadre de la conception, réalisation, installation et mise en service des systèmes industriels pluri technologiques, le technicien supérieur électronicien prend en charge le système de mesures et d'échanges des informations entre capteurs, unité de traitement et actionneurs.

Automatique et robotique					
Champs technologiques	Connaissances associées aux champs technologiques	Niveau d'exigence			
		1	2	3	4
Environnement spécifique au milieu industriel	<ul style="list-style-type: none"> • Contrainte en température • Contraintes aux vibrations • Contraintes aux perturbations 			X	
Sécurité des biens et des personnes	<ul style="list-style-type: none"> • Disjoncteur • Régime de neutre • Habilitation électrique • Production d'une consignation et mise en service d'une installation 		X		
Interconnexions d'équipements	<ul style="list-style-type: none"> • Lecture d'un schéma d'interconnexion entre interfaces • CAO et schématic 				X
Réseaux industriels	<ul style="list-style-type: none"> • FIP, C.A.A.N. • Réseau de terrain • ASI, Ethernet 			X	
Capteurs de technologie mécanique	Choix et utilisation des capteurs <ul style="list-style-type: none"> • Capteurs de position • Capteur de température • Capteur de pression 			X	

Automatique et robotique					
Champs technologiques	Connaissances associées aux champs technologiques	Niveau d'exigence			
		1	2	3	4
Capteurs de technologie électronique	Choix et utilisation des capteurs <ul style="list-style-type: none"> • Capteur magnétique • Capteur capacitif • Capteur optoélectronique • Capteur résistif, piézo-électrique 				X
Pré actionneurs électromécaniques	<ul style="list-style-type: none"> • Distributeurs pneumatique et hydraulique • Commutateur à relais 		X		
Variateur de vitesse	• Electronique de puissance		X		
	• Electronique de commande			X	
	• Electronique de contrôle				X
Actionneurs	• Moteur électrique		X		
	• Moteur hydraulique				
	• Moteur pneumatique				
Automate programmable industriel	• Constitution			X	
	• Programmation		X		
	• Communication			X	
Mécanique	<ul style="list-style-type: none"> • Statique • Dynamique • Cinématique 		X		
Mesurages	<ul style="list-style-type: none"> • Métrologie • Assurance qualité • Appareils e mesures 				X
Aspect économique	<ul style="list-style-type: none"> • Achats • Commercial • Assistance qualité • Prix de revient et retour sur investissement 		X		

D.3 – Extraits de référentiel du STS IRIS

Unité U.4 – Étude d'un système informatisé

CAPACITÉ	Compétence terminale
C3 : CONCEVOIR	C3.1 : Analyser un dossier de spécification

Ressources mises à disposition		
<p>Expression du besoin relatif à un projet logiciel. Documents techniques liés au projet. Tout ou partie des documents suivants :</p> <ul style="list-style-type: none"> - cahier des charges sous forme partielle ou complète, - dossier de spécification technique, - dossier de spécification logicielle. <p>Moyens informatiques requis.</p>		
Compétences composantes	Critères d'évaluation	Niveaux de performance
<ul style="list-style-type: none"> - Rechercher les liens entre besoin exprimé dans le cahier des charges et les spécifications. - Caractériser les fonctionnalités souhaitées. - Corréler les informations issues des différents diagrammes d'un dossier de spécification. - Compléter un diagramme de collaboration ou de séquence à partir des données complètes d'autres diagrammes. 	<ul style="list-style-type: none"> - Repérage des liens entre les spécifications fournies et le besoin. - Critères retenus pour caractériser les fonctionnalités. - Identification des messages corrélant diagramme de séquence et de collaboration, ou diagramme de classe et de collaboration (ou de séquence). - État du diagramme complété (<i>La demande ne doit porter que sur quelques messages</i>). 	<ul style="list-style-type: none"> - Les liens principaux sont identifiés. Les objectifs décrits dans le cahier des charges sont identifiés. Les cas d'utilisation principaux sont repérés. - Formulation claire et sans contre-sens des caractéristiques. <i>Les fonctionnalités peuvent par exemple être caractérisées en terme de qualité, coûts, délais, performances, sécurité, disponibilité, exploitation, maintenance.</i> - Identification des messages sans erreurs. Les corrélations doivent être présentées de manière non ambiguë. - Aucune erreur dans le diagramme complété. Tolérance sur des erreurs concernant les paramètres des messages.

Savoirs et savoir-faire associés :

Développement logiciel :

Analyse globale.

Organisation des fichiers dans un projet.

Modélisation.

Listes des acteurs.

Cas d'utilisation.

Diagramme(s) de séquences, de collaboration, de classes, d'états.

CAPACITÉ	Compétence terminale
C3 : CONCEVOIR	C3.2 : Définir l'architecture globale d'un prototype ou d'un système

Ressources mises à disposition		
Documents techniques liés au projet. Dossier de spécification logicielle. Moyens informatiques requis.		
Compétences composantes	Critères d'évaluation	Niveaux de performance
<ul style="list-style-type: none"> - Interpréter les spécifications logicielles d'une application. - Rechercher et lister les critères de choix au sein des dossiers fournis. - Justifier et adapter les moyens matériels et logiciels à mettre en œuvre. - Rédiger une partie du dossier de conception préliminaire. 	<ul style="list-style-type: none"> - Qualité de la transcription des spécifications en pseudo-code ou sous forme d'algorithme. - Compréhension des critères de choix. - Définition du rôle de chacun des moyens matériels et logiciels. - Qualité du document fourni qui doit faire clairement apparaître les choix matériels et logiciels. 	<ul style="list-style-type: none"> - Les algorithmes sont cohérents avec la spécification. Le pseudo-code est compréhensible. - Les critères exposés montrent que les dossiers fournis ont été lus et compris. - Les fonctions principales de chacun des moyens sont clairement décrites. - L'existence et la qualité du document. <i>(Le niveau d'exigence doit rester limité).</i>

Savoirs et savoir-faire associés :

Développement logiciel :

Analyse globale.

Modélisation.

Listes des acteurs.

Cas d'utilisation.

Diagrammes de séquences, de collaboration, de classes, d'états.

Algorithmique.

E. Informations techniques générales

D.1 - Tables d'échanges MODBUS des différents modules

MODULE commande 2 axes :

Esclave modbus 1 = Carte de commande locomotion robot

Esclave modbus 2 = Carte de commande du bras articulé

Esclave modbus 8 = Carte de commande du module mission pince

Esclave modbus 9 = Carte de commande du module mission caméra

Adresse du mot modbus	Contenu	Type d'accès
0x0001	Vitesse actuelle Axe 1 (8bits de poids faible) Position actuelle Axe 1 (8bits de poids fort)	Lecture
0x0002	Vitesse actuelle Axe 2 (8bits de poids faible) Position actuelle Axe 2 (8bits de poids fort)	Lecture
0x0003	Etat des axes Bit 0 : 0 = Axe 1 pas de surcharge 1 = Axe1 surcharge Bit 1 : 0 = Axe 1 retour codeur incrémental OK 1 = Axe1 défaut retour codeur Bit 2 : 0 = Axe 2 pas de surcharge 1 = Axe1 surcharge Bit 3 : 0 = Axe 2 retour codeur incrémental OK 1 = Axe1 défaut retour codeur Bit 4 : 0 = Pas de défaut général 1 = Défaut carte VAR Bit 5 : 0 = freins activés (absence de courant) 1 = freins désactivé	Lecture
0x0010	Mode de fonctionnement : Bit 0 : 0 = Axe 1 non asservi en position 1 = Axe1 asservi en position Bit 1 :	Lecture / écriture

	<p>0 = Axe 1 non asservi en vitesse 1 = Axe 1 asservi en vitesse</p> <p>Bit 2 : 0 = Axe1 et Axe 2 indépendants 1 = Mode locomotion robot Vitesse Axe1 = Vitesse avance – Direction Vitesse Axe2 = Vitesse avance + Direction</p> <p>Bit 3 : 0 = Axe 2 non asservi en position 1 = Axe2 asservi en position</p> <p>Bit 4 : 0 = Axe 2 non asservi en vitesse 1 = Axe 2 asservi en vitesse</p> <p>Bit 5 : 0 = Freins désactivés 1 = Freins activés</p>																					
0x0011	<p style="text-align: center;">Consignes Axe1</p> <table border="1"> <thead> <tr> <th>Bit 0 de 0x0010</th> <th>Bit 1 de 0x0010</th> <th>Bit 2 de 0x0010</th> <th>Mot signé en CC2* (12 bits significatifs)</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>Non significatif</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>Consigne de position</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>Consigne de vitesse</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>Consigne vitesse avance robot</td> </tr> </tbody> </table>	Bit 0 de 0x0010	Bit 1 de 0x0010	Bit 2 de 0x0010	Mot signé en CC2* (12 bits significatifs)	0	0	0	Non significatif	1	0	0	Consigne de position	0	1	0	Consigne de vitesse	0	0	1	Consigne vitesse avance robot	Lecture / Ecriture
Bit 0 de 0x0010	Bit 1 de 0x0010	Bit 2 de 0x0010	Mot signé en CC2* (12 bits significatifs)																			
0	0	0	Non significatif																			
1	0	0	Consigne de position																			
0	1	0	Consigne de vitesse																			
0	0	1	Consigne vitesse avance robot																			
0x0012	<p style="text-align: center;">Consignes Axe2</p> <table border="1"> <thead> <tr> <th>Bit 3 de 0x0010</th> <th>Bit 4 de 0x0010</th> <th>Bit 2 de 0x0010</th> <th>Mot signé en CC2* (12 bits significatifs)</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>Non significatif</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>Consigne de position</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>Consigne de vitesse</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>Consigne direction robot</td> </tr> </tbody> </table>	Bit 3 de 0x0010	Bit 4 de 0x0010	Bit 2 de 0x0010	Mot signé en CC2* (12 bits significatifs)	0	0	0	Non significatif	1	0	0	Consigne de position	0	1	0	Consigne de vitesse	0	0	1	Consigne direction robot	Lecture / Ecriture
Bit 3 de 0x0010	Bit 4 de 0x0010	Bit 2 de 0x0010	Mot signé en CC2* (12 bits significatifs)																			
0	0	0	Non significatif																			
1	0	0	Consigne de position																			
0	1	0	Consigne de vitesse																			
0	0	1	Consigne direction robot																			

* : CC2 = Code Complément à deux.

MODULE Télémétrie :

Adresse du mot	Contenu	Type d'accès
0x0001	Gain actuel de l'amplificateur du capteur à ultrason (16 bits)	Lecture
0x0002	Distance actuelle de l'obstacle (16 bits), résolution de la mesure : 1LSB = 1mm portée maximal en champ libre : 15m	Lecture
0x0003	Etat du télémètre : Bit 0 : Etat du pointeur laser : 1 : allumé 0 : éteint Bit 1 : état du télémètre : 0 : mesure possible 1 : Hors portée (pas d'écho) Bit 2 : réponse au test: 0 : OK 1 : défaut de test	Lecture
0x00010	Contrôle télémétrie Bit 0 : Etat du pointeur laser : 1 : allumé 0 : éteint Bit 1 : Calibration 0 : Calibration automatique 1 : Calibration sur consigne Bit 2 : Test : 0 : pas de test 1 : Lancement d'un test (calibration + mesure)	Ecriture / lecture
0x00011	Consigne calibration Valeur du gain de calibration (16 bits)	Ecriture / lecture

Module MISSION :

Esclave = 5 : Module Mission de tir et de neutralisation

Adresse du	Contenu	Type
------------	---------	------

mot		d'accès
0x0001	<p>Etat du canon</p> <p>Bit 0 : 0 = Canon non armé 1 = canon armé</p> <p>Bit 1 : 0 = TIR en attente 1 = TIR en cours</p> <p>Bit 2 : 0 = TIR non confirmé 1 = TIR confirmé</p> <p>Bit 3 : 0 = Pas de défaut de TIR 1 = Défaut de TIR (délais trop long)</p> <p>Bit 4 : 0 = Alimentation OK 1 = Erreur alimentation</p> <p>Bit 5 : 0 = Pas d'alarme 1 = Alarme : défaut système de TIR</p> <p>Bit 2 : réponse au test: 0 : OK 1 : défaut de test</p>	Lecture
0x0010	<p>Bit 0 : Demande de TIR 0 : pas de demande 1 : demande de TIR (attente de confirmation)</p> <p>Bit 1 : Validation demande de TIR : 0 : Pas de validation 1 : Ordre de TIR (il faut que le Bit 0 soit aussi à 1 pour avoir un TIR)</p> <p>Bit 2 : Demande de test système de TIR : 0 : pas de demande 1 : Lancement test</p> <p>Bit 2 : Commande projecteur : 0 : Eteint 1 : Allumé</p>	Lecture / Ecriture

MODBUS over Serial Line

Specification & Implementation guide

V1.0

Contents

1	Introduction	4
1.1	Scope of this document	4
1.2	Protocol overview.....	5
1.3	Conventions	5
1.4	Compliance	6
1.5	Glossary.....	6
2	MODBUS Data Link Layer	7
2.1	MODBUS Master / Slaves protocol principle.....	7
2.2	MODBUS Addressing rules.....	8
2.3	MODBUS frame description.....	8
2.4	Master / Slaves State Diagrams.....	9
2.5	The two serial Transmission Modes.....	12
2.6	Error Checking Methods	19
3	Physical Layer.....	20
3.1	Preamble.....	20
3.2	Data Signaling Rates	20
3.3	Electrical Interfaces.....	21
3.4	Multipoint System requirements.....	27
3.5	Mechanical Interfaces	29
3.6	Cables.....	32
3.7	Visual Diagnosis	32
4	Installation and Documentation	33
4.1	Installation.....	33
4.2	User Guide.....	33
5	Implementation Classes	34
6	Appendix.....	35
6.1	Appendix A - Management of Serial Line Diagnostic Counters.....	35
6.2	Appendix B - LRC/CRC Generation.....	38
6.3	Appendix E - References	44

Document modifications		
	Month-Year	Modifications
1.0	Nov 02	<p>Creation.</p> <p>This document comprises a description of Master / slave protocol and of the two different transmission modes (RTU, ASCII).</p> <p>The main features of the physical layer (RS485, RS232) and some recommendations are provided.</p> <p>Implementation classes are proposed to guide the implementation.</p>

1 Introduction

1.1 Scope of this document

The MODBUS standard defines an application layer messaging protocol, positioned at level 7 of the OSI model that provides "client/server" communications between devices connected on different types of buses or networks. It standardizes also a specific protocol on serial line to exchange MODBUS request between a master and one or several slaves.

The objective of this document is to present the MODBUS protocol over serial line, in order to be used by all system designers when they want to implement MODBUS protocol on their serial line products. Thus, this document will facilitate interoperability between devices using the MODBUS protocol.

This document comes in complement to the document called "MODBUS Application Protocol Specification".

In chapter 5 different implementation classes are defined for "MODBUS Serial Line". Specification of a class is the sum of requirements that a device must respect in order to belong to that class.

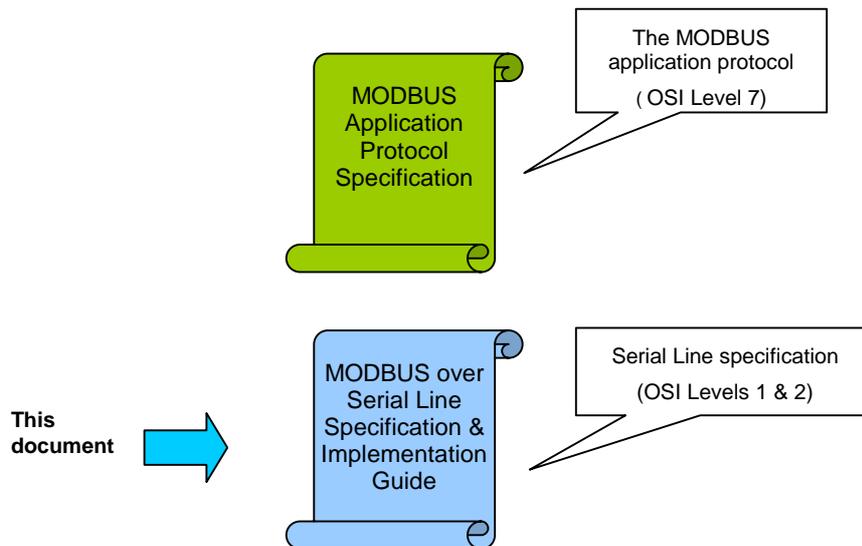


Figure 1: General overview of MODBUS documents

1.2 Protocol overview

This document describes the MODBUS over Serial Line protocol. **MODBUS Serial Line protocol is a Master-Slave protocol.** This protocol takes place at level 2 of the OSI model.

A master-slave type system has one node (the master node) that issues explicit commands to one of the "slave" nodes and processes responses. Slave nodes will not typically transmit data without a request from the master node, and do not communicate with other slaves.

At the physical level, MODBUS over Serial Line systems may use different physical interfaces (RS485, RS232). TIA/EIA-485 (RS485) Two-Wire interface is the most common. As an add-on option, RS485 Four-Wire interface may also be implemented. A TIA/EIA-232-E (RS232) serial interface may also be used as an interface, when only short point to point communication is required. (see chapter "Physical Layer")

The following figure gives a general representation of MODBUS serial communication stack compared to the 7 layers of the OSI model.

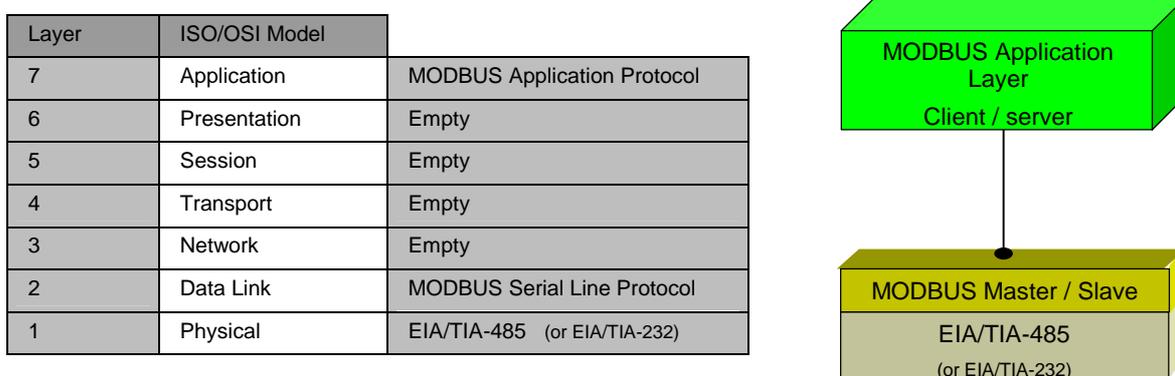


Figure 2: MODBUS Protocols and ISO/OSI Model

MODBUS application layer messaging protocol, positioned at level 7 of the OSI model, provides client/server communication between devices connected on buses or networks. On MODBUS serial line the client role is provided by the Master of the serial bus and the Slaves nodes act as servers.

1.3 Conventions

In this document, the following words are used to define the significance of each particular **requirement**.

- **"MUST" / "REQUIRED"**

All requirements containing the word **"MUST"** are mandatory. The word **MUST**, or the adjective **"REQUIRED"**, means that the item is an absolute requirement of the implementation. These words are underlined.

- **"SHOULD" / "RECOMMENDED"**

All recommendations containing the word **"SHOULD"**, or the adjective **"RECOMMENDED"**, are considered desired behavior. These recommendations should be used as a guideline when choosing between different options to implement functionality. There may be valid reasons in particular circumstances to ignore this item, but the full implications should be understood and the case carefully weighed before choosing a different course. These words are underlined.

- **"MAY" / "OPTIONAL"**

The word **"MAY"**, or the adjective **"OPTIONAL"**, means that this item is truly optional. One designer may choose to include the item because a particular marketplace requires it or because it enhances the product, for example; another designer may omit the same item.

1.4 Compliance

An implementation is **not in conformity** if it fails to satisfy one or more of the **MUST** requirements from its implementation class.

An implementation that satisfies all the MUST requirements and all the SHOULD recommendations is said to be "**unconditionally compliant**".

One that satisfies all the MUST requirements but not all the SHOULD recommendations is said to be "**conditionally compliant**".

1.5 Glossary

Definition of particular words, symbols, and abbreviations used in this document.

2W	The Two-Wire configuration defined in the "Electrical Interface" chapter, or one of its interfaces.
4W	The Four-Wire configuration defined in the "Electrical Interface" chapter, or one of its interfaces.
AUI	Attachment Unit Interface
Common	The Signal Common in EIA/TIA Standards. In a 2W-or 4W-RS485 MODBUS Network, Signal and optional Power Supply Common
DCE	a MODBUS Device, for example a programmable controller adapter, which implements an RS232 Data Circuit-terminating Equipment, also named Data Communication Equipment.
Device	or "MODBUS device" : see this definition.
Driver	Generator, or Transmitter.
DTE	a MODBUS Device, for example a programming panel or a PC, which implements an RS232 Data Terminal Equipment.
ITr	Physical bus Interface on Trunk side.
IDv	Physical bus Interface on Derivation (or tap or device drop) side.
LT	Line Termination.
MODBUS Device	a Device that implements MODBUS over Serial Line and respects this Technical Note.
RS232	EIA/ TIA -232 Standard.
RS485	EIA/ TIA -485 Standard.
RS485-MODBUS	A 2W-or 4W-Network in accordance with this Technical Note.
Transceiver	a Transmitter and a Receiver (or Driver and Receiver).

2 MODBUS Data Link Layer

2.1 MODBUS Master / Slaves protocol principle

The MODBUS Serial Line protocol is a Master-Slaves protocol. Only one master (at the same time) is connected to the bus, and one or several (247 maximum number) slaves nodes are also connected to the same serial bus. A MODBUS communication is always initiated by the master. The slave nodes will never transmit data without receiving a request from the master node. The slave nodes will never communicate with each other. The master node initiates only one MODBUS transaction at the same time.

The master node issues a MODBUS request to the slave nodes in two modes :

→ In **unicast mode**, the master addresses an individual slave. After receiving and processing the request, the slave returns a message (a 'reply') to the master .

In that mode, a MODBUS transaction consists of 2 messages : a request from the master, and a reply from the slave.

Each slave must have an unique address (from 1 to 247) so that it can be addressed independently from other nodes.

→ In **broadcast mode**, the master can send a request to all slaves.

No response is returned to broadcast requests sent by the master. The broadcast requests are necessarily writing commands. **All devices must accept the broadcast for writing function.** The address 0 is reserved to identify a broadcast exchange.

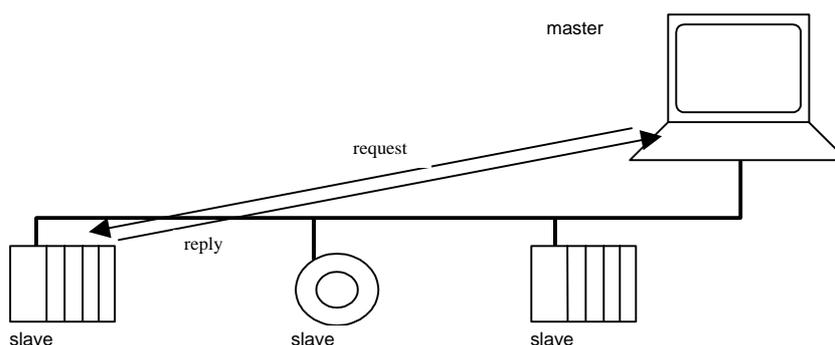


Figure 3: Unicast mode

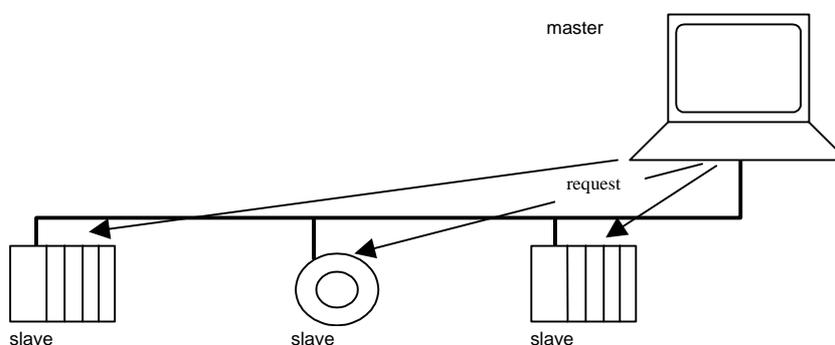


Figure 4: Broadcast mode

2.2 MODBUS Addressing rules

The MODBUS addressing space comprises 256 different addresses.

0	From 1 to 247	From 248 to 255
Broadcast address	Slave individual addresses	Reserved

The Address 0 is reserved as the broadcast address. All slave nodes must recognise the broadcast address.

The MODBUS Master node has no specific address, only the slave nodes must have an address. This address must be unique on a MODBUS serial bus.

2.3 MODBUS frame description

The MODBUS application protocol [1] defines a simple **Protocol Data Unit (PDU)** independent of the underlying communication layers:

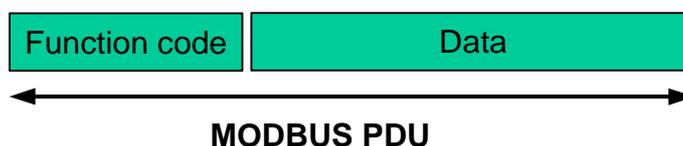


Figure 5: MODBUS Protocol Data Unit

The mapping of MODBUS protocol on a specific bus or network introduces some additional fields on the **Protocol Data Unit**. The client that initiates a MODBUS transaction builds the MODBUS PDU, and then adds fields in order to build the appropriate communication PDU.

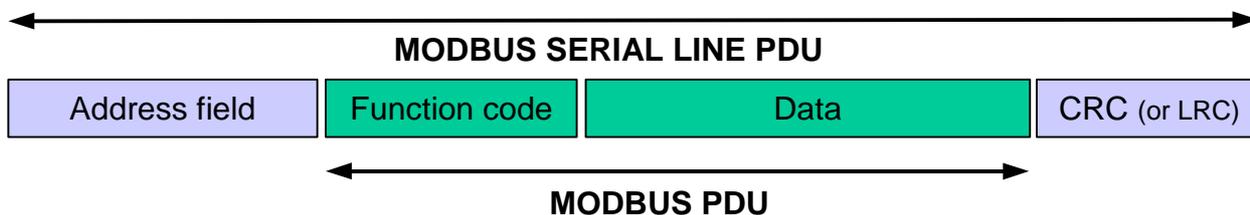


Figure 6: MODBUS frame over Serial Line

- On MODBUS Serial Line, the Address field only contains the slave address.

As described in the previous section the valid slave nodes addresses are in the range of 0 – 247 decimal. The individual slave devices are assigned addresses in the range of 1 – 247. A master addresses a slave by placing the slave address in the address field of the message. When the slave returns its response, it places its own address in the response address field to let the master know which slave is responding.

- The function code indicates to the server what kind of action to perform. The function code can be followed by a data field that contains request and response parameters.
- Error checking field is the result of a "Redundancy Checking" calculation that is performed on the message contents. Two kinds of calculation methods are used depending on the transmission mode that is being used (RTU or ASCII). (see 2.5 section, "The two serial Transmission Modes")

2.4 Master / Slaves State Diagrams

The MODBUS data link layer comprises two separate sub layers :

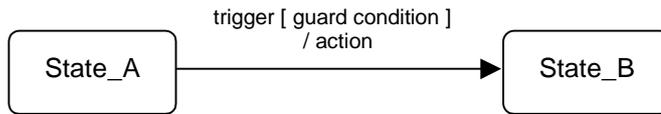
- The Master / slave protocol
- The transmission mode (RTU vs ASCII modes)

The following sections describes the state diagrams of a master and a slave that are independent of transmission modes used.

The RTU and ASCII transmission modes are specified in next chapters using two state diagrams. The reception and the sending of a frame are described.

Syntax of state diagram :

The following state diagrams are drawn in compliance with UML standard notations. The notation is briefly recalled below :



When a "trigger" event occurs in a system being in "State_A", system is going into "State_B", only if "guard condition" is true. An action "action" is then performed.

2.4.1 Master State diagram

The following drawing explains the Master behavior :

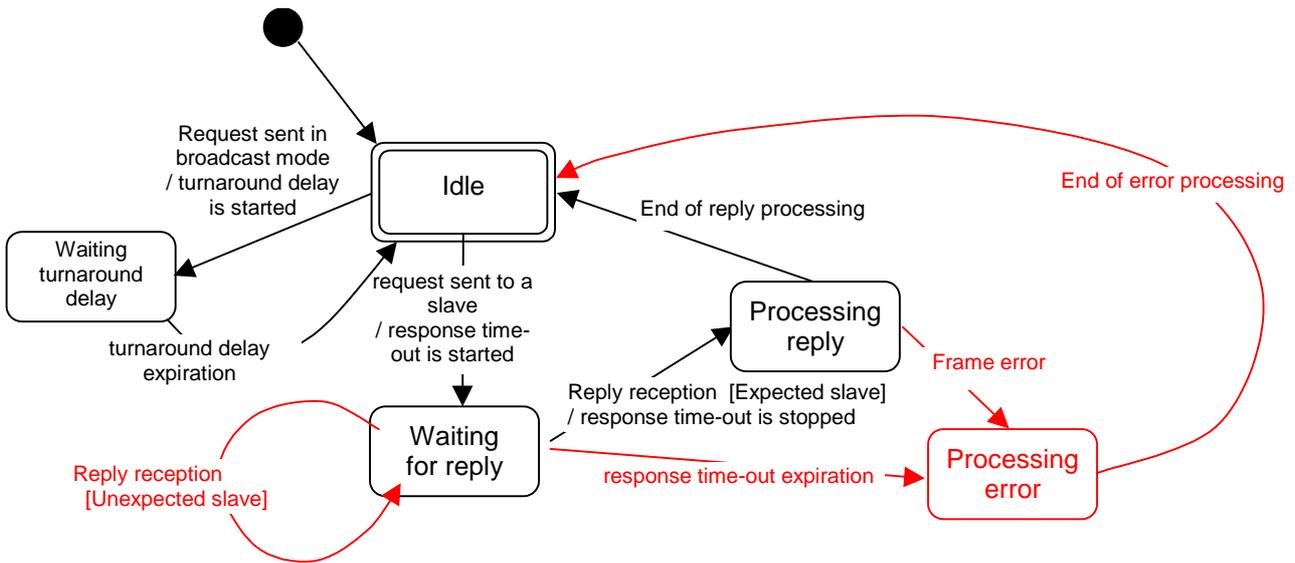


Figure 7: Master state diagram

Some explanations about the state diagram above :

- State "Idle" = no pending request. This is the initial state after power-up. A request can only be sent in "Idle" state. After sending a request, the Master leaves the "Idle" state, and cannot send a second request at the same time
- When a unicast request is sent to a slave, the master goes into "Waiting for reply" state, and a "Response Time-out" is started. It prevents the Master from staying indefinitely in "Waiting for reply" state. Value of the Response time-out is application dependant.
- When a reply is received, the Master checks the reply before starting the data processing. The checking may result in an error, for example a reply from an unexpected slave, or an error in the received frame. In case of a reply received from an unexpected slave, the Response time-out is kept running. In case of an error detected on the frame, a retry may be performed.
- If no reply is received, the Response time-out expires, and an error is generated. Then the Master goes into "Idle" state, enabling a retry of the request. The maximum number of retries depends on the master set-up.

- When a broadcast request is sent on the serial bus, no response is returned from the slaves. Nevertheless a delay is respected by the Master in order to allow any slave to process the current request before sending a new one. This delay is called "Turnaround delay". Therefore the master goes into "Waiting Turnaround delay" state before going back in "idle" state and before being able to send another request.
- In unicast the Response time out must be set long enough for any slave to process the request and return the response, in broadcast the Turnaround delay must be long enough for any slave to process only the request and be able to receive a new one. Therefore the Turnaround delay should be shorter than the Response time-out. Typically the Response time-out is from 1s to several second at 9600 bps; and the Turnaround delay is from 100 ms to 200ms.
- Frame error consists of : 1) Parity checking applied to each character; 2) Redundancy checking applied to the entire frame. See §2.6 "Error Checking Methods" for more explanations.

The state diagram is intentionally very simple. It does not take into account access to the line, message framing, or retry following transmission error, etc ... For more details about frame transmission, please refer to 2.5 paragraph, "The two serial Transmission Modes".

2.4.2 Slave State Diagram

The following drawing explains the Slave behavior :

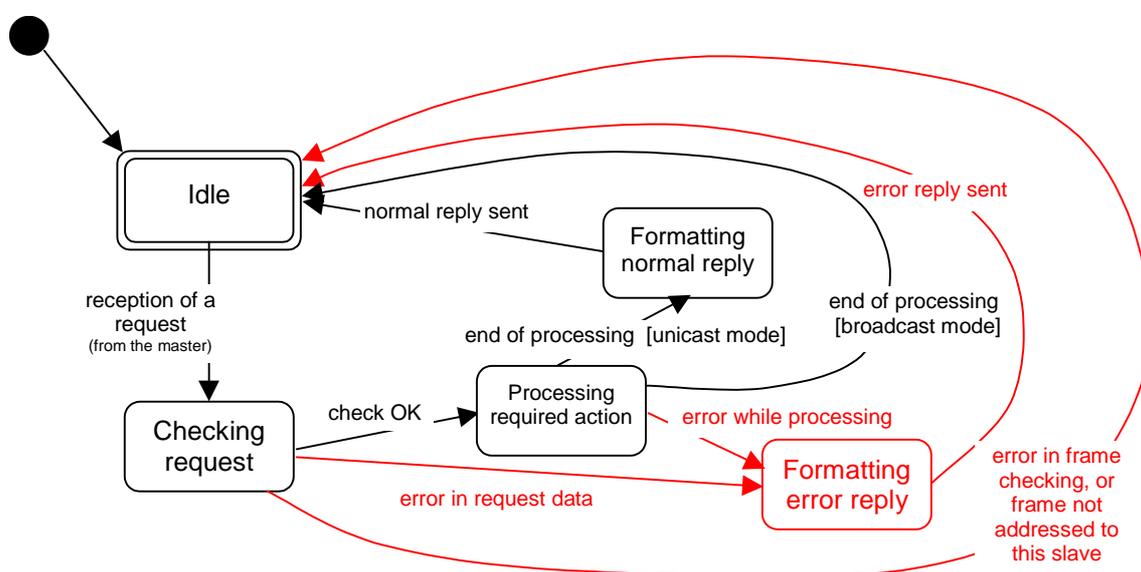


Figure 8: Slave state diagram

Some explanations about the above state diagram :

- State "Idle" = no pending request. This is the initial state after power-up.
- When a request is received, the slave checks the packet before performing the action requested in the packet. Different errors may occur : format error in the request, invalid action, ... In case of error, a reply must be sent to the master.
- Once the required action has been completed, a unicast message requires that a reply must be formatted and sent to the master.
- If the slave detects an error in the received frame, no respond is returned to the master.
- MODBUS diagnostics counters are defined and should be managed by any slave in order to provide diagnostic information. These counters can be get using the Diagnostic MODBUS function (see Appendix A, and the MODBUS application protocol specification [1]).

2.4.3 Master / Slave communication time diagram

This following figure shows the time diagram of 3 typical scenarios of Master / Slave communications.

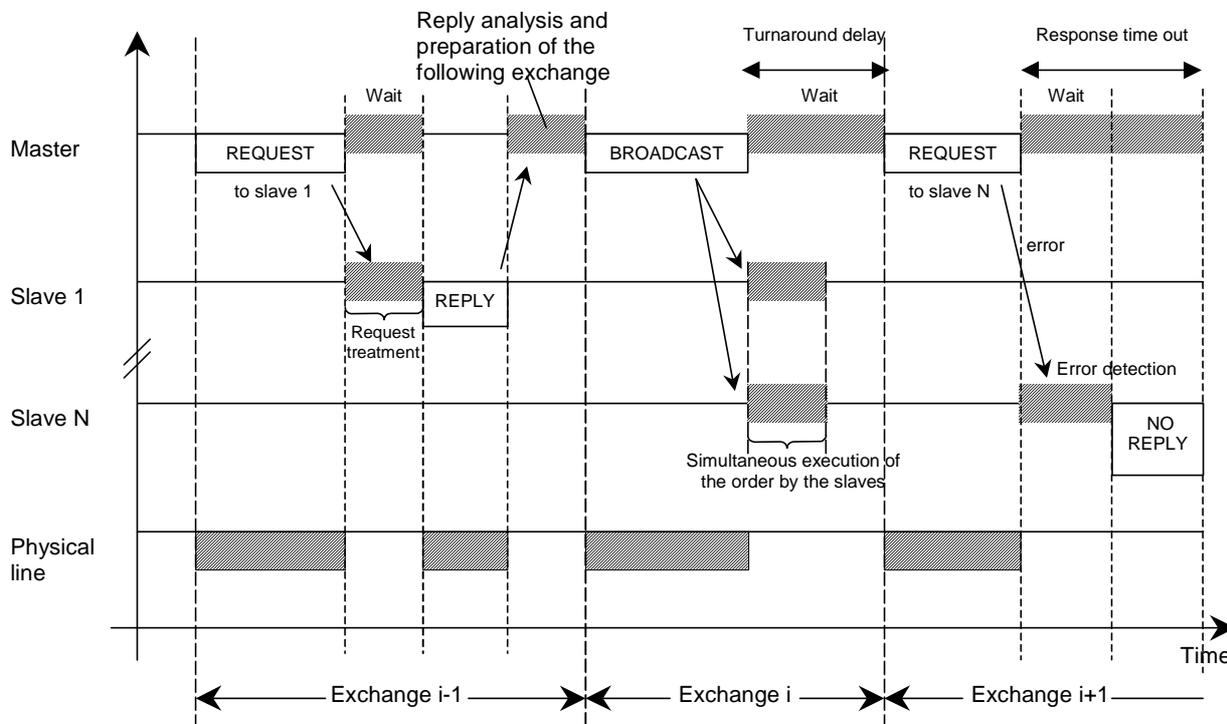


Figure 9: Master / Slave scenario time diagram

Remarks :

- the duration of the REQUEST, REPLY, BROADCAST phases depends on the communication features (frame length and throughput).
- the duration of the WAIT and TREATMENT phases depends on the request processing time needed for the slave application.

2.5 The two serial Transmission Modes

Two different serial transmission modes are defined : The RTU mode and the ASCII mode.

It defines the bit contents of message fields transmitted serially on the line. It determines how information is packed into the message fields and decoded.

The transmission mode (and serial port parameters) must be the same for all devices on a MODBUS Serial Line.

Although the ASCII mode is required in some specific applications, interoperability between MODBUS devices can be reached only if each device has the same transmission mode : **All devices must implement the RTU Mode.** The ASCII transmission mode is an option.

Devices should be set up by the users to the desired transmission mode, RTU or ASCII. Default setup must be the RTU mode.

2.5.1 RTU Transmission Mode

When devices communicate on a MODBUS serial line using the RTU (Remote Terminal Unit) mode, each 8-bit byte in a message contains two 4-bit hexadecimal characters. The main advantage of this mode is that its greater character density allows better data throughput than ASCII mode for the same baud rate. Each message must be transmitted in a continuous stream of characters.

The format for each byte (11 bits) in RTU mode is :

Coding System: 8-bit binary
Bits per Byte: 1 start bit
 8 data bits, least significant bit sent first
 1 bit for parity completion
 1 stop bit

Even parity is required, other modes (odd parity, no parity) may also be used. In order to ensure a maximum compatibility with other products, it is recommended to support also No parity mode. The default parity mode must be even parity.

Remark : the use of no parity requires 2 stop bits.

How Characters are Transmitted Serially :

Each character or byte is sent in this order (left to right):

Least Significant Bit (LSB) . . . Most Significant Bit (MSB)

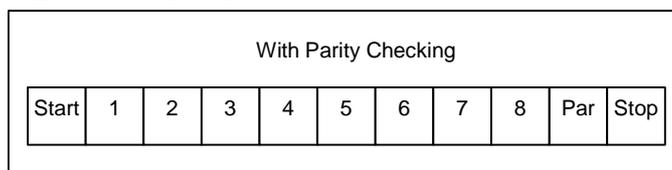


Figure 10: Bit Sequence in RTU mode

Devices may accept by configuration either Even, Odd, or No Parity checking. If No Parity is implemented, an additional stop bit is transmitted to fill out the character frame to a full 11-bit asynchronous character :

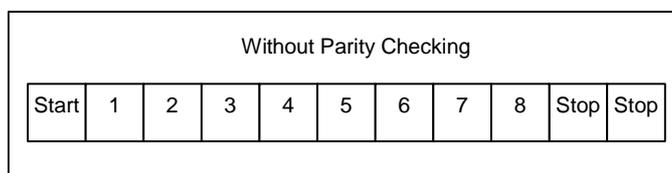


Figure 11: Bit Sequence in RTU mode (specific case of No Parity)

Frame Checking Field : Cyclical Redundancy Checking (CRC)

Frame description :

Slave Address	Function Code	Data	CRC
1 byte	1 byte	0 up to 252 byte(s)	2 bytes CRC Low CRC Hi

Figure 12: RTU Message Frame

→ The maximum size of a MODBUS RTU frame is 256 bytes.

2.5.1.1 MODBUS Message RTU Framing

A MODBUS message is placed by the transmitting device into a frame that has a known beginning and ending point. This allows devices that receive a new frame to begin at the start of the message, and to know when the message is completed. Partial messages must be detected and errors must be set as a result.

In RTU mode, message frames are separated by a silent interval of at least 3.5 character times. In the following sections, this time interval is called t_{3,5}.

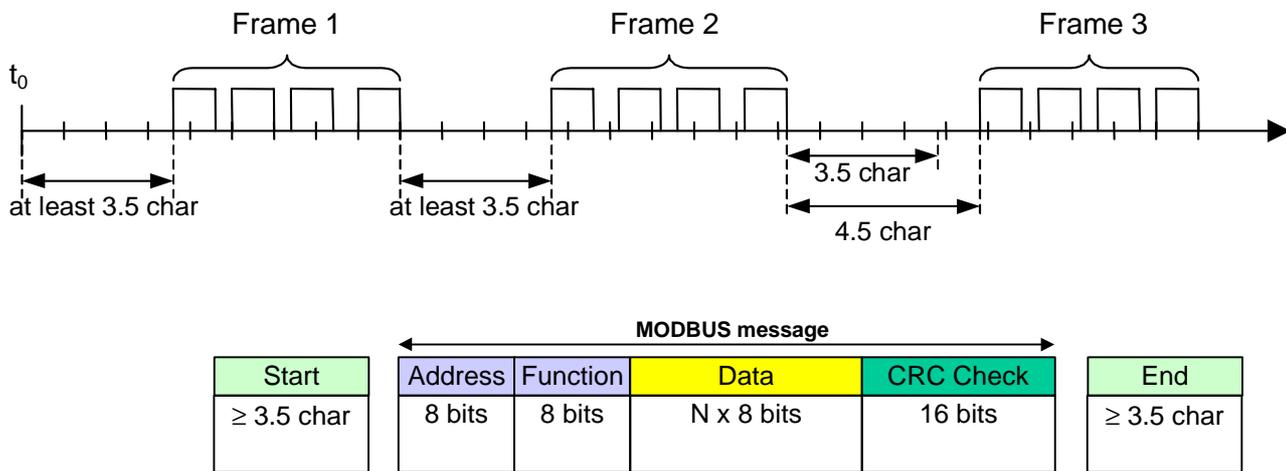
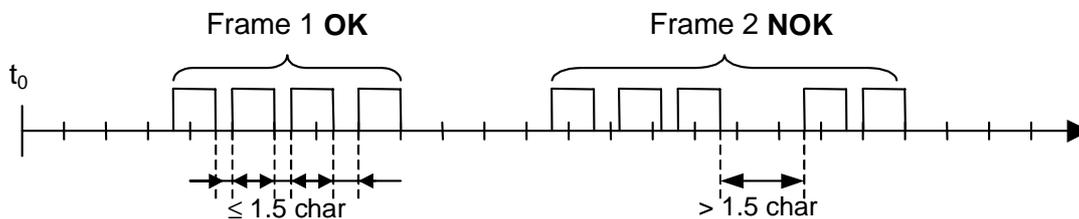


Figure 13: RTU Message Frame

The entire message frame must be transmitted as a continuous stream of characters.

If a silent interval of more than 1.5 character times occurs between two characters, the message frame is declared incomplete and should be discarded by the receiver.



Remark :

The implementation of RTU reception driver may imply the management of a lot of interruptions due to the t_{1,5} and t_{3,5} timers. With high communication baud rates, this leads to a heavy CPU load. Consequently these two timers must be strictly respected when the baud rate is equal or lower than 19200 Bps. For baud rates greater than 19200 Bps, fixed values for the 2 timers should be used: it is recommended to use a value of 750µs for the inter-character time-out (t_{1,5}) and a value of 1.750ms for inter-frame delay (t_{3,5}).

The following drawing provides a description of the RTU transmission mode state diagram. Both "master" and "slave" points of view are expressed in the same drawing :

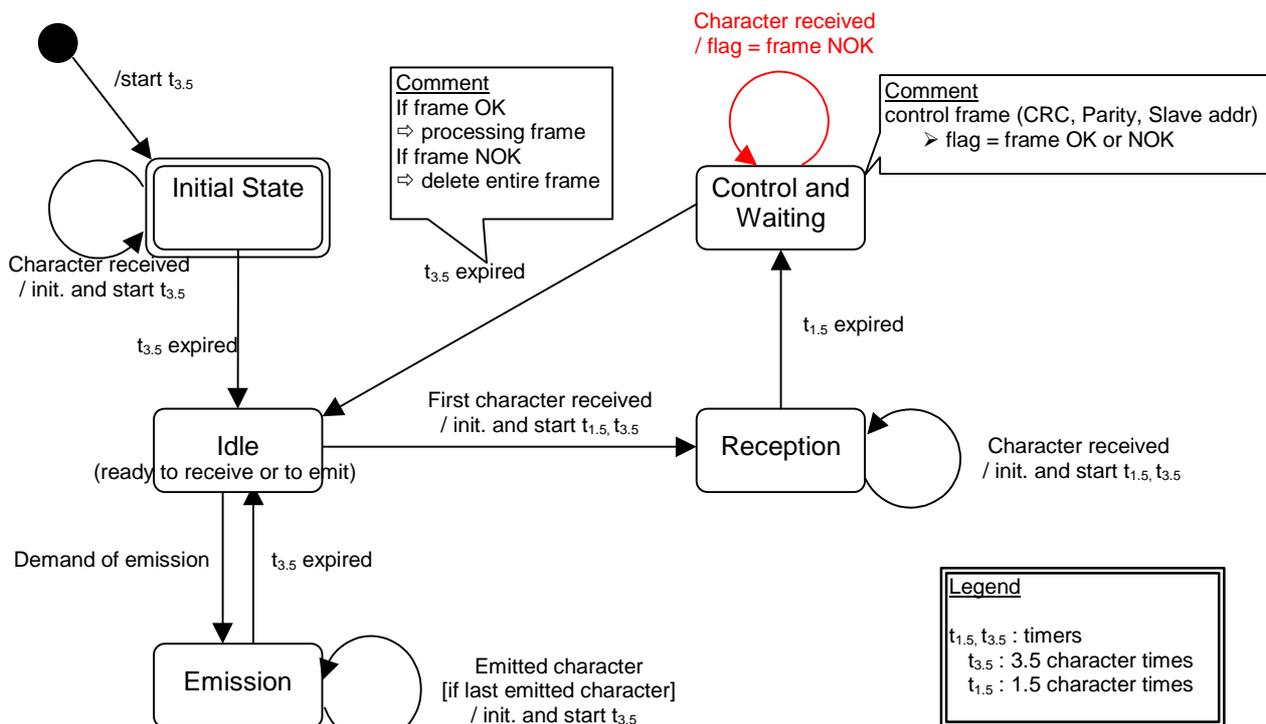


Figure 14: RTU transmission mode state diagram

Some explanations about the above state diagram:

- Transition from "Initial State" to "Idle" state needs t_{3.5} time-out expiration : that insures inter-frame delay
- "Idle" state is the normal state when neither emission nor reception is active.
- In RTU mode, the communication link is declared in "idle" state when there is no transmission activity after a time interval equal to at least 3,5 characters.
- When the link is in idle state, each transmitted character detected on the link is identified as the **start of a frame**. The link goes to the "active" state. Then, the **end of frame** is identified when no more character is transmitted on the link after the time interval t_{3.5}.
- After detection of the end of frame, the CRC calculation and checking is completed. Afterwards the address field is analysed to determine if the frame is for the device. If not the frame is discarded. In order to reduce the reception processing time the address field can be analysed as soon as it is received without waiting the end of frame. In this case the CRC will be calculated and checked only if the frame is addressed to the slave (broadcast frame included).

2.5.1.2 CRC Checking

The RTU mode includes an error-checking field that is based on a Cyclical Redundancy Checking (**CRC**) method performed on the message contents.

The CRC field checks the contents of the entire message. It is applied regardless of any parity checking method used for the individual characters of the message.

The CRC field contains a 16-bit value implemented as two 8-bit bytes.

The CRC field is appended to the message as the last field in the message. When this is done, the low-order byte of the field is appended first, followed by the high-order byte. The CRC high-order byte is the last byte to be sent in the message.

The CRC value is calculated by the sending device, which appends the CRC to the message. The receiving device recalculates a CRC during receipt of the message, and compares the calculated value to the actual value it received in the CRC field. If the two values are not equal, an error results.

The CRC calculation is started by first pre-loading a 16-bit register to all 1's. Then a process begins of applying successive 8-bit bytes of the message to the current contents of the register. Only the eight bits of data in each character are used for generating the CRC. Start and stop bits and the parity bit, do not apply to the CRC.

During generation of the CRC, each 8-bit character is exclusive ORed with the register contents. Then the result is shifted in the direction of the least significant bit (LSB), with a zero filled into the most significant bit (MSB) position. The LSB is extracted and examined. If the LSB was a 1, the register is then exclusive ORed with a preset, fixed value. If the LSB was a 0, no exclusive OR takes place.

This process is repeated until eight shifts have been performed. After the last (eight) shift, the next 8-bit byte is exclusive ORed with the register's current value, and the process repeats for eight more shifts as described above. The final content of the register, after all the bytes of the message have been applied, is the CRC value.

When the CRC is appended to the message, the low-order byte is appended first, followed by the high-order byte. A detailed example of CRC generation is contained in Appendix B.

2.5.2 The ASCII Transmission Mode

When devices are setup to communicate on a MODBUS serial line using ASCII (American Standard Code for Information Interchange) mode, each 8-bit byte in a message is sent as two ASCII characters. This mode is used when the physical communication link or the capabilities of the device does not allow the conformance with RTU mode requirements regarding timers management.

Remark : this mode is less efficient than RTU since each byte needs two characters.

→ Example : The byte 0X5B is encoded as two characters : 0x35 and 0x42 (0x35 ="5", and 0x42 ="B" in ASCII).

The format for each byte (10 bits) in ASCII mode is :

- Coding System:** Hexadecimal, ASCII characters 0–9, A–F
 One hexadecimal character contains 4-bits of data within each ASCII character of the message
- Bits per Byte:** 1 start bit
 7 data bits, least significant bit sent first
 1 bit for parity completion;
 1 stop bit

Even parity is required, other modes (odd parity, no parity) may also be used. In order to ensure a maximum compatibility with other products, it is recommended to support also No parity mode. The default parity mode must be Even parity.

Remark : the use of no parity requires 2 stop bits.

How Characters are Transmitted Serially :

Each character or byte is sent in this order (left to right):

Least Significant Bit (LSB) . . . Most Significant Bit (MSB)

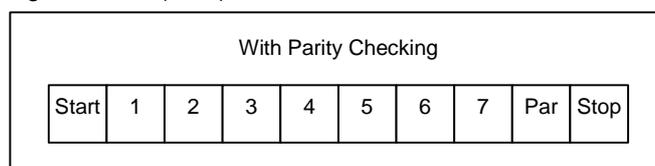


Figure 15: Bit Sequence in ASCII mode

Devices may accept by configuration either Even, Odd, or No Parity checking. If No Parity is implemented, an additional stop bit is transmitted to fill out the character frame :

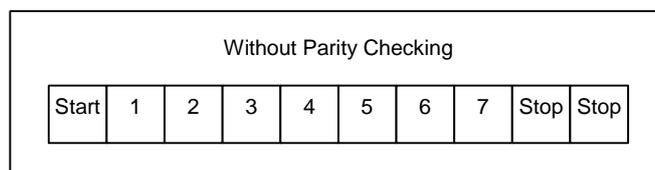


Figure 16: Bit Sequence in ASCII mode (specific case of No Parity)

Frame Checking Field: Longitudinal Redundancy Checking (LRC)

2.5.2.1 MODBUS Message ASCII Framing

A MODBUS message is placed by the transmitting device into a frame that has a known beginning and ending point. This allows devices that receive a new frame to begin at the start of the message, and to know when the message is completed. Partial messages must be detected and errors must be set as a result.

The address field of a message frame contains two characters.

In ASCII mode, a message is delimited by specific characters as Start-of-frames and End-of-frames. A message must start with a 'colon' (:) character (ASCII 3A hex), and end with a 'carriage return – line feed' (CRLF) pair (ASCII 0D and 0A hex).

Remark : The LF character can be changed using a specific MODBUS application command (see MODBUS application protocol specification).

The allowable characters transmitted for all other fields are hexadecimal 0–9, A–F (ASCII coded). The devices monitor the bus continuously for the 'colon' character. When this character is received, each device decodes the next character until it detects the End-Of-Frame.

Intervals of up to one second may elapse between characters within the message. Unless the user has configured a longer timeout, an interval greater than 1 second means an error has occurred. Some Wide-Area-Network application may require a timeout in the 4 to 5 second range.

A typical message frame is shown below.

Start	Address	Function	Data	LRC	End
1 char :	2 chars	2 chars	0 up to 2x252 char(s)	2 chars	2 chars CR,LF

Figure 17: ASCII Message Frame

Remark : Each data byte needs two characters for encoding. Thus, to ensure compatibility at MODBUS application level between ASCII mode and RTU mode, the maximum data size for ASCII data field (2x252) is the double the maximum data size for RTU data field (252). Consequently, the maximum size of a MODBUS ASCII frame is 513 characters.

The ASCII framing requirements are synthesized in the following state diagram. Both "master" and "slave" points of view are expressed in the same drawing :

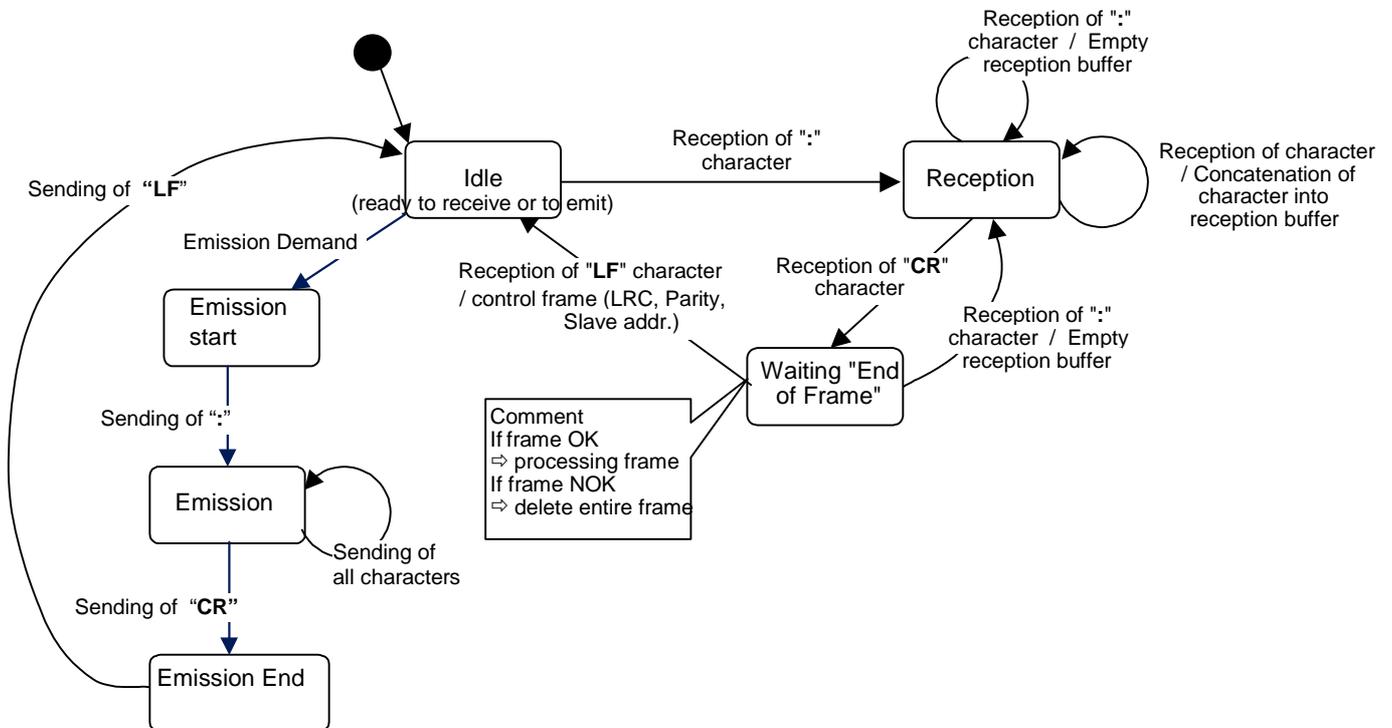


Figure 18: ASCII Transmission mode State diagram

Some explanations about the above state diagram :

- "Idle" state is the normal state when neither emission nor reception is active.
- Each reception of a ":" character means a beginning of a new message. If a message was in process of reception while receiving such a character, the current message is declared incomplete and it is discarded. A new reception buffer is then allocated.
- After detection of the end of frame, the LRC calculation and checking is completed. Afterwards the address field is analyzed to determine if the frame is for the device. If not the frame is discarded. In order to reduce the reception processing time the address field can be analyzed as soon as it is reserved without waiting the end of frame.

2.5.2.2 LRC Checking

In ASCII mode, messages include an error-checking field that is based on a Longitudinal Redundancy Checking (**LRC**) calculation that is performed on the message contents, exclusive of the beginning 'colon' and terminating CRLF pair characters. It is applied regardless of any parity checking method used for the individual characters of the message.

The LRC field is one byte, containing an 8-bit binary value. The LRC value is calculated by the device that emits, which appends the LRC to the message. The device that receives calculates an LRC during receipt of the message, and compares the calculated value to the actual value it received in the LRC field. If the two values are not equal, an error results.

The LRC is calculated by adding together successive 8-bit bytes of the message, discarding any carries, and then two's complementing the result. It is performed on the ASCII message field contents excluding the 'colon' character that begins the message, and excluding the CRLF pair at the end of the message. In ASCII mode, the resulting LRC is ASCII encoded into two bytes and placed at the end of ASCII mode frame prior to the CRLF.

A detailed example of LRC generation is contained in Appendix B.

2.6 Error Checking Methods

The security of standard MODBUS Serial Line is based on two kinds of error checking :

- Parity checking (even or odd) should be applied to each character.
- Frame checking (LRC or CRC) must be applied to the entire message.

Both the character checking and message frame checking are generated in the device (master or slave) that emits and applied to the message contents before transmission. The device (slave or master) checks each character and the entire message frame during receipt.

The master is configured by the user to wait for a predetermined timeout interval (Response time-out) before aborting the transaction. This interval is set to be long enough for any slave to respond normally (unicast request). If the slave detects a transmission error, the message will not be acted upon. The slave will not construct a response to the master. Thus the timeout will expire and allow the master's program to handle the error. Note that a message addressed to a nonexistent slave device will also cause a timeout.

2.6.1 Parity Checking

Users may configure devices for Even (required) or Odd Parity checking, or for No Parity checking (recommended). This will determine how the parity bit will be set in each character.

If either Even or Odd Parity is specified, the quantity of 1 bits will be counted in the data portion of each character (seven data bits for ASCII mode, or eight for RTU). The parity bit will then be set to a 0 or 1 to result in an Even or Odd total of 1 bits.

For example, these eight data bits are contained in an RTU character frame:

1100 0101

The total quantity of 1 bits in the frame is four. If Even Parity is used, the frame's parity bit will be a 0, making the total quantity of 1 bits still an even number (four). If Odd Parity is used, the parity bit will be a 1, making an odd quantity (five).

When the message is transmitted, the parity bit is calculated and applied to the frame of each character. The device that receives counts the quantity of 1 bits and sets an error if they are not the same as configured for that device (all devices on the MODBUS Serial Line must be configured to use the same parity checking method).

Note that parity checking can only detect an error if an odd number of bits are picked up or dropped in a character frame during transmission. For example, if Odd Parity checking is employed, and two 1 bits are dropped from a character containing three 1 bits, the result is still an odd count of 1 bits.

If No Parity checking is specified, no parity bit is transmitted and no parity checking can be made. An additional stop bit is transmitted to fill out the character frame.

2.6.2 Frame Checking

Two kinds of frame checking is used depending on the transmission mode, RTU or ASCII.

- In RTU mode, messages include an error-checking field that is based on a Cyclical Redundancy Checking (CRC) method. The CRC field checks the contents of the entire message. It is applied regardless of any parity checking method used for the individual characters of the message.
- In ASCII mode, messages include an error-checking field that is based on a Longitudinal Redundancy Checking (LRC) method. The LRC field checks the contents of the message, exclusive of the beginning 'colon' and ending CRLF pair. It is applied regardless of any parity checking method used for the individual characters of the message.

The detailed information about error checking methods is contained in the previous sections.

3 Physical Layer

3.1 Preamble

A new MODBUS solution over serial line should implement an electrical interface in accordance with EIA/TIA-485 standard (also known as RS485 standard). This standard allows point to point and multipoint systems, in a "two-wire configuration". In addition, some devices may implement a "Four-Wire" RS485-Interface.

A device may also implement an RS232-Interface.

In such a MODBUS system, a Master Device and one or several Slave Devices communicate on a passive serial line.

On standard MODBUS system, all the devices are connected (in parallel) on a trunk cable constituted by 3 conductors. Two of those conductors (the "Two-Wire" configuration) form a balanced twisted pair, on which bi-directional data are transmitted, typically at the bit rate of 9600 bits per second.

Each device may be connected (see figure 19):

- either directly on the trunk cable, forming a daisy-chain,
- either on a passive Tap with a derivation cable,
- either on an active Tap with a specific cable.

Screw Terminals, RJ45, or D-shell 9 connectors may be used on devices to connect cables (see the chapter "Mechanical Interfaces").

3.2 Data Signaling Rates

9600 bps and 19.2 Kbps are required and 19.2 is the required default

Other baud rates may optionally be implemented : 1200, 2400, 4800, ... 38400 bps, 56 Kbps, 115 Kbps, ...

Every implemented baud rate must be respected better than 1% in transmission situation, and must accept an error of 2% in reception situation.

3.3 Electrical Interfaces

3.3.1 Multipoint Serial Bus Infrastructure

Figure 19 gives a general overview of the serial bus infrastructure in a MODBUS multipoint Serial Line system.

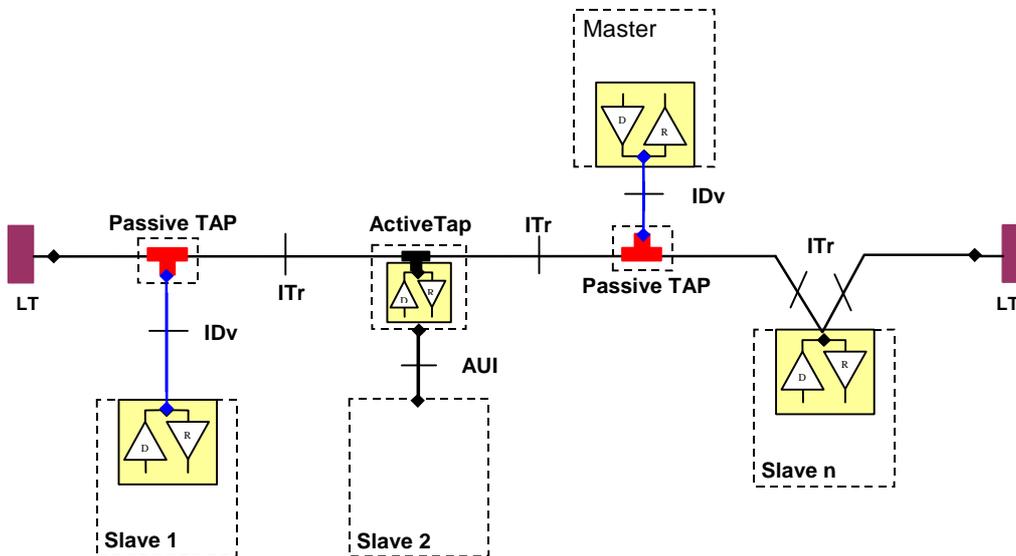


Figure 19 : Serial bus infrastructure

A multipoint MODBUS Serial Line bus is made of a principal cable (**the Trunk**), and possibly some **derivation** cables. Line terminations are necessary at each extremity of the trunk cable for impedance adaptation (see § "Two-Wire MODBUS Definition" & "Optional Four-Wire MODBUS Definition" for details).

As shown in figure 19, different implementations may operate in the same MODBUS Serial Line system :

- the device integrates the communication transceiver and is connected to the trunk using a **Passive Tap** and a derivation cable (**case of Slave 1 and Master**) ;
- the device doesn't integrate the communication transceiver and is connected to the trunk using an **Active Tap** and a derivation cable (the active TAP integrates the transceiver) (**case of Slave 2**) ;
- the device is connected directly to the trunk cable, in a **Daisy-Chain** (**case of Slave n**)

The following conventions are adopted :

- The interface with the **trunk** is named **ITr** (Trunk Interface)
- The interface between the device and the **Passive Tap** is named **IDv** (Derivation Interface)
- The interface between the device and the **Active Tap** is named **AUI** (Attachment Unit Interface)

Remarks :

1. In some cases, the Tap may be connected directly to the IDv-socket or the AUI-socket of the device, without using a derivation cable.
2. A Tap may have several IDv sockets to connect several devices. Such a Tap is named **Distributor** when it is a passive one.
3. When using an active Tap, power supply of the Tap may be provided either via its AUI or ITr interface.

ITr and **IDv** interfaces are described in the following chapters (see § "Two-Wire MODBUS DEFINITION" & "Four-Wire MODBUS DEFINITION").

3.3.2 Two-Wire MODBUS Definition

A MODBUS solution over serial line should implement a “Two-Wire” electrical interface in accordance with EIA/TIA-485 standard.

On such a 2W-bus, at any time one driver only has the right for transmitting.

In fact a third conductor must also interconnect all the devices of the bus : the common.

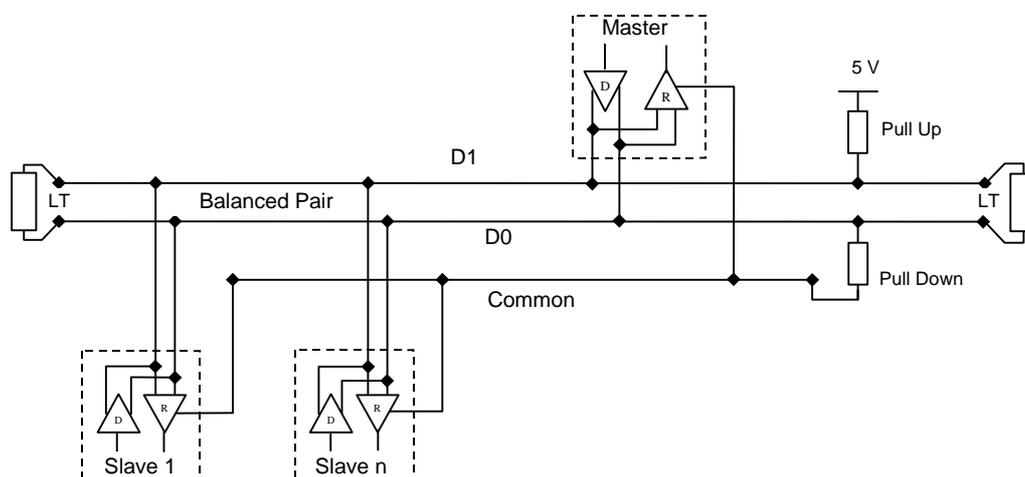


Figure 20: General 2-Wire Topology

2W-MODBUS Circuits Definition

Required Circuits		For device	Required on device	EIA/TIA-485 name	Description
on ITr	on IDv				
D1	D1	I/O	X	B/B'	Transceiver terminal 1, V1 Voltage (V1 > V0 for binary 1 [OFF] state)
D0	D0	I/O	X	A/A'	Transceiver terminal 0, V0 Voltage (V0 > V1 for binary 0 [ON] state)
Common	Common	--	X	C/C'	Signal and optional Power Supply Common

Notes :

- For Line Termination (LT), Pull Up and Pull Down resistors, please refer to section “Multipoint System requirements”.
- D0, D1, and Common circuit names must be used in the documentation related to the device and the Tap (User Guide, Cabling Guide, ...) to facilitate interoperability.
- Optional electrical interfaces may be added, for example :
 - **Power Supply** : 5..24 V D.C.
 - **Port mode control** : PMC circuit (TTL compatible). When needed, port mode may be controlled either by this external circuit and/or by another way (a switch on the device for example). In the first case while an open circuit PMC will ask for the 2W-MODBUS mode, a Low level on PMC will switch the port into 4W-MODBUS or RS232-MODBUS Mode, depending on the implementation.

3.3.3 Optional Four-Wire MODBUS Definition

Optionally, such MODBUS devices also permit to implement a **2-pair** bus (4 wires) of mono directional data. The data on the **master pair** (RXD1-RXD0) are only received by the slaves ; the data on the **slave pair** (TXD1-TXD0) are only received by the only master.

In fact a fifth conductor must also interconnect all the devices of the 4W-bus : the common.

In the same way as on a 2W-MODBUS, at any time one driver only has the right for emitting.

Such a device must implement, for each balanced pair, a driver and a transceiver **in accordance with EIA/ TIA-485**. (Sometimes this solution has been named "RS422", which is not correct : the RS422 standard does not support several drivers on one balanced pair.)

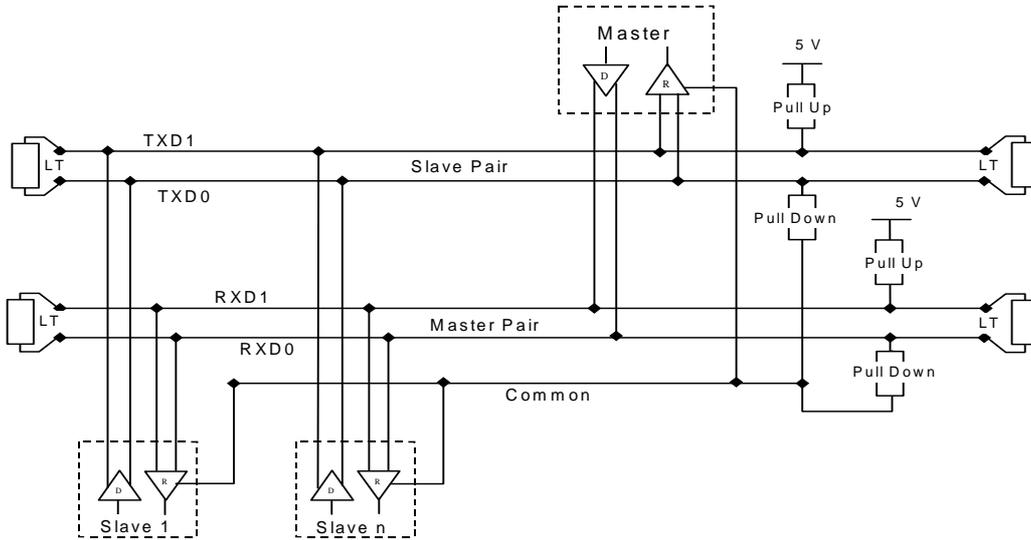


Figure 21: General 4-wire topology

Optional 4W-MODBUS Circuits Definition

Required Circuits		For device	Required on device	EIA/TIA-485 name	Description for IDv
on ITr	on IDv				
TXD1	TXD1	Out	X	B	Generator terminal 1 , Vb Voltage (Vb > Va for binary 1 [OFF] state)
TXD0	TXD0	Out	X	A	Generator terminal 0 , Va Voltage (Va > Vb for binary 0 [ON] state)
RXD1	RXD1	In	(1)	B'	Receiver terminal 1 , Vb' Voltage (Vb' > Va' for binary 1 [OFF] state)
RXD0	RXD0	In	(1)	A'	Receiver terminal 0 , Va' Voltage (Va' > Vb' for binary 0 [ON] state)
Common	Common	--	X	C/C'	Signal and optional Power Supply Common

Notes :

- For Line Termination (LT), Pull Up and Pull Down resistors, please refer to section "Multipoint System requirements".
- Those circuits (1) are required only if an 4W-MODBUS option is implemented.
- The name of the 5 required circuits must be used in the documentation related to the device and the Tap (User Guide, Cabling Guide, ...) to facilitate interoperability.
- Optional electrical interfaces may be added, for example :
 - **Power Supply** : 5..24 V D.C.
 - **PMC circuit** : See above (In 2W-MODBUS Circuits Definition) the note about this optional circuit.

3.3.3.1 4W-Cabling System Important Topic

In such a 4W-MODBUS, Master Device and Slave Devices have IDv interfaces with the same 5 required circuits. As the master has to :

- receive from the slave the data on the slave pair (TXD1-TXD0),
- and transmit on the master pair (RXD1-RXD0 , received by the slaves) ,

the 4W-cabling system must cross the two pairs of the bus between ITr and the IDv of the master :

	Signal on Master IDv		EIA/TIA-485 Name	Circuit on ITr
	Name	Type		
Slave Pair	RXD1	In	B'	TXD1
	RXD0	In	A'	TXD0
Master Pair	TXD1	Out	B	RXD1
	TXD0	Out	A	RXD0
	Common	--	C/C'	Common

This crossing may be implemented by crossed cables, but the connection of such crossed cables in a 2-wire system may cause damages. To connect a 4W master device (which have a MODBUS connector) a better solution is to use a Tap which includes the crossing function.

3.3.3.2 Compatibility between 4-Wire and 2-Wire cabling

In order to connect devices implementing a 2-Wire physical interface to an already existing 4-Wire system, the 4-Wire cabling system can be modified as described below :

- TxD0 signal shall be wired with the RxD0 signal, turning them to the D0 signal
- TxD1 signal shall be wired with the RxD1 signal, turning them to the D1 signal.
- Pull-up, Pull-down and line terminations resistors shall be re-arranged to correctly adapt the D0, D1 signals.

The figure hereafter gives an example where slaves 2 and 3 which use a 2-Wire interface can operate with the Master and the slave 1 which use a 4-Wire interface.

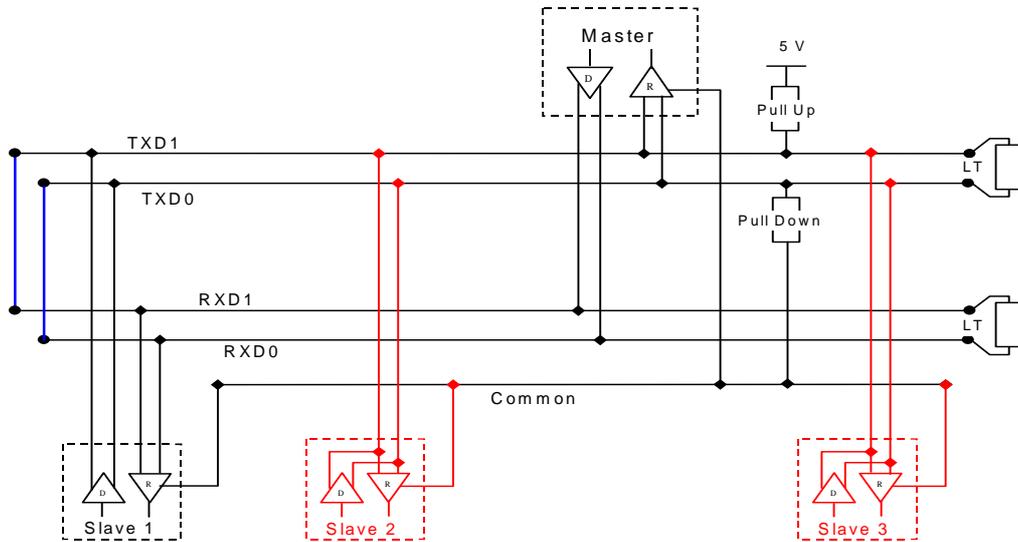


Figure 22 : Changing a 4-Wire cabling system into a 2-Wire cabling system

In order to connect devices implementing a 4-Wire physical interface to an already existing 2-Wire system, the 4-Wire interface of the new coming devices can be arranged as describe below :

On each 4-Wire device interface :

- TXD0 signal shall be wired with the RxD0 signal and then connected to the D0 signal of the trunk ;
- TXD1 signal shall be wired with the RxD1 signal and then connected to the D1 signal of the trunk.

The figure hereafter gives an example where slaves 2 and 3 which use a 4-Wire interface can operate with the Master and the slave 1 which use a 2-Wire interface.

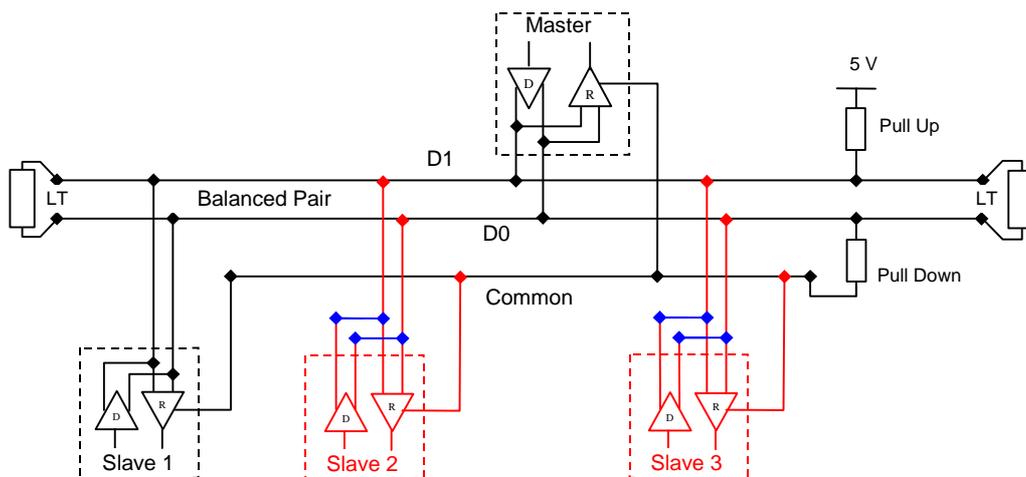


Figure 23 : Connecting devices with 4-Wire interface to a 2-Wire cabling system

3.3.4 RS232-MODBUS Definition

Some devices may implement an RS232-Interface between a DCE and a DTE.

Optional RS232-MODBUS Circuits Definition

Signal	For DCE	<u>Required on DCE (1)</u>	<u>Required on DTE (1)</u>	Description
Common	--	X	X	Signal Common
CTS	In			Clear to Send
DCD	--			Data Carrier Detected (from DCE to DTE)
DSR	In			Data Set Ready
DTR	Out			Data Terminal Ready
RTS	Out			Request to Send
RXD	In	X	X	Received Data
TXD	Out	X	X	Transmitted Data

Notes :

- “X” marked signals are required only if an RS232-MODBUS option is implemented.
- Signals are in accordance with EIA/ TIA-232.
- Each TXD must be wired with RXD of the other device ;
- RTS may be wired with CTS of the other device,
- DTR may be wired with DSR of the other device.
- Optional electrical interfaces may be added, for example :
 - **Power Supply :** 5..24 V D.C.
 - **PMC circuit :** See above (In 2W-MODBUS Circuits Definition) the note about this optional circuit.

3.3.5 RS232-MODBUS requirements

This optional MODBUS on Serial Line system should only be used for short length (typically less than 20m) point to point inter-connection.

Then, the EIA/TIA-232 standard must be respected :

- ⇒ circuits definition,
- ⇒ maximum wire capacitance to ground (2500 pF, then 25 m for a 100 pF/m cable).

Please refer to chapter “Cables” for the shield, and for the possibility to use Category 5 Cables.

Documentation of the device must indicate :

- ⇒ if the device must be considered as a DCE either as a DTE,
- ⇒ how optional circuits must work if such is the case.

3.4 Multipoint System requirements

For any EIA/ TIA-485 multipoint system, in either 2-wire or 4-wire configuration, the following requirements all apply.

3.4.1 Maximum number of devices without repeater

A figure of **32 devices** is always authorized on any RS485-MODBUS system without repeater.

Depending of :

- all the possible addresses,
- the figure of RS485 Unit Load used by the devices,
- and the line polarization in need be,

A RS485 system may implement a larger number of devices. Some devices allow the implementation of a RS485-MODBUS serial line with more than 32 devices, without repeater.

In this case these MODBUS devices must be documented to say how many of such devices are authorized without repeater.

The use of a **repeater** between two heavy loaded RS485-MODBUS is also possible.

3.4.2 Topology

An RS485-MODBUS configuration without repeater has one trunk cable, along which devices are connected, directly (daisy chaining) or by short derivation cables.

The trunk cable, also named "Bus", can be long (see hereafter). Its two ends must be connected on Line Terminations.

The use of repeaters between several RS485-MODBUS is also possible.

3.4.3 Length

The end to end length of the **trunk cable** must be limited. The maximum length depends on the baud rate, the cable (Gauge, Capacitance or Characteristic Impedance), the number of loads on the daisy chain, and the network configuration (*2-wire or 4-wire*).

For a maximum 9600 Baud Rate and AWG26 (or wider) gauge, the maximum length is 1000m. In the specific case shown in the figure 22 (4 Wire cabling used as a 2 Wire cabling system) the maximum length must be divided by two.

The **derivations** must be short, never more than 20m. If a multi-port tap is used with n derivations, each one must respect a maximum length of 40m divided by n.

3.4.4 Grounding Arrangements

The « Common » circuit (Signal and optional Power Supply Common) must be connected directly to protective ground, preferably at **one point only** for the entire bus. Generally this point is to choose on the master device or on its Tap.

3.4.5 Line Termination

A reflection in a transmission line is the result of an impedance discontinuity that a travelling wave sees as it propagates down the line. To minimize the reflections from the end of the RS485-cable it is required to place a Line Termination **near each of the 2 Ends** of the Bus.

It is important that the line be terminated at **both** ends since the propagation is bi-directional, but it is not allowed to place more than 2 LT on one passive D0-D1 balanced pair . Never place any LT on a derivation cable.

Each line termination must be connected between the two conductors of the balanced line : D0 and D1.

Line termination may be a 150 ohms value (0.5 W) resistor.

A serial capacitor (1 nF, 10 V minimum) with a 120 Ohms (0.25 W) resistor is a better choice when a polarization of the pair must be implemented (see here after).

In a 4W-system, each pair must be terminated at each end of the bus.

In an RS232 interconnections, no termination should be wired.

3.4.6 Line Polarization

When there is no data activity on an RS-485 balanced pair, the lines are not driven and, thus susceptible to external noise or interference. To insure that its receiver stays in a constant state, when no data signal is present, some devices need to bias the network.

Each MODBUS device must be documented to say :

- if the device needs a line polarization,
- if the device implements, or can implement, such a line polarization.

If one or several devices need polarization, **one** pair of resistors must be connected on the RS-485 balanced pair :

- a Pull-Up Resistor to a 5V Voltage on D1 circuit,
- a Pull-Down Resistor to the common circuit on D0 circuit.

The value of those resistors must be between 450 Ohms and 650 Ohms. 650 Ohms resistors value may allow a higher number of devices on the serial line bus.

In this case, a polarization of the pair must be implemented **at one location for the whole Serial Bus**. Generally this point is to choose on the master device or on its Tap. Other devices must not implement any polarization.

The maximum number of devices authorized on such a MODBUS Serial Line is reduced by 4 from a MODBUS without polarization.

3.5 Mechanical Interfaces

Screw Terminals may be used for both IDv and ITr connections. All information must be provided to the users about the exact location of each signal, with names in accordance with the previous chapter “Electrical Interface”.

If a RJ45 (or a mini-DIN or a D-Shell) **connector** is used on an equipment for a MODBUS mechanical interface, a **shielded female connector** must be chosen. Then the cable-end must have a shielded male connector.

3.5.1 Connectors pin-out for 2W-MODBUS

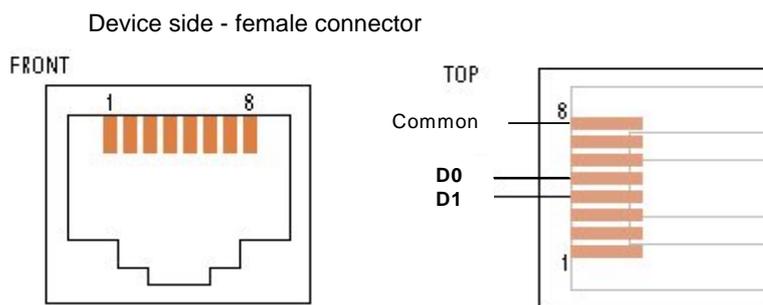


Figure 24: 2W- MODBUS on RJ45 connector (required pin-out)

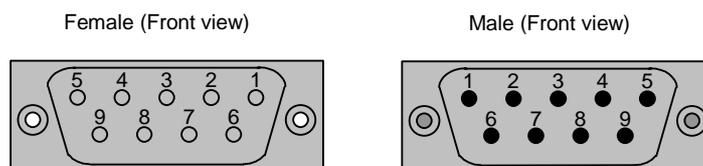


Figure 25: D-shell 9-pin connector

Screw type connectors can also be used.

If an RJ45 or a 9-pin D-shell connector is used for a standard MODBUS device, the pinouts hereafter must be respected for every implemented circuit.

2W-MODBUS RJ45 and 9-pin D-shell Pinouts

Pin on RJ45	Pin on D9-shell	Level of requirement	IDv Circuit	ITr Circuit	EIA/TIA-485 name	Description for IDv
3	3	optional	PMC	--	--	Port Mode Control
4	5	required	D1	D1	B/B'	Transceiver terminal 1, V1 Voltage (V1 > V0 for binary 1 [OFF] state)
5	9	required	D0	D0	A/A'	Transceiver terminal 0, V0 Voltage (V0 > V1 for binary 0 [ON] state)
7	2	recommended	VP	--	--	Positive 5...24 V D.C. Power Supply
8	1	required	Common	Common	C/C'	Signal and Power Supply Common

3.5.2 Connectors pin-out for optional 4W-MODBUS

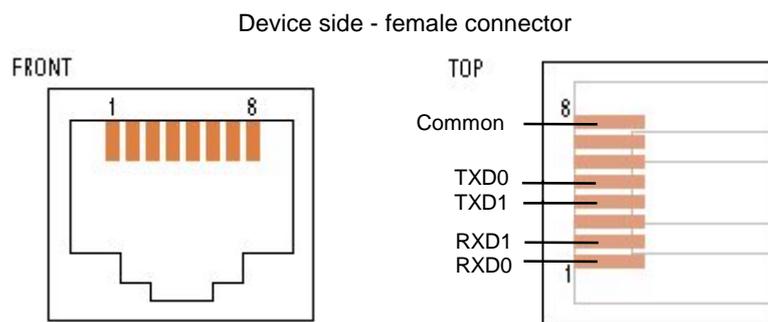


Figure 26: 4W-MODBUS on RJ45 connector (required pin-out)

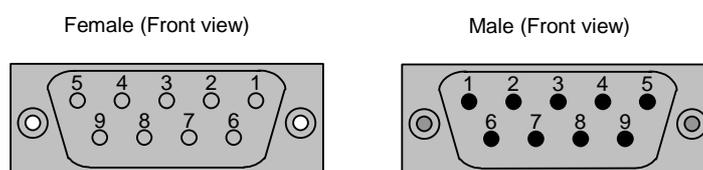


Figure 27: D-shell 9-pin connector

Screw type connectors can also be used.

If an RJ45 or a 9-pin D-shell connector is used for a 4W-MODBUS device, the pinouts hereafter must be respected for every implemented circuit.

Optional 4W-MODBUS RJ45 and 9-pin D-shell Pinouts

Pin on RJ45	Pin on D9-shell	Level of requirement	IDv Signal	ITr Signal	EIA/TIA-485 name	Description for IDv
1	8	required	RXD0	RXD0	A'	Receiver terminal 0, Va' Voltage (Va' > Vb' for binary 0 [ON] state)
2	4	required	RXD1	RXD1	B'	Receiver terminal 1, Vb' Voltage (Vb' > Va' for binary 1 [OFF] state)
3	3	optional	PMC	--	--	Port Mode Control
4	5	required	TXD1	TXD1	B	Generator terminal 1, Vb Voltage (Vb > Va for binary 1 [OFF] state)
5	9	required	TXD0	TXD0	A	Generator terminal 0, Va Voltage (Va > Vb for binary 0 [ON] state)
7	2	recommended	VP	--	--	Positive 5...24 V DC Power Supply
8	1	required	Common	Common	C/C'	Signal and Power Supply Common

Note : When both 2 and 4-Wire configurations are implemented on the same port, the **4W notations must be used.**

3.5.3 RJ45 and 9-pin D-shell Pinouts for optional RS232-MODBUS

If an RJ45 or a 9-pin D-shell connector is used for a RS232-MODBUS device, the pinouts hereafter must be respected for every implemented circuit.

DCE <u>Underlined</u> pins can be output			Circuit			DTE <u>Underlined</u> pins can be output		
Pin on RJ45	Pin on D9-shell	Level of requirement	Name	Description	RS232 Source	Level of requirement	Pin on RJ45	Pin on D9-shell
<u>1</u>	<u>2</u>	required	TXD	Transmitted Data	DTE	required	<u>2</u>	<u>3</u>
2	3	required	RXD	Received Data	DCE	required	1	2
3	7	optional	CTS	Clear to Send	DCE	optional	6	8
<u>6</u>	<u>8</u>	optional	RTS	Request to Send	DTE	optional	<u>3</u>	<u>7</u>
8	5	required	Common	Signal Common	--	required	8	5

Important Note : Some DCE Pinouts are crossed with DTE Pinouts with the same name :

A directly pin to pin wired cable (without any crossing) must be used between one DTE (a PC for example) and a DCE (a PLC for example).

3.6 Cables

A MODBUS over Serial Line Cable **must be shielded**. At one end of each cable its shield must be connected to protective ground. If a connector is used at this end, the shell of the connector is connected to the shield of the cable.

An RS485-MODBUS must use **a balanced pair** (for D0-D1) **and a third wire** (for the Common). In addition to that a second balanced pair must be used in a 4W-MODBUS system (for RXD0-RXD1).

If a connectorized 4 pairs Category 5 Cable is used, please remember to the user in the User Guides :

“Connection of a crossed cable in a 2-wire MODBUS system may cause damages”.

To minimize errors in cabling, a Color Code is recommended for the wires in the RS485-MODBUS Cables :

	Signal Names	Recommended Color
	D1-TXD1	yellow
	D0-TXD0	brown
	Common	grey
<i>4W (Optional)</i>	<i>RXD0</i>	<i>white</i>
<i>4W (Optional)</i>	<i>RXD1</i>	<i>blue</i>

Figure 28: Color code for RS485-MODBUS wires

Note : *Category 5 Cables use other colors.*

For RS485-MODBUS, Wire Gauge must be chosen sufficiently wide to permit the maximum length (1000 m). AWG 24 is always sufficient for the MODBUS Data.

Category 5 cables may operate for RS485-MODBUS, to a maximum length of 600m.

For the balanced pairs used in an RS485-system, a **Characteristic Impedance** with a value higher than 100 Ohms may be preferred, especially for 19200 and higher baud rates.

3.7 Visual Diagnosis

For a visual diagnosis, communication status and device status must be indicated by LEDs :

LED	Level of requirement	State	Recommended colour
Communication	required	Switched ON during frame reception or sending. (2 LEDs for frame reception and frame sending, or 1 LED for both purposes.)	Yellow
Error	recommended	Switched ON : internal fault Flashing : Other faults (Communication fault or configuration error)	Red
Device status	optional	Switched ON : device powered	Green

4 Installation and Documentation

4.1 Installation

Product vendor should pay attention to give to the user of a MODBUS System or MODBUS Devices all useful information to prevent them **from any error in cabling** or bad utilization of cabling accessories :

- Some other Fieldbuses, CANopen for example, use the same connector types (D-shell, RJ45...) .
- Studies are conducted on Ethernet, with power supply on the same Balanced Pairs Cable.
- Some Products use for I/O circuits the same connector types (D-shell, RJ45...).

On these connectors, for the most part, **no foolproofing is available** (polarizing notch or other implementation) .

4.2 User Guide

The User Guide of any MODBUS Device or Cabling System Component must include in a non exhaustive manner one or two types of information:

4.2.1 For any MODBUS Product :

The following information should be documented :

- All the implemented requests.
- The operating modes.
- The visual diagnostics.
- The reachable registers and supported function codes.
- Installation rules.

- The required information in the following sections should also be documented :
 - ⇒ "Two-Wire MODBUS Definition" (to mention the Required Circuits) ;
 - ⇒ "Optional Four-Wire MODBUS Definition" (to mention the Required Circuits) ;
 - ⇒ "Line Polarization" (to mention a possible Need or an Implementation) ;
 - ⇒ "Cables" (special care of crossed cables).

- **A specific indication** relating to the devices addresses, is to be written in the form of an important warning :

"It is of great importance to ensure at the time of the procedure of devices addressing, that there is not two devices with the same address. In such a case, an abnormal behavior of the whole serial bus can occur, the Master being then in the impossibility to communicate with all present slaves on the bus."

- A "**Getting Started**" chapter is highly recommended, with the documented description of a typical application example, for an easy start.

4.2.2 For a MODBUS Product with implemented Options :

The different optional parameters must be clearly detailed :

- ⇒ Optional serial Transmission mode ;
- ⇒ Optional Parity Checking ;
- ⇒ Optional Baud Rates ;
- ⇒ Optional Circuit(s) : Power Supply, Port Configuration ;
- ⇒ Optional Interface(s) ;
- ⇒ Maximum number of devices (without repeater) if greater than 32.

5 Implementation Classes

Each device on a MODBUS Serial Line must respect all the **mandatory** requirements of a same implementation class.

The following parameters are used to classify the MODBUS Serial Line devices :

- Addressing
- Broadcasting
- Transmission mode
- Baud rate
- Character format
- Electrical interface parameter

Two implementation classes are proposed, the Basic and the Regular classes.

The regular class must provide configuration capabilities.

	BASIC		REGULAR	Default value
Addressing	Slave : configurable address from 1 to 247	Master : to be able to address a slave from address 1 to 247	Same as Basic	-
Broadcast	Yes		Yes	-
Baud Rate	9600 (19200 is also recommended)		9600, 19200 + additional configurable baud rates	19200 (if implemented, else 9600)
Parity	EVEN		EVEN + possibility to configure NO and ODD parity	EVEN
Mode	RTU		RTU + ASCII	RTU
Electrical Interface	RS485 2W-cabling or RS232		RS485 2W-cabling (and 4W-cabling as an additional option) or RS232	RS485 2W-cabling
Connector Type	RJ 45 (recommended)			-

6 Appendix

6.1 Appendix A - Management of Serial Line Diagnostic Counters

6.1.1 General description

MODBUS Serial Line defines a list of diagnostic counters to allow performance and error management.

These counters are accessible using the MODBUS application protocol and its Diagnostic function (function code 08).

Each counter can be get by a sub-function code bound to the counter number. All counters can be cleared using the sub-function code 0x0A.

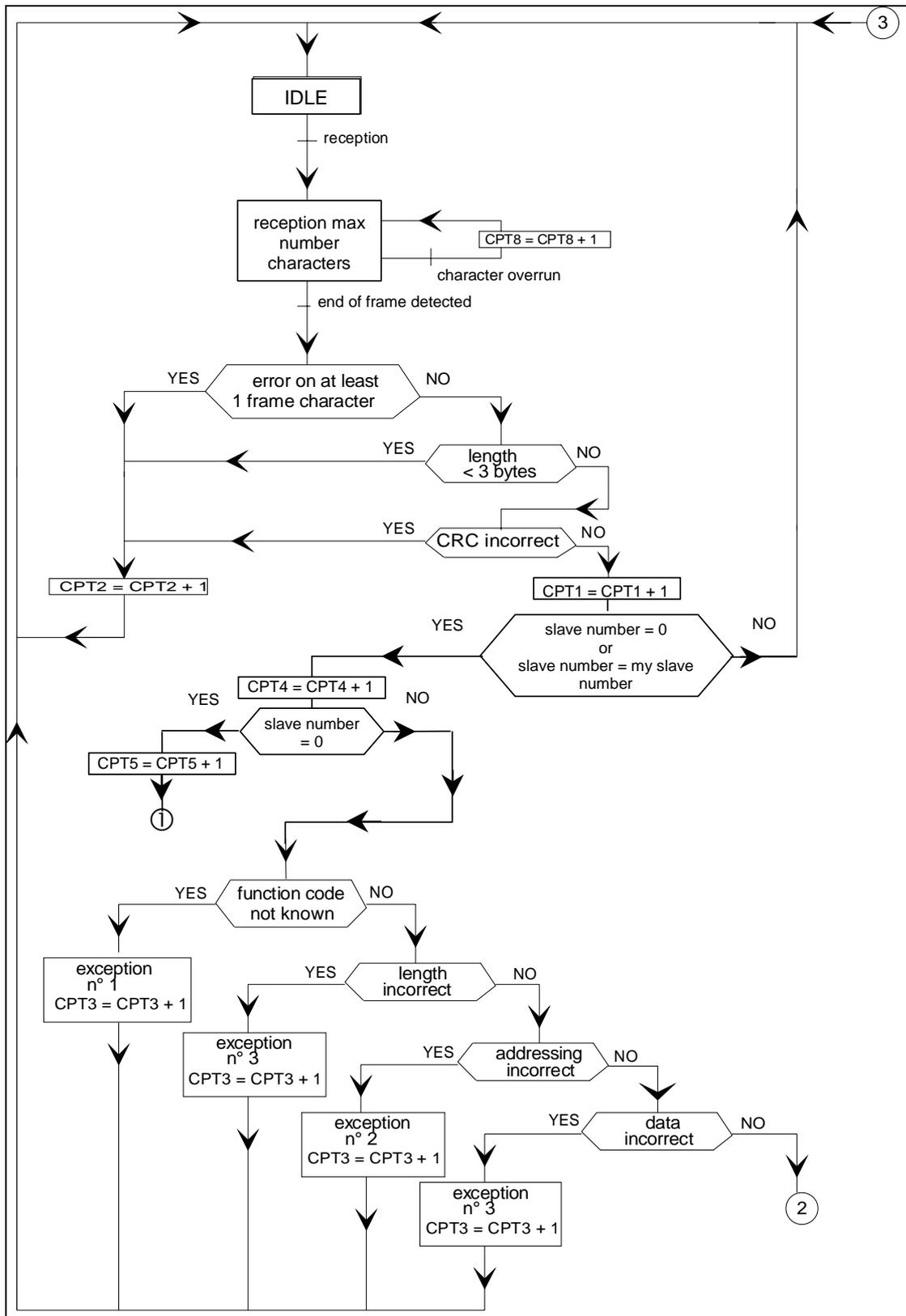
The format of the Diagnostic function is described in the MODBUS application protocol specification.

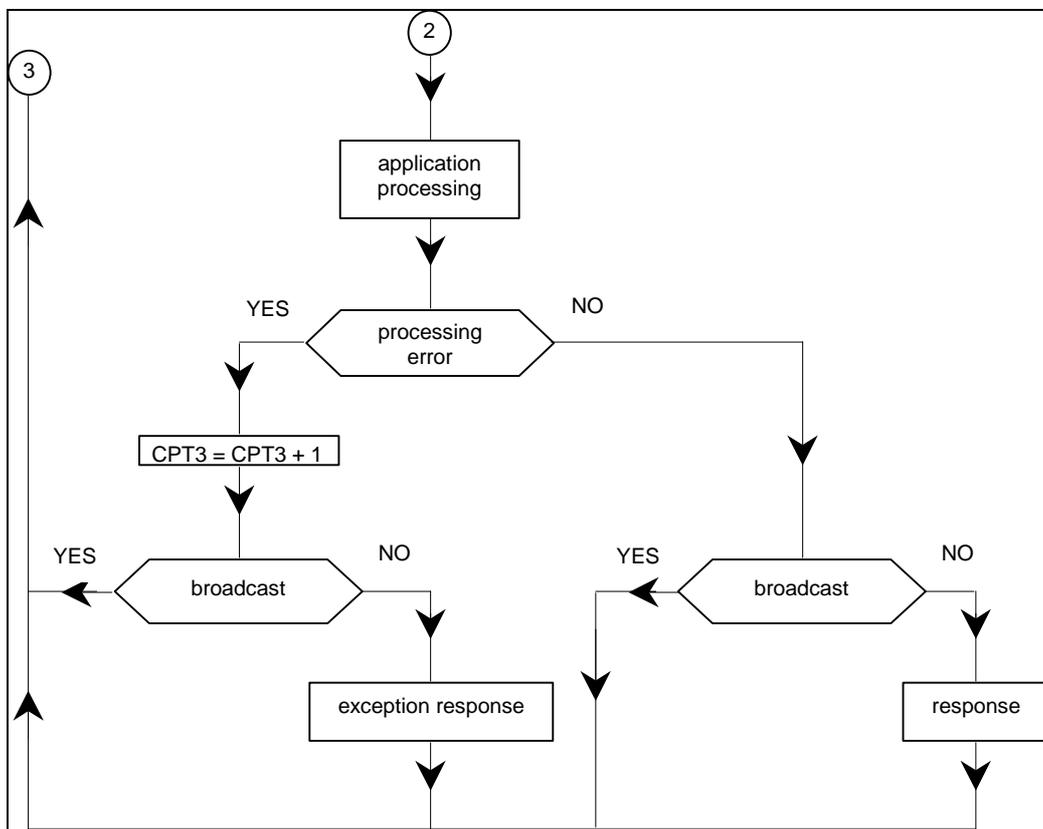
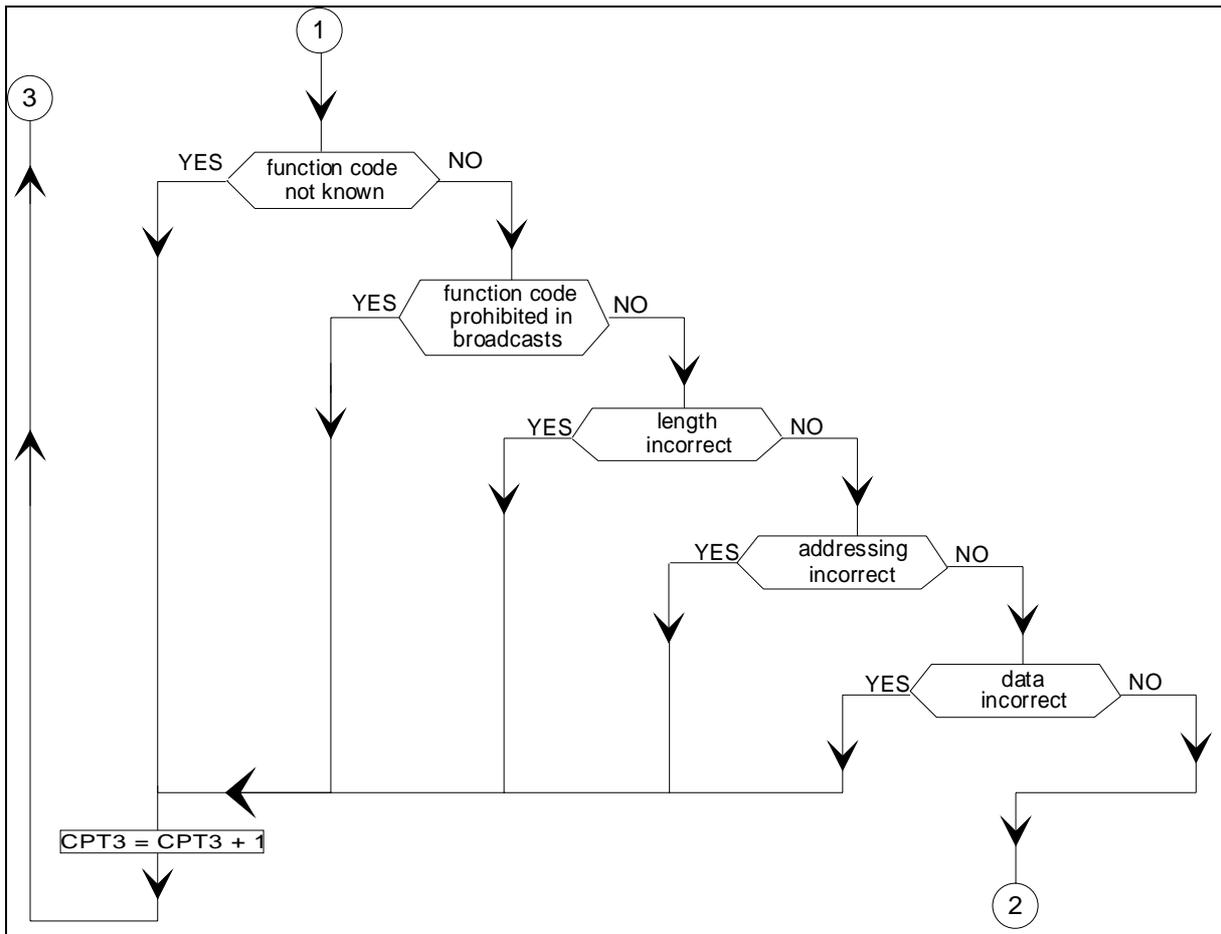
Herein is the list of diagnostics and associated sub-function codes supported by a serial line device.

Sub-function code	Counter number		Counters Name	Comments (for diagram below)
	Hex	Dec		
0x0B	1		Return Bus Message Count	Quantity of messages that the remote device has detected on the communications system since its last restart, clear counters operation, or power-up. Messages with bad CRC are not taken into account.
0x0C	2		Return Bus Communication Error Count	Quantity of CRC errors encountered by the remote device since its last restart, clear counters operation, or power-up. In case of an error detected on the character level, (overrun, parity error), or in case of a message length < 3 bytes, the receiving device is not able to calculate the CRC. In such cases, this counter is also incremented.
0x0D	3		Return Slave Exception Error Count	Quantity of MODBUS exception error detected by the remote device since its last restart, clear counters operation, or power-up. It comprises also the error detected in broadcast messages even if an exception message is not returned in this case. Exception errors are described and listed in "MODBUS Application Protocol Specification" document.
0x0E	4		Return Slave Message Count	Quantity of messages addressed to the remote device, including broadcast messages, that the remote device has processed since its last restart, clear counters operation, or power-up.
0x0F	5		Return Slave No Response Count	Quantity of messages received by the remote device for which it returned no response (neither a normal response nor an exception response), since its last restart, clear counters operation, or power-up. Then, this counter counts the number of broadcast messages it has received.
0x10	6		Return Slave NAK Count	Quantity of messages addressed to the remote device for which it returned a Negative Acknowledge (NAK) exception response, since its last restart, clear counters operation, or power-up. Exception responses are described and listed in "MODBUS Application Protocol Specification" document.
0x11	7		Return Slave Busy Count	Quantity of messages addressed to the remote device for which it returned a Slave Device Busy exception response, since its last restart, clear counters operation, or power-up. Exception responses are described and listed in "MODBUS Application Protocol Specification" document
0x12	8		Return Bus Character Overrun Count	Quantity of messages addressed to the remote device that it could not handle due to a character overrun condition, since its last restart, clear counters operation, or power-up. A character overrun is caused by data characters arriving at the port faster than they can be stored, or by the loss of a character due to a hardware malfunction.

6.1.2 Counters Management Diagram

The following diagrams describe when each previous counters must be incremented.





6.2 Appendix B - LRC/CRC Generation

6.2.1 LRC Generation

The Longitudinal Redundancy Checking (LRC) field is one byte, containing an 8-bit binary value. The LRC value is calculated by the transmitting device, which appends the LRC to the message. The device that receives recalculates an LRC during receipt of the message, and compares the calculated value to the actual value it received in the LRC field. If the two values are not equal, an error results.

The LRC is calculated by adding together successive 8-bit bytes in the message, discarding any carries, and then two's complementing the result. The LRC is an 8-bit field, therefore each new addition of a character that would result in a value higher than 255 decimal simply 'rolls over' the field's value through zero. Because there is no ninth bit, the carry is discarded automatically.

A procedure for generating an LRC is:

1. Add all bytes in the message, excluding the starting 'colon' and ending CRLF. Add them into an 8-bit field, so that carries will be discarded.
2. Subtract the final field value from FF hex (all 1's), to produce the ones-complement.
3. Add 1 to produce the twos-complement.

Placing the LRC into the Message

When the 8-bit LRC (2 ASCII characters) is transmitted in the message, the high-order character will be transmitted first, followed by the low-order character. For example, if the LRC value is 61 hex (0110 0001):

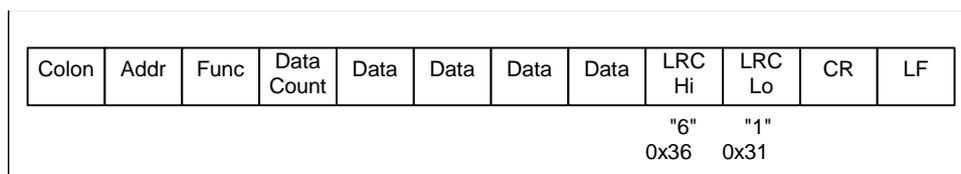


Figure 29: LRC Character Sequence

Example: an example of a C language function performing LRC generation is shown below.

The function takes two arguments:

- unsigned char *auchMsg; A pointer to the message buffer containing binary data to be used for generating the LRC,
- unsigned short usDataLen; The quantity of bytes in the message buffer.

LRC Generation Function

```
static unsigned char LRC(auchMsg, usDataLen) /* the function returns the LRC as a type unsigned char */
unsigned char *auchMsg ; /* message to calculate LRC upon */
unsigned short usDataLen ; /* quantity of bytes in message */
{
    unsigned char uchLRC = 0 ; /* LRC char initialized */
    while (usDataLen--) /* pass through message buffer */
        uchLRC += *auchMsg++ ; /* add buffer byte without carry */
    return ((unsigned char)(~((char)uchLRC))) ; /* return twos complement */
}
```

6.2.2 CRC Generation

The Cyclical Redundancy Checking (CRC) field is two bytes, containing a 16-bit binary value. The CRC value is calculated by the transmitting device, which appends the CRC to the message. The device that receives recalculates a CRC during receipt of the message, and compares the calculated value to the actual value it received in the CRC field. If the two values are not equal, an error results.

The CRC is started by first preloading a 16-bit register to all 1's. Then a process begins of applying successive 8-bit bytes of the message to the current contents of the register. Only the eight bits of data in each character are used for generating the CRC. Start and stop bits and the parity bit, do not apply to the CRC.

During generation of the CRC, each 8-bit character is exclusive ORed with the register contents. Then the result is shifted in the direction of the least significant bit (LSB), with a zero filled into the most significant bit (MSB) position. The LSB is extracted and examined. If the LSB was a 1, the register is then exclusive ORed with a preset, fixed value. If the LSB was a 0, no exclusive OR takes place.

This process is repeated until eight shifts have been performed. After the last (eighth) shift, the next 8-bit character is exclusive ORed with the register's current value, and the process repeats for eight more shifts as described above. The final content of the register, after all the characters of the message have been applied, is the CRC value.

A procedure for generating a CRC is:

1. Load a 16-bit register with FFFF hex (all 1's). Call this the CRC register.
2. Exclusive OR the first 8-bit byte of the message with the low-order byte of the 16-bit CRC register, putting the result in the CRC register.
3. Shift the CRC register one bit to the right (toward the LSB), zero-filling the MSB. Extract and examine the LSB.
4. (If the LSB was 0): Repeat Step 3 (another shift).
(If the LSB was 1): Exclusive OR the CRC register with the polynomial value 0xA001 (1010 0000 0000 0001).
5. Repeat Steps 3 and 4 until 8 shifts have been performed. When this is done, a complete 8-bit byte will have been processed.
6. Repeat Steps 2 through 5 for the next 8-bit byte of the message. Continue doing this until all bytes have been processed.
7. The final content of the CRC register is the CRC value.
8. When the CRC is placed into the message, its upper and lower bytes must be swapped as described below.

Placing the CRC into the Message

When the 16-bit CRC (two 8-bit bytes) is transmitted in the message, the low-order byte will be transmitted first, followed by the high-order byte.

For example, if the CRC value is 1241 hex (0001 0010 0100 0001):

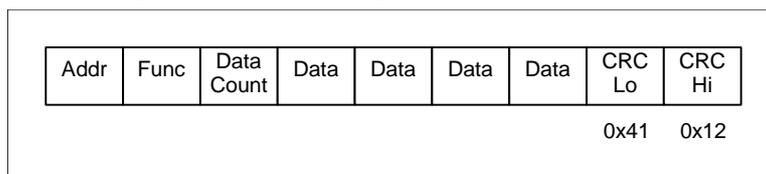
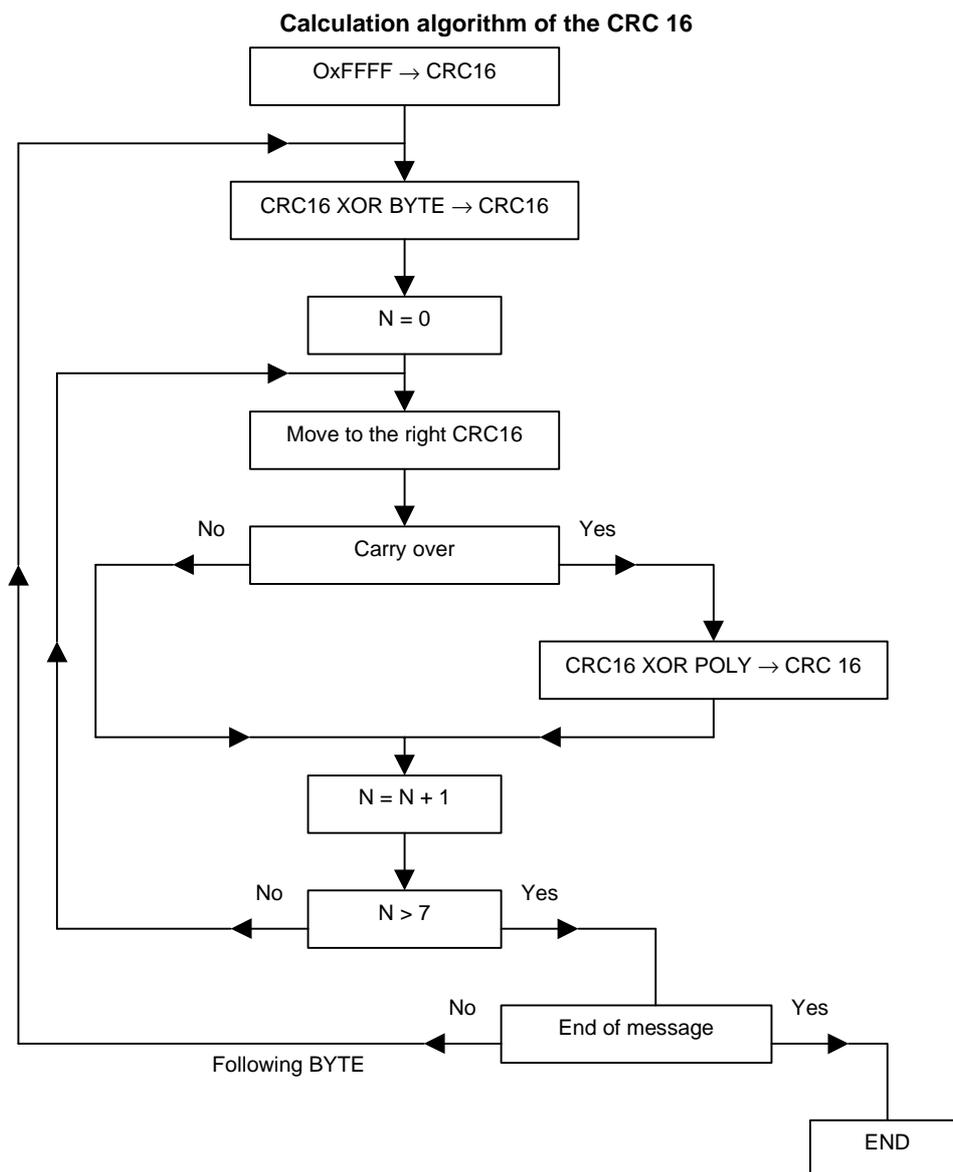


Figure 30: CRC Byte Sequence



XOR = exclusive or

N = number of information bits

POLY = calculation polynomial of the CRC 16 = 1010 0000 0000 0001

(Generating polynomial = $1 + x_2 + x_{15} + x_{16}$)

In the CRC 16, the 1st byte transmitted is the least significant one.

Example of CRC calculation (frame 02 07)

CRC register initialization		1111	1111	1111	1111
XOR 1st character		0000	0000	0000	0000
		<hr/>			
	Move 1	1111	1111	1111	1101
		0111	1111	1111	1110 1
		1010	0000	0000	0001
		<hr/>			
Flag to 1, XOR polynomial		1101	1111	1111	1111
	Move 2	0110	1111	1111	1111 1
Flag to 1, XOR polynomial		1010	0000	0000	0001
		<hr/>			
		1100	1111	1111	1110
	Move 3	0110	0111	1111	1110 0
	Move 4	0011	0011	1111	1111 1
		1010	0000	0000	0001
		<hr/>			
		1001	0011	1111	1110
	Move 5	0100	1001	1111	1111 0
	Move 6	0010	0100	1111	1111 1
		1010	0000	0000	0001
		<hr/>			
		1000	0100	1111	1110
	Move 7	0100	0010	0111	1111 0
	Move 8	0010	0001	0011	1111 0
		1010	0000	0000	0001
		<hr/>			
		1000	0001	0011	1110
		0000	0000	0000	0111
		<hr/>			
XOR 2nd character		1000	0001	0011	1001
	Move 1	0100	0000	1001	1100 1
		1010	0000	0000	0001
		<hr/>			
		1110	0000	1001	1101
	Move 2	0111	0000	0100	1110 1
		1010	0000	0000	0001
		<hr/>			
		1101	0000	0100	1111
	Move 3	0110	1000	0010	0111 1
		1010	0000	0000	0001
		<hr/>			
		1100	1000	0010	0110
	Move 4	0110	0100	0001	0011 0
	Move 5	0011	0010	0000	1001 1
		1010	0000	0000	0001
		<hr/>			
		1001	0010	0000	1000
	Move 6	0100	1001	0000	0100 0
	Move 7	0010	0100	1000	0010 0
	Move 8	0001	0010	0100	0001 0



The CRC 16 of the frame is then: 4112

Example

An example of a C language function performing CRC generation is shown on the following pages. All of the possible CRC values are preloaded into two arrays, which are simply indexed as the function increments through the message buffer. One array contains all of the 256 possible CRC values for the high byte of the 16-bit CRC field, and the other array contains all of the values for the low byte.

Indexing the CRC in this way provides faster execution than would be achieved by calculating a new CRC value with each new character from the message buffer.

Note: This function performs the swapping of the high/low CRC bytes internally. The bytes are already swapped in the CRC value that is returned from the function.

Therefore the CRC value returned from the function can be directly placed into the message for transmission.

The function takes two arguments:

unsigned char *puchMsg; A pointer to the message buffer containing binary data to be used for generating the CRC
 unsigned short usDataLen; The quantity of bytes in the message buffer.

CRC Generation Function

```

unsigned short CRC16 ( puchMsg, usDataLen )                     /* The function returns the CRC as a unsigned short type */
unsigned char *puchMsg ;                                         /* message to calculate CRC upon                             */
unsigned short usDataLen ;                                       /* quantity of bytes in message                                 */
{
    unsigned char uchCRChi = 0xFF ;                             /* high byte of CRC initialized                                 */
    unsigned char uchCRCLo = 0xFF ;                             /* low byte of CRC initialized                                   */
    unsigned ulIndex ;                                            /* will index into CRC lookup table                             */

    while (usDataLen--)                                           /* pass through message buffer                                   */
    {
        ulIndex = uchCRCLo ^ *puchMsgg++ ;                     /* calculate the CRC                                               */
        uchCRCLo = uchCRChi ^ auchCRChi[ulIndex] ;
        uchCRChi = auchCRCLo[ulIndex] ;
    }
    return (uchCRChi << 8 | uchCRCLo) ;
}
  
```

High-Order Byte Table

/* Table of CRC values for high-order byte */

```
static unsigned char auchCRCHi[] = {
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81,
    0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0,
    0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01,
    0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81,
    0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0,
    0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01,
    0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80,
    0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0,
    0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01,
    0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81,
    0x40,
    0x40
};
```

Low-Order Byte Table

/* Table of CRC values for low-order byte */

```
static char auchCRCLo[] = {
    0x00, 0xC0, 0xC1, 0x01, 0xC3, 0x03, 0x02, 0xC2, 0xC6, 0x06, 0x07, 0xC7, 0x05, 0xC5, 0xC4,
    0x04, 0xCC, 0x0C, 0x0D, 0xCD, 0x0F, 0xCF, 0xCE, 0x0E, 0x0A, 0xCA, 0xCB, 0x0B, 0xC9, 0x09,
    0x08, 0xC8, 0xD8, 0x18, 0x19, 0xD9, 0x1B, 0xDB, 0xDA, 0x1A, 0x1E, 0xDE, 0xDF, 0x1F, 0xDD,
    0x1D, 0x1C, 0xDC, 0x14, 0xD4, 0xD5, 0x15, 0xD7, 0x17, 0x16, 0xD6, 0xD2, 0x12, 0x13, 0xD3,
    0x11, 0xD1, 0xD0, 0x10, 0xF0, 0x30, 0x31, 0xF1, 0x33, 0xF3, 0xF2, 0x32, 0x36, 0xF6, 0xF7,
    0x37, 0xF5, 0x35, 0x34, 0xF4, 0x3C, 0xFC, 0xFD, 0x3D, 0xFF, 0x3F, 0x3E, 0xFE, 0xFA, 0x3A,
    0x3B, 0xFB, 0x39, 0xF9, 0xF8, 0x38, 0x28, 0xE8, 0xE9, 0x29, 0xEB, 0x2B, 0x2A, 0xEA, 0xEE,
    0x2E, 0x2F, 0xEF, 0x2D, 0xED, 0xEC, 0x2C, 0xE4, 0x24, 0x25, 0xE5, 0x27, 0xE7, 0xE6, 0x26,
    0x22, 0xE2, 0xE3, 0x23, 0xE1, 0x21, 0x20, 0xE0, 0xA0, 0x60, 0x61, 0xA1, 0x63, 0xA3, 0xA2,
    0x62, 0x66, 0xA6, 0xA7, 0x67, 0xA5, 0x65, 0x64, 0xA4, 0x6C, 0xAC, 0xAD, 0x6D, 0xAF, 0x6F,
    0x6E, 0xAE, 0xAA, 0x6A, 0x6B, 0xAB, 0x69, 0xA9, 0xA8, 0x68, 0x78, 0xB8, 0xB9, 0x79, 0xBB,
    0x7B, 0x7A, 0xBA, 0xBE, 0x7E, 0x7F, 0xBF, 0x7D, 0xBD, 0xBC, 0x7C, 0xB4, 0x74, 0x75, 0xB5,
    0x77, 0xB7, 0xB6, 0x76, 0x72, 0xB2, 0xB3, 0x73, 0xB1, 0x71, 0x70, 0xB0, 0x50, 0x90, 0x91,
    0x51, 0x93, 0x53, 0x52, 0x92, 0x96, 0x56, 0x57, 0x97, 0x55, 0x95, 0x94, 0x54, 0x9C, 0x5C,
    0x5D, 0x9D, 0x5F, 0x9F, 0x9E, 0x5E, 0x5A, 0x9A, 0x9B, 0x5B, 0x99, 0x59, 0x58, 0x98, 0x88,
    0x48, 0x49, 0x89, 0x4B, 0x8B, 0x8A, 0x4A, 0x4E, 0x8E, 0x8F, 0x4F, 0x8D, 0x4D, 0x4C, 0x8C,
    0x44, 0x84, 0x85, 0x45, 0x87, 0x47, 0x46, 0x86, 0x82, 0x42, 0x43, 0x83, 0x41, 0x81, 0x80,
    0x40
};
```

6.3 Appendix E - References

ANSI/ TIA/ EIA-232-F-1997	Interface Between Data Terminal Equipment and Data Circuit-Terminating Equipment Employing Serial Binary Data Interchange.
ANSI/ TIA/ EIA-485-A-1998	Electrical Characteristics of Generators and Receivers for Use in Balanced Digital Multipoint Systems.
MODBUS.org	MODBUS application protocol specification

IPS5451/IPS5451S

FULLY PROTECTED HIGH SIDE POWER MOSFET SWITCH

Features

- Over temperature protection (with auto-restart)
- Over current shutdown
- Active clamp
- E.S.D protection
- Status feedback
- Open load detection
- Logic ground isolated from power ground

Description

The IPS5451/IPS5451S are fully protected five terminal high side switch with built in short circuit, over-temperature, ESD protection, inductive load capability and diagnostic feedback. The over-current protection latches off the device if the output current exceeds $I_{shutdown}$. It can be reset by turning the input pin low. The over-temperature protection turns off the high side switches if the junction temperature exceeds $T_{shutdown}$. It will automatically restart after the junction has cooled 7°C below $T_{shutdown}$. A diagnostic pin is provided for status feedback of over-current, over-temperature and open load detection. The double level shifter circuitry allows large offsets between the logic ground and the load ground.

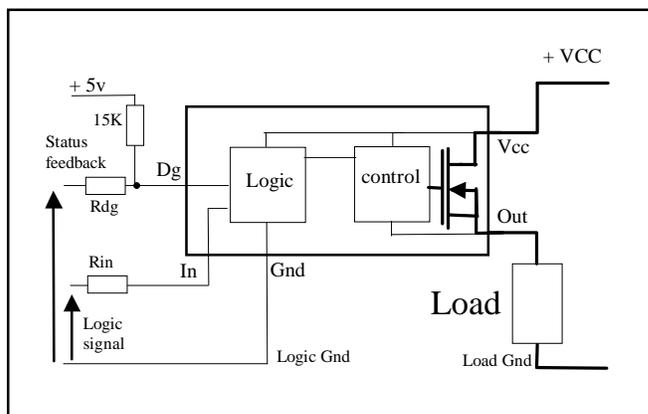
Product Summary

$R_{ds(on)}$	25m Ω (max)
V_{clamp}	50V
$I_{shutdown}$	35A
$I_{open\ load}$	1A

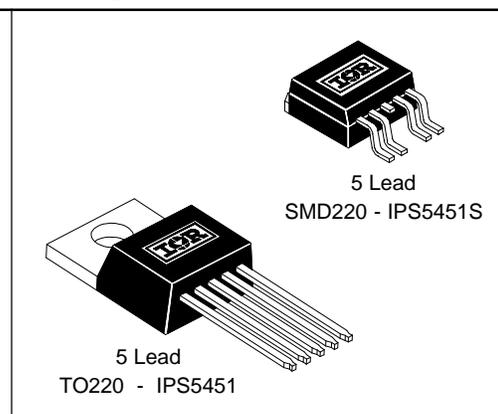
Truth Table

Op. Conditions	In	Out	Dg
Normal	H	H	H
Normal	L	L	H
Open load	H	H	L
Open load	L	X	H
Over current	H	L (latched)	L
Over current	L	L	H
Over-temperature	H	L (cycling)	L (cycling)
Over-temperature	L	L	H

Typical Connection



Packages



IPS5451/IPS5451S

Absolute Maximum Ratings

Absolute maximum ratings indicate sustained limits beyond which damage to the device may occur. All voltage parameters are referenced to GROUND lead. (T_{Ambient} = 25°C unless otherwise specified).

Symbol	Parameter	Min.	Max.	Units	Test Conditions
V _{out}	Maximum output voltage	V _{CC} -45	V _{CC} +0.3	V	
V _{offset}	Maximum logic ground to load ground offset	V _{CC} -45	V _{CC} +0.3		
V _{in}	Maximum Input voltage	-0.3	5.5		
I _{in, max}	Maximum IN current	-5	10	mA	
V _{dg}	Maximum diagnostic output voltage	-0.3	5.5	V	
I _{dg, max}	Maximum diagnostic output current	-1	10	mA	
I _{sd cont.}	Diode max. continuous current ⁽¹⁾ (r _{th} =62°C/W) IPS5451 (r _{th} =80°C/W) IPS5451S	—	2.8	A	
		—	2.2		
		—	45		
I _{sd pulsed}	Diode max. pulsed current ⁽¹⁾	—	45		
ESD1	Electrostatic discharge voltage (Human Body)	—	4	kV	C=100pF, R=1500Ω,
ESD2	Electrostatic discharge voltage (Machine Model)	—	0.5		C=200pF, R=0Ω, L=10μH
P _d	Maximum power dissipation ⁽¹⁾ (r _{th} =62°C/W) IPS5451 (r _{th} =80°C/W) IPS5451S	—	2	W	
		—	1.56		
		—	300		
T _{j max.}	Max. storage & operating junction temp.	-40	+150	°C	
T _{lead}	Lead temperature (soldering 10 seconds)	—	300		
V _{cc max.}	Maximum V _{cc} voltage	—	45	V	

Thermal Characteristics

Symbol	Parameter	Min.	Typ.	Max.	Units	Test Conditions
R _{th 1}	Thermal resistance junction to case	—	2	—	°C/W	TO-220
R _{th 2}	Thermal resistance junction to ambient	—	55	—		
R _{th 1}	Thermal resistance with standard footprint	—	60	—		D ² PAK (SMD220)
R _{th 2}	Thermal resistance with 1" square footprint	—	35	—		
R _{th 3}	Thermal resistance junction to case	—	5	—		

(1) Limited by junction temperature (pulsed current limited also by internal wiring)

Recommended Operating Conditions

These values are given for a quick design. For operation outside these conditions, please consult the application notes.

Symbol	Parameter	Min.	Max.	Units
V _{CC}	Continuous V _{CC} voltage	5.5	18	V
V _{IH}	High level input voltage	4	5.5	
V _{IL 1}	Low level input voltage	-0.3	0.9	
I _{out}	Continuous output current (T _{Ambient} = 85°C, T _J = 125°C, R _{th} = 62°C/W) IPS5451	—	4	A
	(T _{Ambient} = 85°C, T _J = 125°C, R _{th} = 80°C/W) IPS5451S	—	3.5	
I _{out} T _C =85°C	Continuous output current (T _{Case} = 85°C, I _N = 5V, T _J = 125°C, R _{th} = 5°C/W)	—	14	
R _{in}	Recommended resistor in series with IN pin	4	6	kΩ
R _{dg}	Recommended resistor in series with DG pin	10	20	

Static Electrical Characteristics

(T_J = 25°C, V_{CC} = 14V unless otherwise specified.)

Symbol	Parameter	Min.	Typ.	Max.	Units	Test Conditions
R _{ds(on)} @T _J =25°C	ON state resistance T _J = 25°C	—	19	25	mΩ	V _{in} = 5V, I _{out} = 14A
R _{ds(on)} (V _{CC} =6V)	ON state resistance @ V _{CC} = 6V	—	22	30		V _{in} = 5V, I _{out} = 7A
R _{ds(on)} @T _J =150°C	ON state resistance T _J = 150°C	—	32	—		V _{in} = 5V, I _{out} = 14A
V _{CC oper.}	Functional operating range	5.5	—	18	V	
V _{clamp 1}	V _{CC} to OUT clamp voltage 1	45	49	—		I _d = 10mA (see Fig.1 & 2)
V _{clamp 2}	V _{CC} to OUT clamp voltage 2	—	50	60		I _d = I _{shutdown} (see Fig.1 & 2)
V _f	Body diode forward voltage	—	0.9	1.2		I _d = 14A, V _{in} = 0V
I _{out} leakage	Output leakage current	—	10	50	μA	V _{out} = 0V, T _J = 25°C
I _{CC off}	Supply current when OFF	—	10	50		V _{in} = 0V, V _{out} = 0V
I _{CC on}	Supply current when ON	—	3.5	10	mA	V _{in} = 5V
I _{CC ac}	Ripple current when ON (AC RMS)	—	20	—	μA	V _{in} = 5V
V _{dgl}	Low level diagnostic output voltage	—	0.1	0.4	V	I _{dg} = 1.6 mA
I _{dg} leakage	Diagnostic output leakage current	—	1.5	10	μA	V _{dg} = 4.5V
V _{ih}	IN high threshold voltage	—	2.7	3.4	V	
V _{il}	IN low threshold voltage	1	2.0	—		
I _{in, on}	ON state IN positive current	—	30	80	μA	V _{in} = 4V
V _{CCUV+}	V _{CC} UVLO positive going threshold	—	4.7	5.5	V	
V _{CCUV-}	V _{CC} UVLO negative going threshold	3.0	4.4	—		
I _{nhyst.}	Input hysteresis	0.2	0.6	1.5		

IPS5451/IPS5451S

International
IR Rectifier

Switching Electrical Characteristics

$V_{CC} = 14V$, Resistive Load = 1Ω , $T_j = 25^\circ C$, (unless otherwise specified).

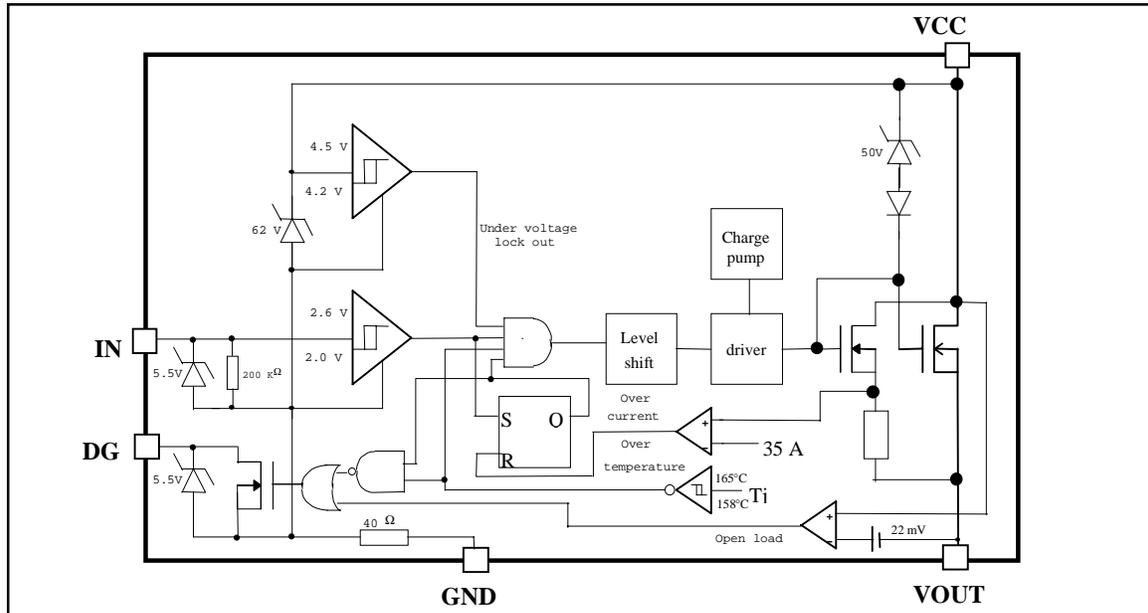
Symbol	Parameter	Min.	Typ.	Max.	Units	Test Conditions
T_{don}	Turn-on delay time	—	5	20	μs	See figure 3
T_{r1}	Rise time to $V_{out} = V_{CC} - 5V$	—	4	20		
T_{r2}	Rise time from the end of T_{r1} to $V_{out} = 90\%$ of V_{CC}	—	65	150		
dV/dt (on)	Turn ON dV/dt	—	3	6	$V/\mu s$	
E_{on}	Turn ON energy	—	3	—	mJ	See figure 4
T_{doff}	Turn-off delay time	—	65	150	μs	
T_f	Fall time to $V_{out} = 10\%$ of V_{CC}	—	8	20		
dV/dt (off)	Turn OFF dV/dt	—	5	10	$V/\mu s$	
E_{off}	Turn OFF energy	—	0.75	—	mJ	

Protection Characteristics

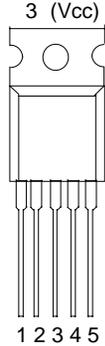
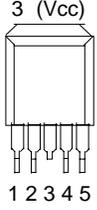
Symbol	Parameter	Min.	Typ.	Max.	Units	Test Conditions
T_{sd+}	Over-temp. positive going threshold	—	165	—	$^\circ C$	See fig. 2
T_{sd-}	Over-temp. negative going threshold	—	158	—	$^\circ C$	See fig. 2
I_{sd}	Over-current threshold	22	35	50	A	See fig. 2
$I_{open\ load}$	Open load detection threshold	0.3	1	2	A	
T_{reset}	Minimum time to reset protections	—	50	—	μs	$V_{in} = 0V$
T_{dg}	Blanking time before considering D_g	—	7	100	μs	Part turned on with $V_{in} = 5V$

Functional Block Diagram

All values are typical



Lead Assignments

 <p>3 (Vcc)</p> <p>1 2 3 4 5</p> <p>5 Lead - TO220</p>	<p>1 - Ground 2 - In 3 - Vcc 4 - DG 5 - Out</p>	 <p>3 (Vcc)</p> <p>1 2 3 4 5</p> <p>5 Lead - D²PAK (SMD220)</p>
IPS5451		IPS5451S
Part Number		

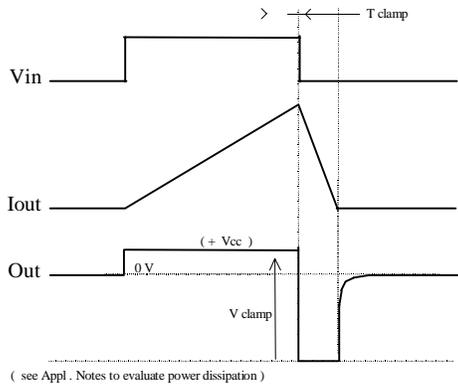


Figure 1 - Active clamp waveforms

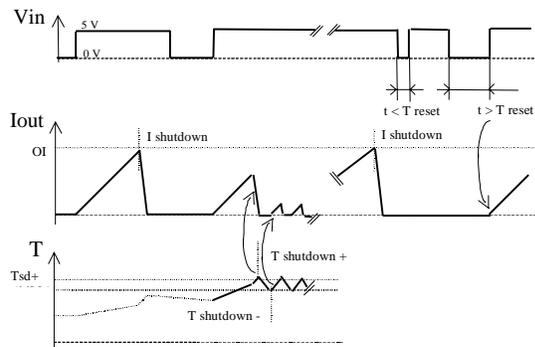


Figure 2 - Protection timing diagram

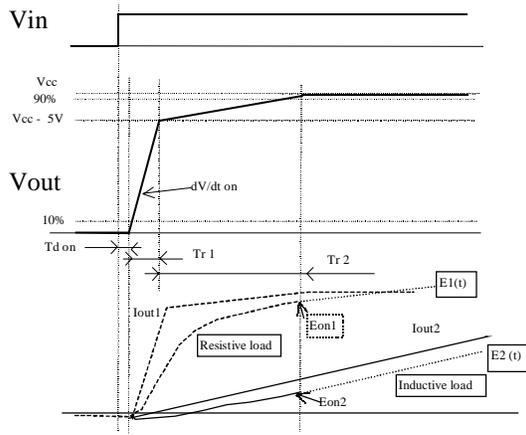


Figure 3 - Switching times definition (turn-on)
Turn on energy with a resistive or an inductive load

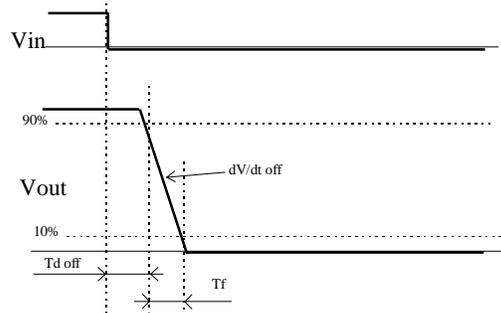


Figure 4 - Switching times definition (turn-off)

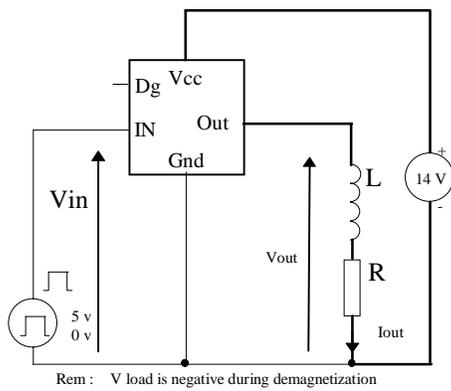


Figure 5 - Active clamp test circuit

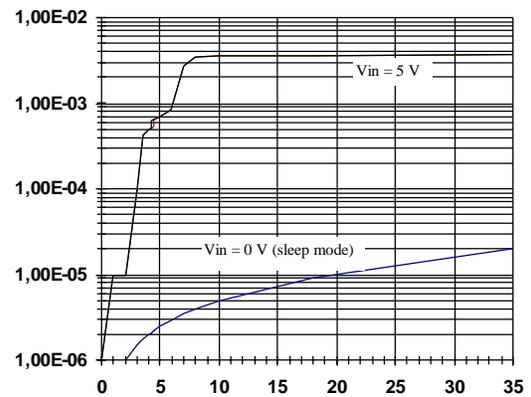


Figure 6 - Icc (mA) Vs Vcc (V)

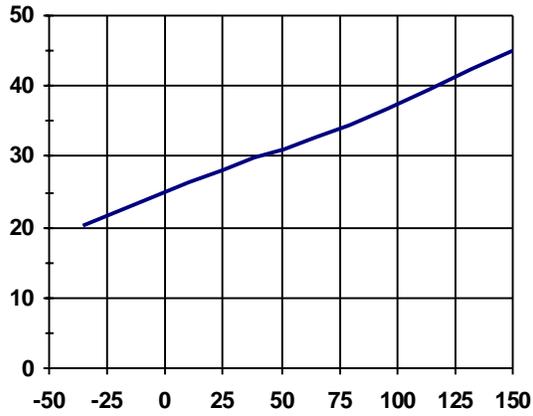


Figure 7 - Iin (μA) Vs Tj (°C)

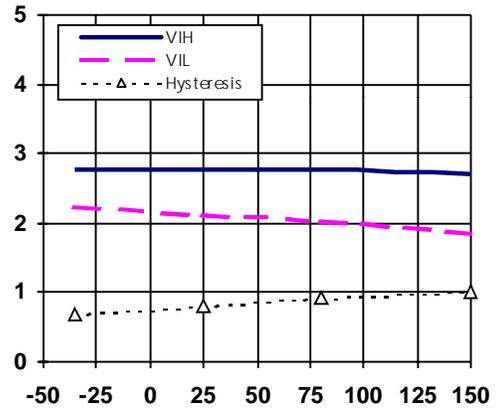


Figure 8 - VIH, VIL threshold (V) Vs Tj (°C)

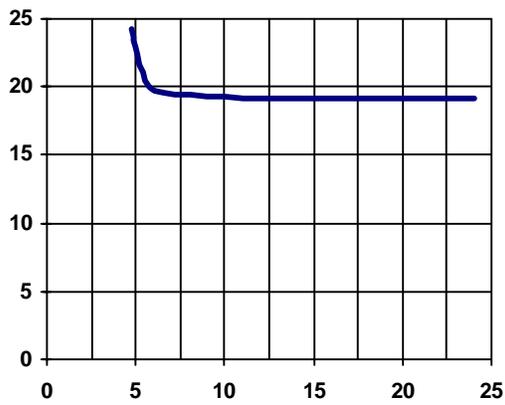


Figure 9 - Rdson (mΩ) vs Vcc (V)

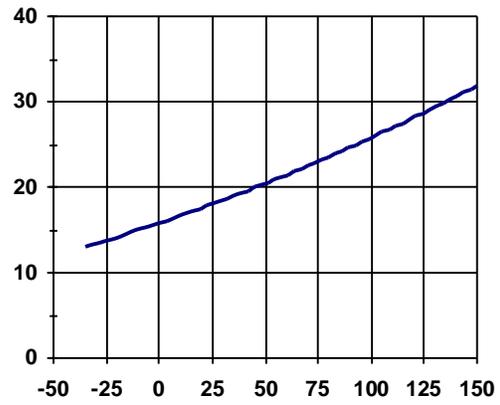


Figure 10 - Rdson (mΩ) vs Tj (°C)

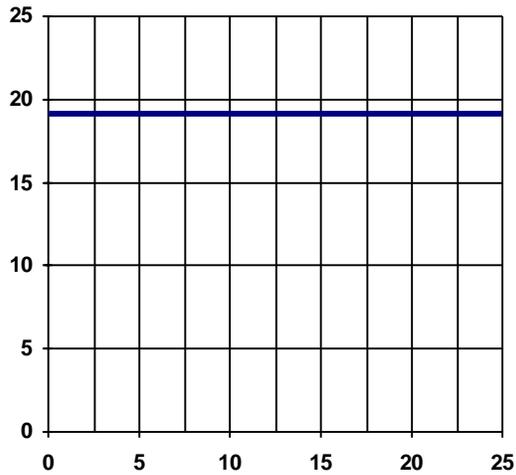


Figure 11 - Rdson (mΩ) vs Iout (A)

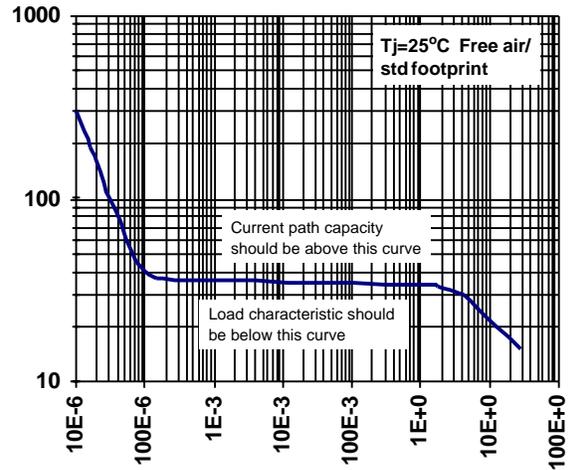


Figure 12 - Isd (A) vs Time (S)

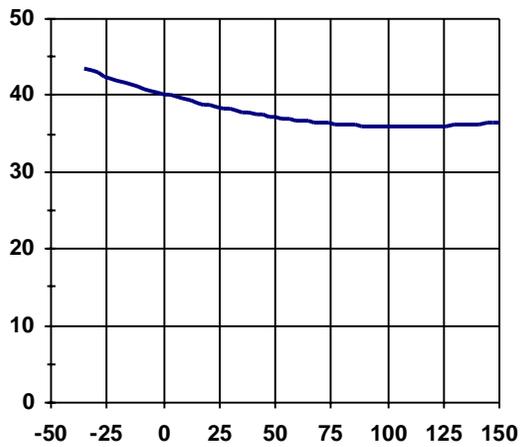


Figure 13 - Isd (A) vs Tj (°C)

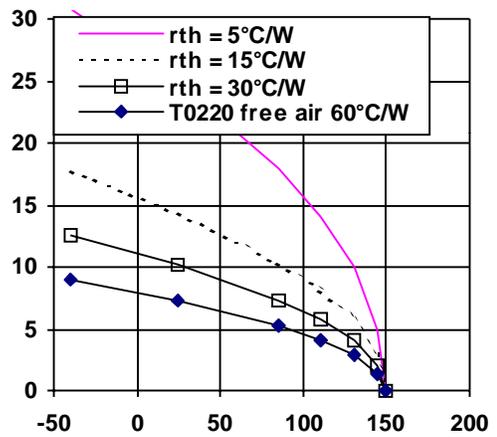


Figure 14 - Max. Cont. Ids (A) Vs
Amb. Temperature (°C)

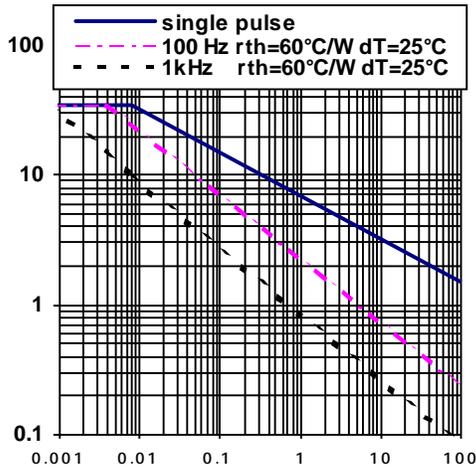


Figure 15 -Max. I clamp (A) Vs Inductive Load (m H)

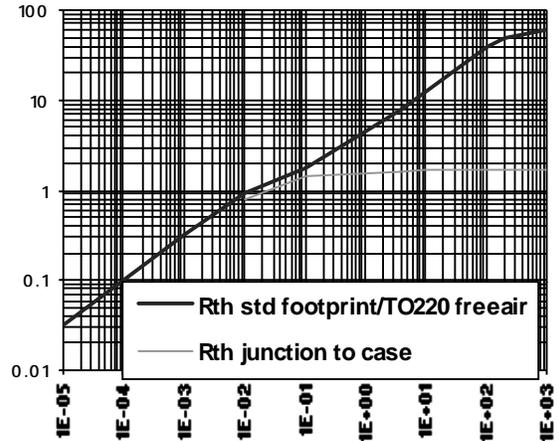


Figure 16 - Transient Rth (°C/W) Vs Time (s)

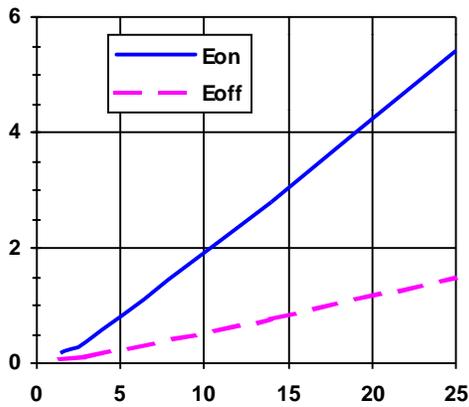


Figure 17 - Eon, Eoff (mJ) vs Iout (A)

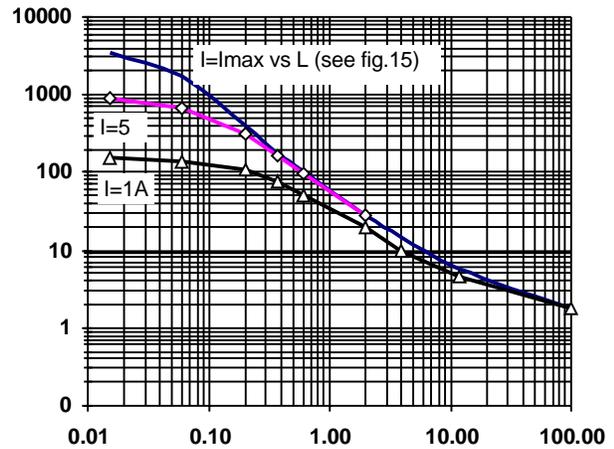


Figure 18 - Eon @ Vcc=14V (mJ) vs Inductance (mH)

IPS5451/IPS5451S

International
IR Rectifier

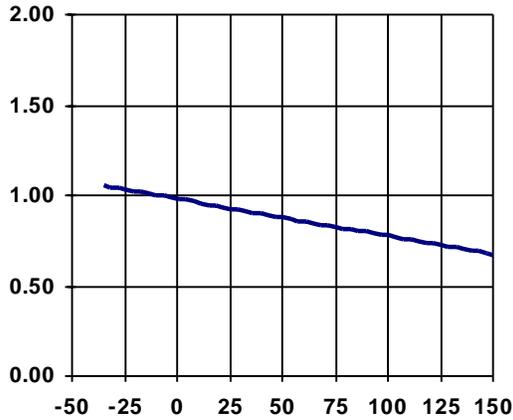


Figure 19 - I open load (A) vs Tj (°C)

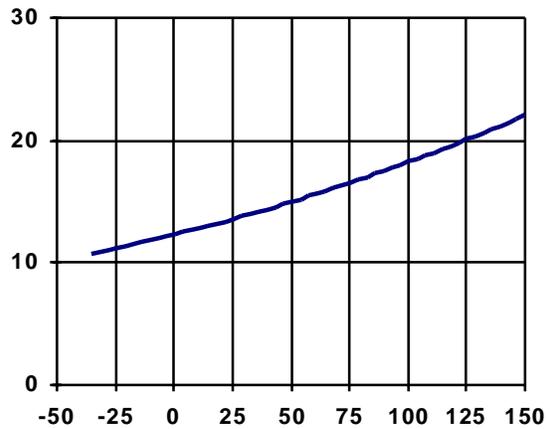
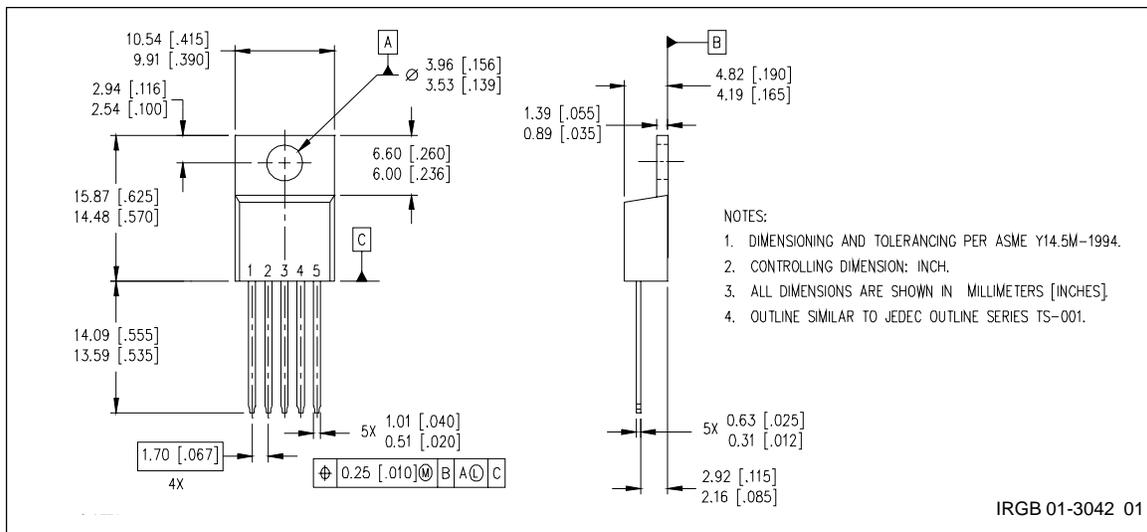
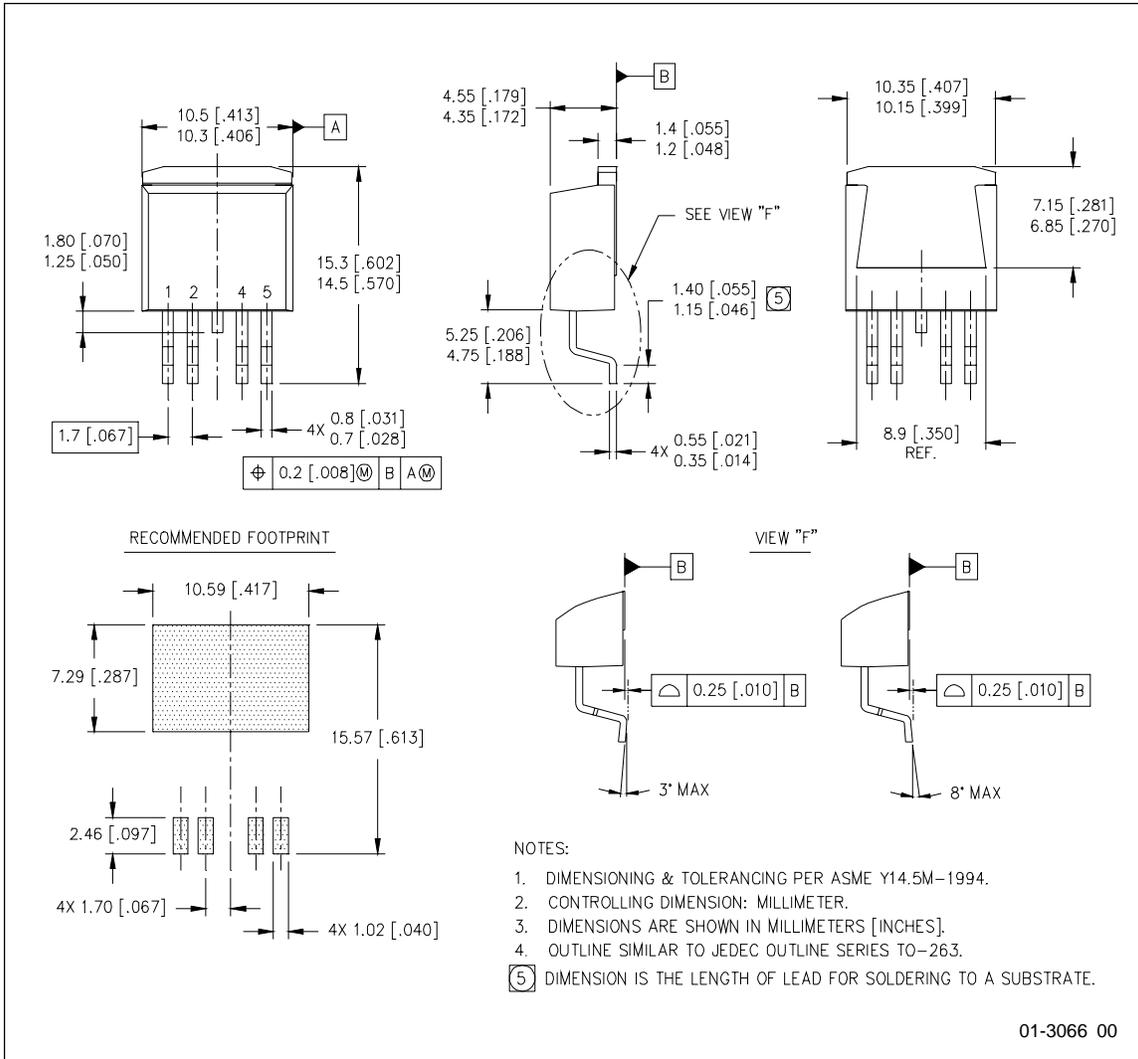


Figure 20 - Icc off (µA) vs Tj (°C)

Case Outline - TO220 (5 lead)



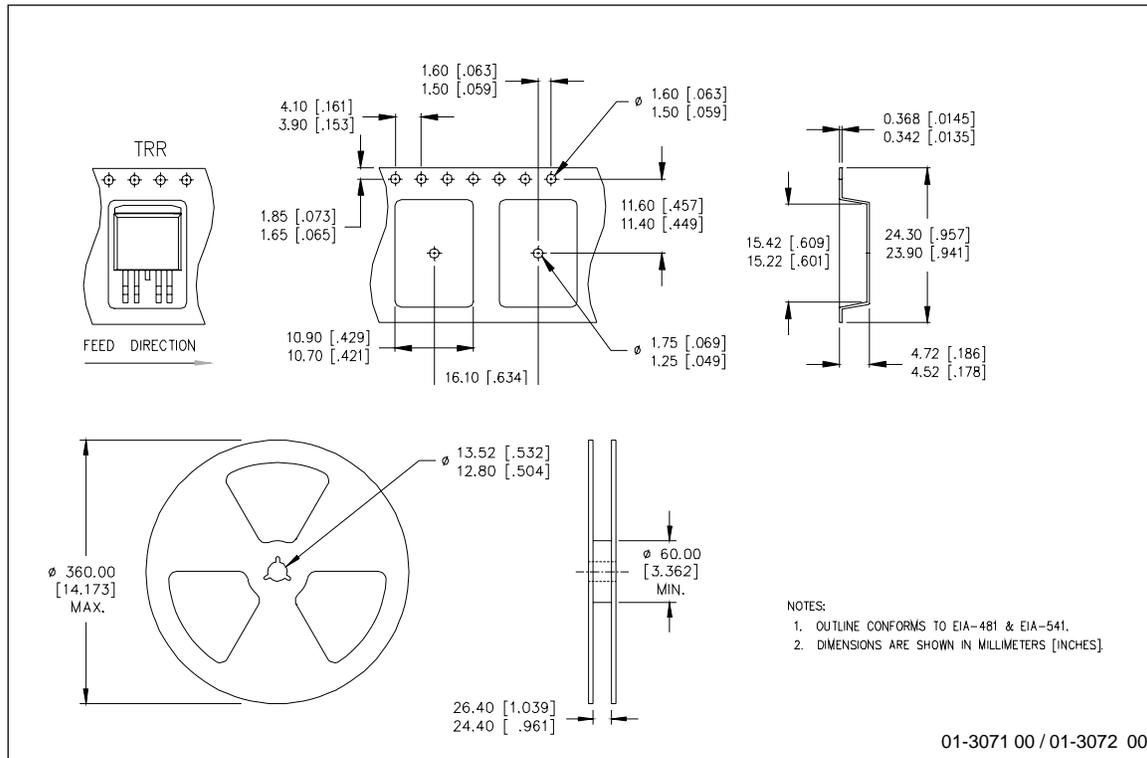
Case Outline - D²PAK (SMD220) - 5 Lead



IPS5451/IPS5451S

International
IR Rectifier

Tape & Reel - D²PAK (SMD220) - 5 Lead



International
IR Rectifier

IR WORLD HEADQUARTERS: 233 Kansas St., El Segundo, California 90245 Tel: (310) 252-7105
IR EUROPEAN REGIONAL CENTRE: 439/445 Godstone Rd., Whyteleafe, Surrey CR3 0BL, United Kingdom
 Tel: ++ 44 (0) 20 8645 8000
IR JAPAN: K&H Bldg., 2F, 30-4 Nishi-Ikebukuro 3-Chome, Toshima-Ku, Tokyo, Japan 171-0021 Tel: 8133 983 0086
IR HONG KONG: Unit 308, #F, New East Ocean Centre, No. 9 Science Museum Road, Tsimshatsui East, Kowloon
 Hong Kong Tel: (852) 2803-7380
Data and specifications subject to change without notice. 8/7/2000

FULLY PROTECTED POWER MOSFET SWITCH

Features

- Over temperature shutdown
- Over current shutdown
- Active clamp
- Low current & logic level input
- E.S.D protection

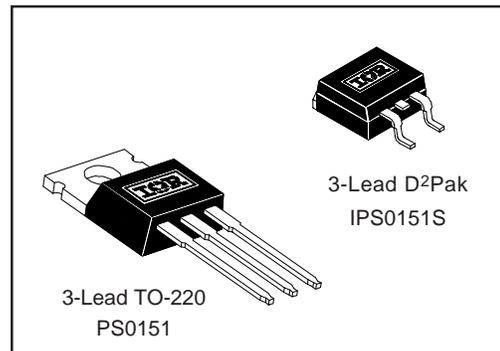
Description

The IPS0151/IPS0151S are fully protected three terminal SMART POWER MOSFETs that feature over-current, over-temperature, ESD protection and drain to source active clamp. These devices combine a HEXFET® POWER MOSFET and a gate driver. They offer full protection and high reliability required in harsh environments. The driver allows short switching times and provides efficient protection by turning OFF the power MOSFET when the temperature exceeds 165°C or when the drain current reaches 35A. The device restarts once the input is cycled. The avalanche capability is significantly enhanced by the active clamp and covers most inductive load demagnetizations.

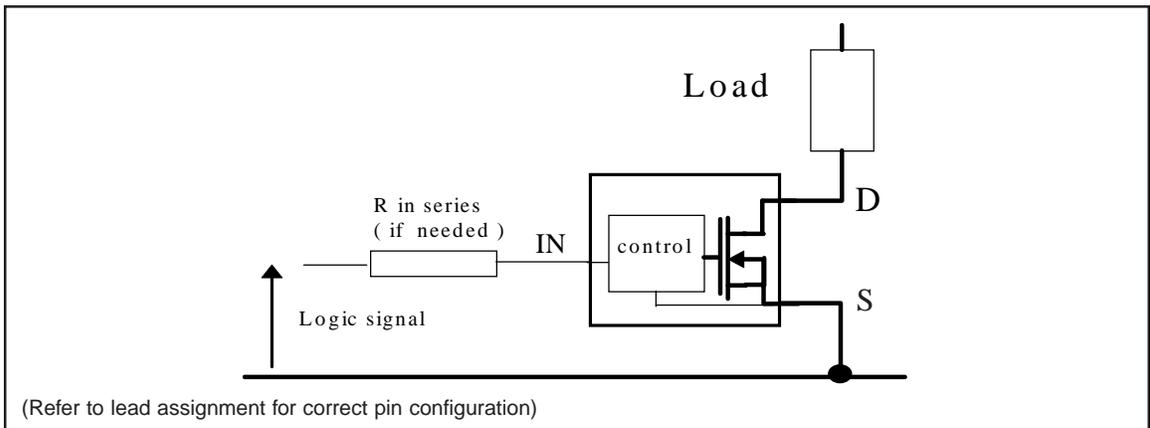
Product Summary

$R_{ds(on)}$	25mΩ (max)
V_{clamp}	50V
$I_{shutdown}$	35A
T_{on}/T_{off}	1.5μs

Packages



Typical Connection



IPS0151(S)

Absolute Maximum Ratings

Absolute maximum ratings indicate sustained limits beyond which damage to the device may occur. All voltage parameters are referenced to SOURCE lead. (T_{Ambient} = 25°C unless otherwise specified). PCB mounting uses the standard footprint with 70 µm copper thickness.

Symbol	Parameter	Min.	Max.	Units	Test Conditions
V _{ds}	Maximum drain to source voltage	—	47	V	
V _{in}	Maximum Input voltage	-0.3	7		
I _{in, max}	Maximum IN current	-10	+10	mA	
I _{sd cont.}	Diode max. continuous current ⁽¹⁾ rth=62°C/W IPS0151 rth=5°C/W IPS015135 rth=80°C/W IPS0151S	—	2.8	A	TO220 free air
		—	35		TO220 with Rth=5°C/W
		—	2.2		SMD220 Std footprint
I _{sd pulsed}	Diode max. pulsed current ⁽¹⁾	—	45		
P _d	Maximum power dissipation ⁽¹⁾ (rth=62°C/W) IPS0151 (rth=80°C/W) IPS0151S	—	2	W	
		—	1.56		
ESD1	Electrostatic discharge voltage (Human Body)	—	4	kV	C=100pF, R=1500Ω,
ESD2	Electrostatic discharge voltage (Machine Model)	—	0.5		C=200pF, R=0Ω, L=10µH
T _{stor.}	Max. storage temperature	-55	150	°C	
T _{j max.}	Max. junction temperature	-40	+150		
T _{lead}	Lead temperature (soldering, 10 seconds)	—	300		

Thermal Characteristics

Symbol	Parameter	Min.	Typ.	Max.	Units	Test Conditions
R _{th 1}	Thermal resistance free air	—	55	—	°C/W	TO-220
R _{th 2}	Thermal resistance junction to case	—	2	—		
R _{th 1}	Thermal resistance with standard footprint	—	60	—		D ² PAK (SMD220)
R _{th 2}	Thermal resistance with 1" square footprint	—	35	—		
R _{th 3}	Thermal resistance junction to case	—	2	—		

Recommended Operating Conditions

These values are given for a quick design. For operation outside these conditions, please consult the application notes.

Symbol	Parameter	Min.	Max.	Units
V _{ds (max)}	Continuous drain to source voltage	—	35	V
V _{IH}	High level input voltage	4	6	
V _{IL}	Low level input voltage	0	0.5	
I _{ds} T _{amb} =85°C	Continuous drain current (T _{Ambient} = 85°C, I _N = 5V, rth = 60°C/W, T _j = 125°C) IPS0151 (T _{Ambient} = 85°C, I _N = 5V, rth = 80°C/W, T _j = 125°C) IPS0151S	—	4.3	A
		—	3.8	
R _{in}	Recommended resistor in series with IN pin	0.2	5	kΩ
T _{r-in (max)}	Max recommended rise time for IN signal (see fig. 2)	—	1	µS
F _{r-Isc} ⁽²⁾	Max. frequency in short circuit condition (V _{cc} = 14V)	0	1	kHz

(1) Limited by junction temperature (pulsed current limited also by internal wiring)

(2) Operations at higher switching frequencies is possible. See Application Notes.

Static Electrical Characteristics

(T_j = 25°C unless otherwise specified.)

Symbol	Parameter	Min.	Typ.	Max.	Units	Test Conditions	
R _{ds(on)}	ON state resistance	T _j = 25°C	10	20	25	mΩ	V _{in} = 5V, I _{ds} = 1A
		T _j = 150°C	—	35	45		
I _{dss1} @T _j =25°C	Drain to source leakage current	0	0.5	25	μA	V _{CC} = 14V, T _j = 25°C	
I _{dss2} @T _j =25°C	Drain to source leakage current	0	5	50		V _{CC} = 40V, T _j = 25°C	
V _{clamp 1}	Drain to source clamp voltage 1	47	52	56	V	I _d = 20mA (see Fig.3 & 4)	
V _{clamp 2}	Drain to source clamp voltage 2	50	55	60		I _d =I _{shutdown} (see Fig.3 & 4)	
V _{in clamp}	IN to source clamp voltage	7	8.1	9.5		I _{in} = 1 mA	
V _{in th}	IN threshold voltage	1	1.6	2		I _d = 50mA, V _{ds} = 14V	
I _{in, -on}	ON state IN positive current	25	90	200	μA	V _{in} = 5V	
I _{in, -off}	OFF state IN positive current	50	130	250		V _{in} = 5V over-current triggered	

Switching Electrical Characteristics

V_{CC} = 14V, Resistive Load = 3Ω, Rin_{put} = 50Ω, 100μs pulse, T_j = 25°C, (unless otherwise specified).

Symbol	Parameter	Min.	Typ.	Max.	Units	Test Conditions
T _{on}	Turn-on delay time	0.05	0.25	0.6	μs	See figure 2
T _r	Rise time	0.2	0.9	1.5		
T _{rf}	Time to (final R _{ds(on)} 1.3%)	—	3.8	—		
T _{off}	Turn-off delay time	0.8	1.5	2		See figure 2
T _f	Fall time	0.4	1.1	2		
Q _{in}	Total gate charge	—	30	—	nC	V _{in} = 5V

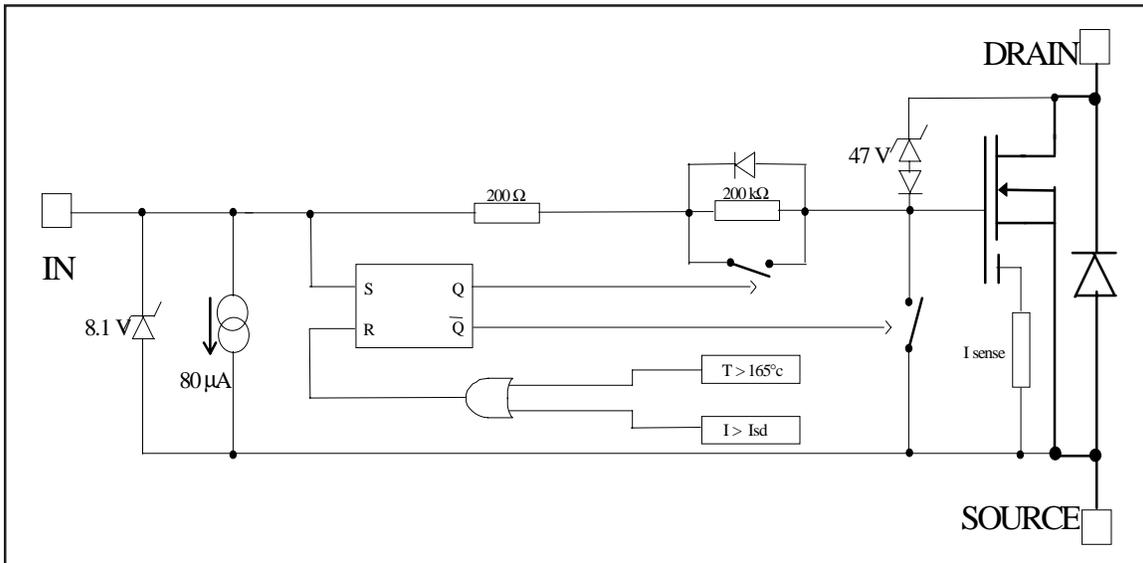
Protection Characteristics

Symbol	Parameter	Min.	Typ.	Max.	Units	Test Conditions
T _{sd}	Over temperature threshold	—	165	—	°C	See fig. 1
I _{sd}	Over current threshold	20	35	50	A	See fig. 1
V _{reset}	IN protection reset threshold	1.5	2.3	3	V	
T _{reset}	Time to reset protection	2	10	40	μs	V _{in} = 0V, T _j = 25°C
EOI_OT	Short circuit energy (see application note)	—	400	—	μJ	V _{CC} = 14V

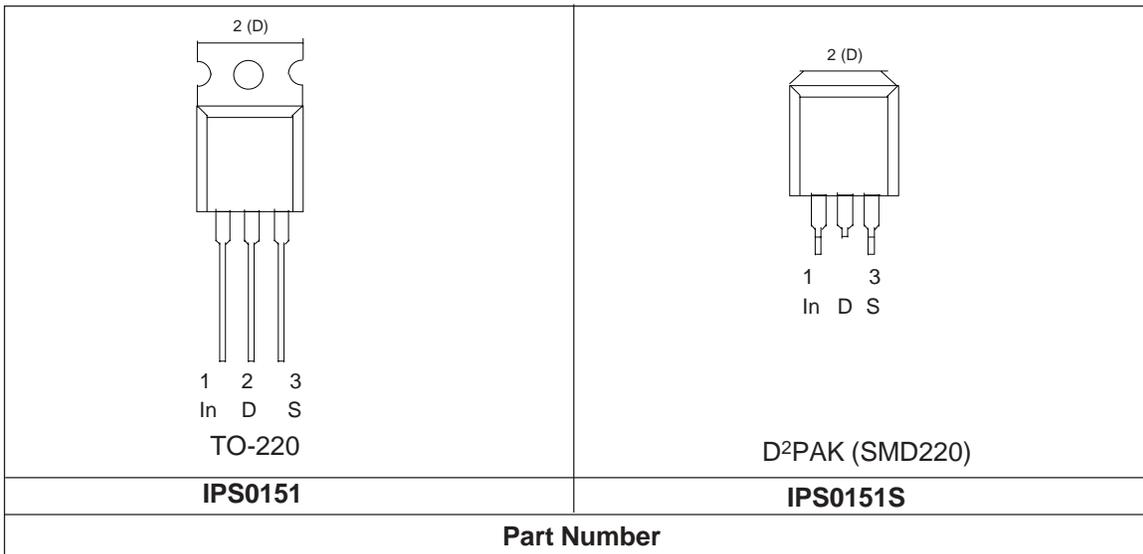
IPS0151(S)

Functional Block Diagram

All values are typical



Lead Assignments



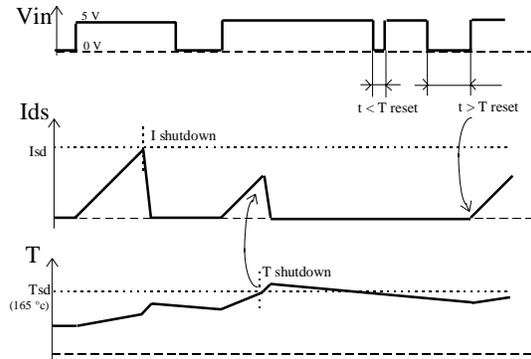


Figure 1 - Timing diagram

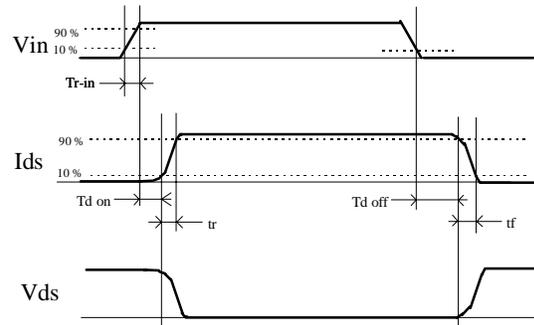


Figure 2 - IN rise time & switching time definitions

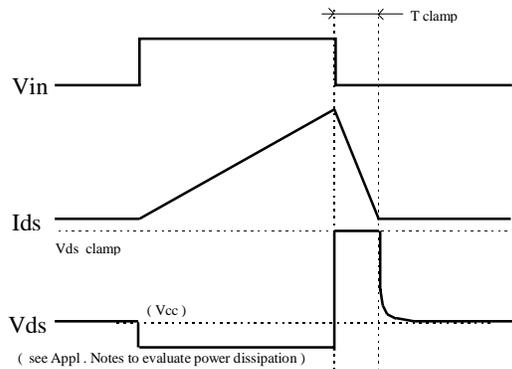


Figure 3 - Active clamp waveforms

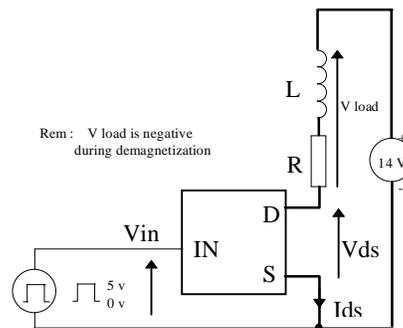


Figure 4 - Active clamp test circuit

All curves are typical values with standard footprints. Operating in the shaded area is not recommended.

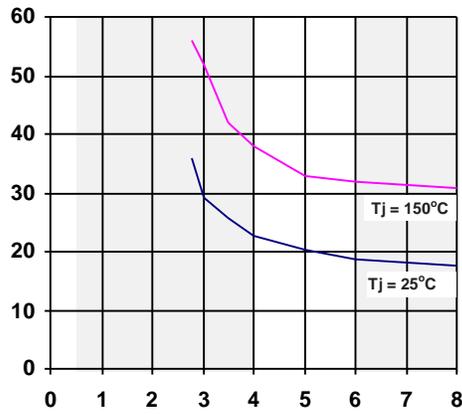


Figure 5 - Rds ON (mΩ) Vs Input Voltage (V)

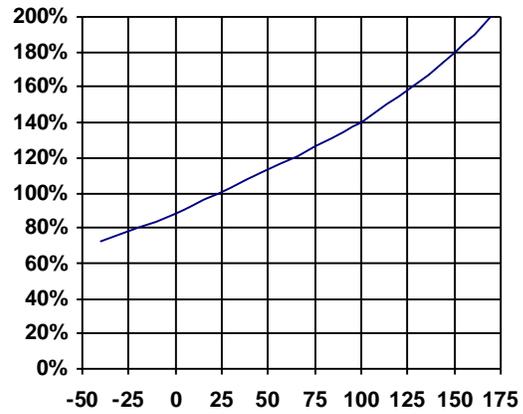


Figure 6 - Normalised Rds ON (%) Vs Tj (°C)

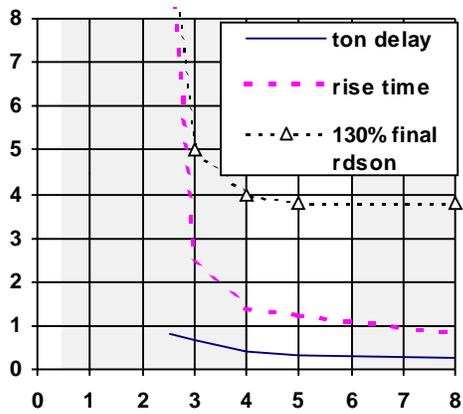


Figure 7 - Turn-ON Delay Time, Rise Time & Time to 130% final Rds(on) (us) Vs Input Voltage (V)

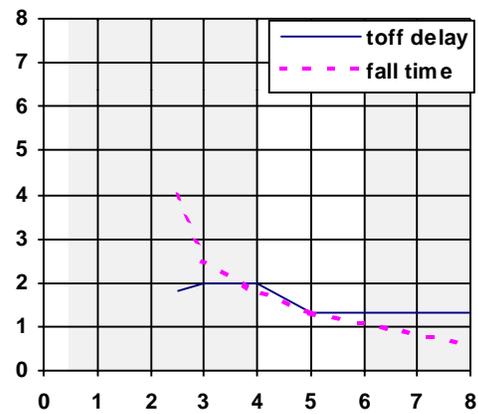


Figure 8 - Turn-OFF Delay Time & Fall Time (us) Vs Input Voltage (V)

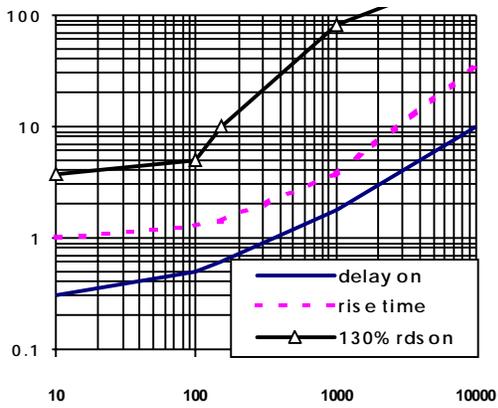


Figure 9 - Turn-ON Delay Time, Rise Time & Time to 130% final Rds(on) (us) Vs IN Resistor (Ω)

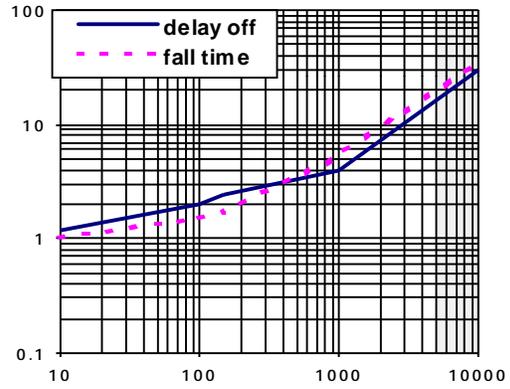


Figure 10 - Turn-OFF Delay Time & Fall Time (us) Vs IN Resistor (Ω)

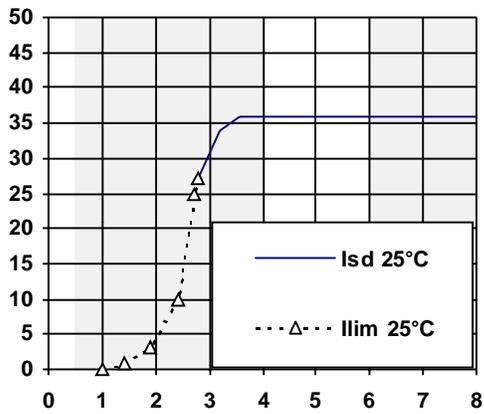


Figure 11 - Current lim. & Ishutdown (A) Vs Vin (V)

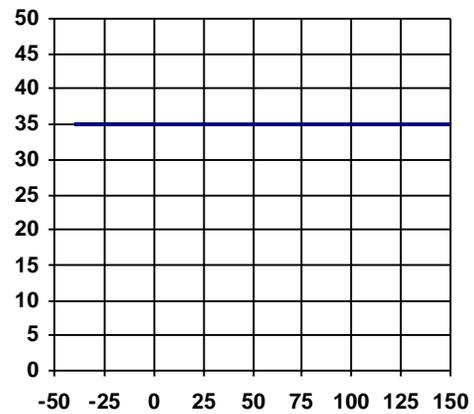


Figure 12 - Ishutdown (A) Vs Temperature (°C)

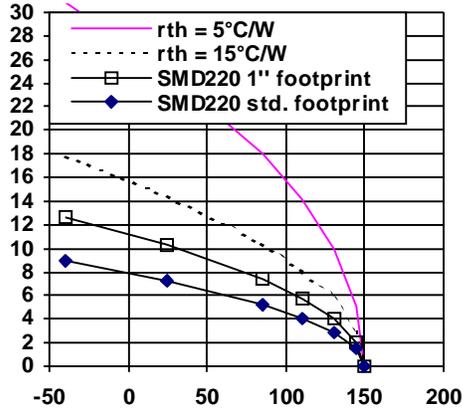


Figure 13 - Max.Cont. Ids (A) Vs Ambient Temperature (°C)

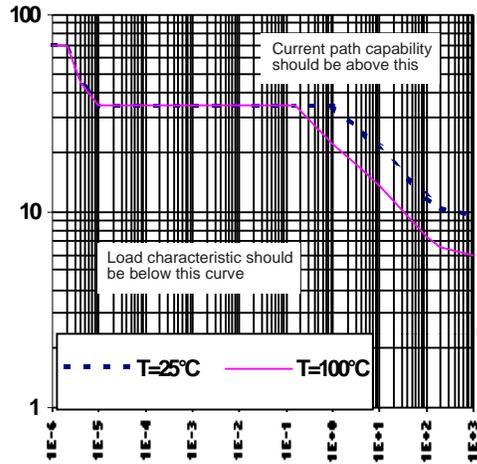


Figure 14 - Max.Cont. Ids (A) Vs Ambient Temperature (°C)

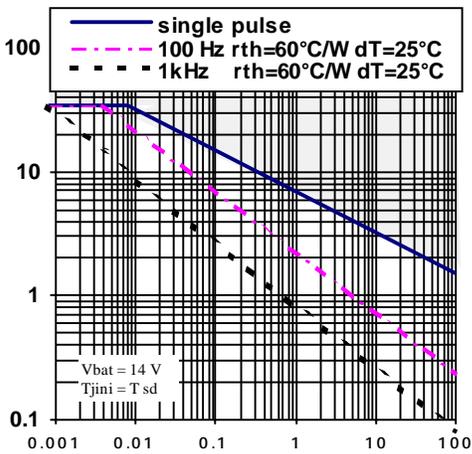


Figure 15 - Iclamp (A) Vs Inductive Load (mH)

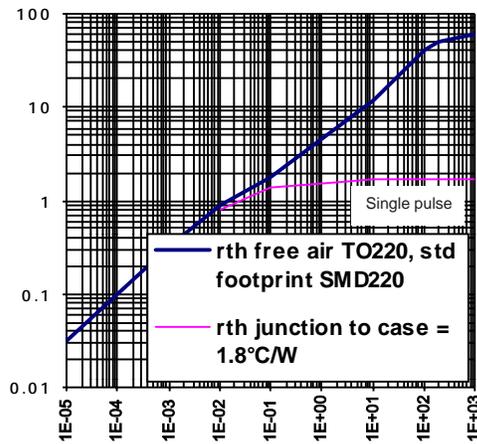


Figure 16 - Transient Thermal Imped. (°C/W) Vs Time (s) - IPS0151/IPS015S

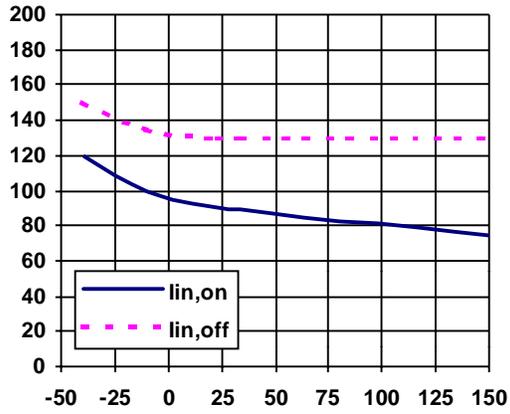


Figure 17 - Input current (µA) Vs Junction (°C)

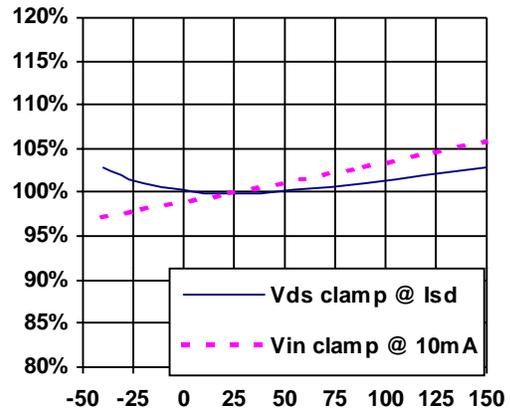


Figure 18 - Vin clamp and V clamp2 (V) Vs Tjunction (°C)

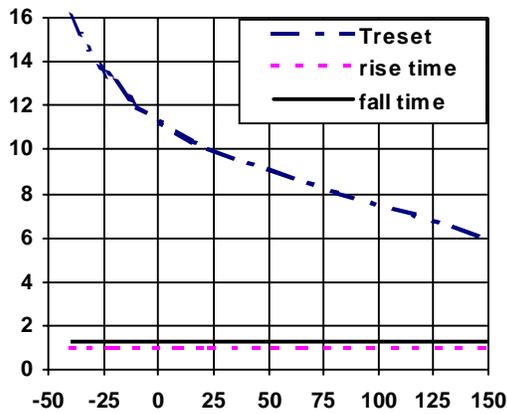
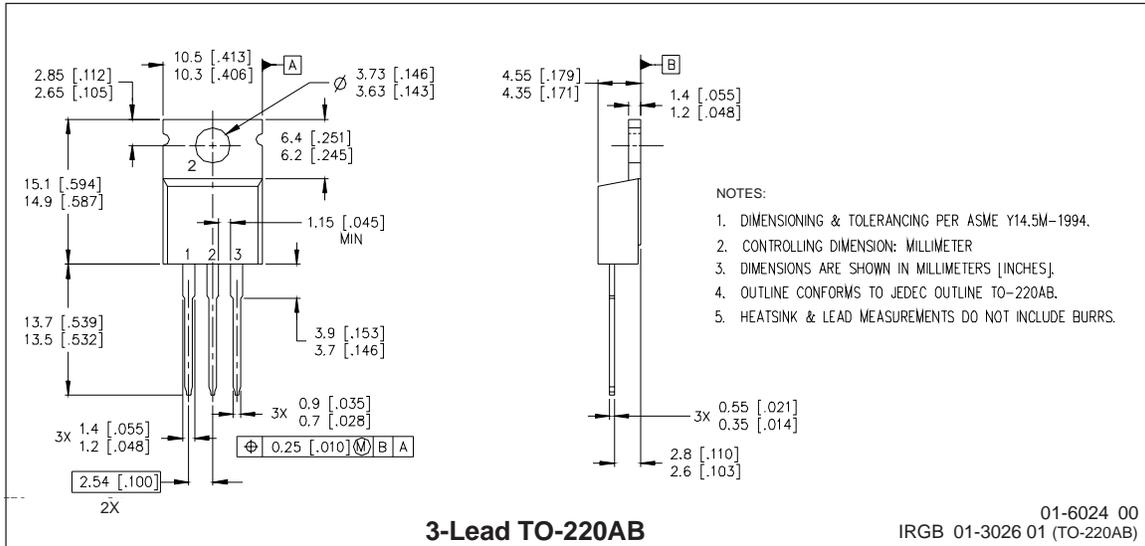
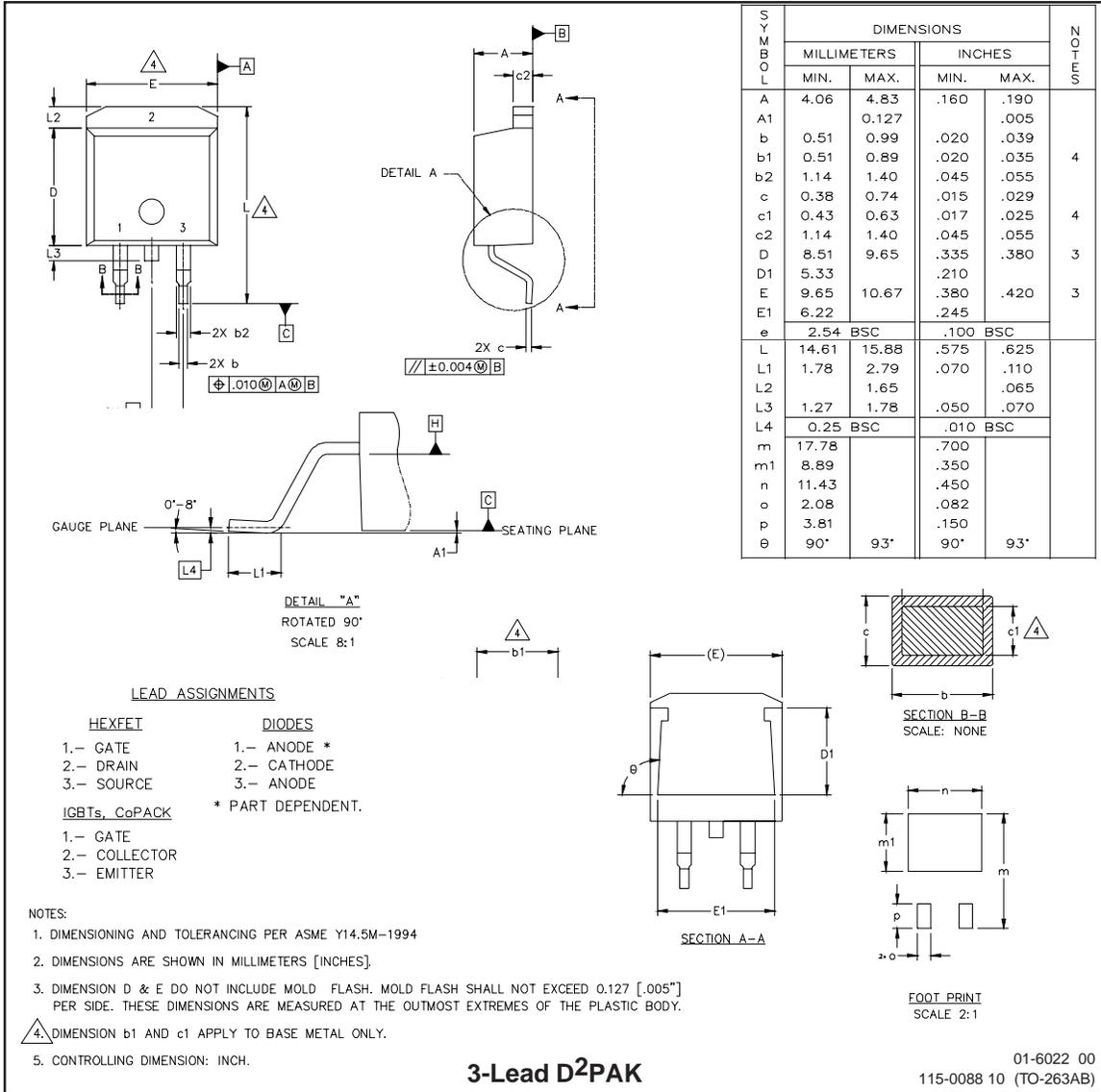


Figure 19 - Turn-on, Turn-off, and Treset (µs) Vs Tjunction (°C)

Case Outline



Case Outline

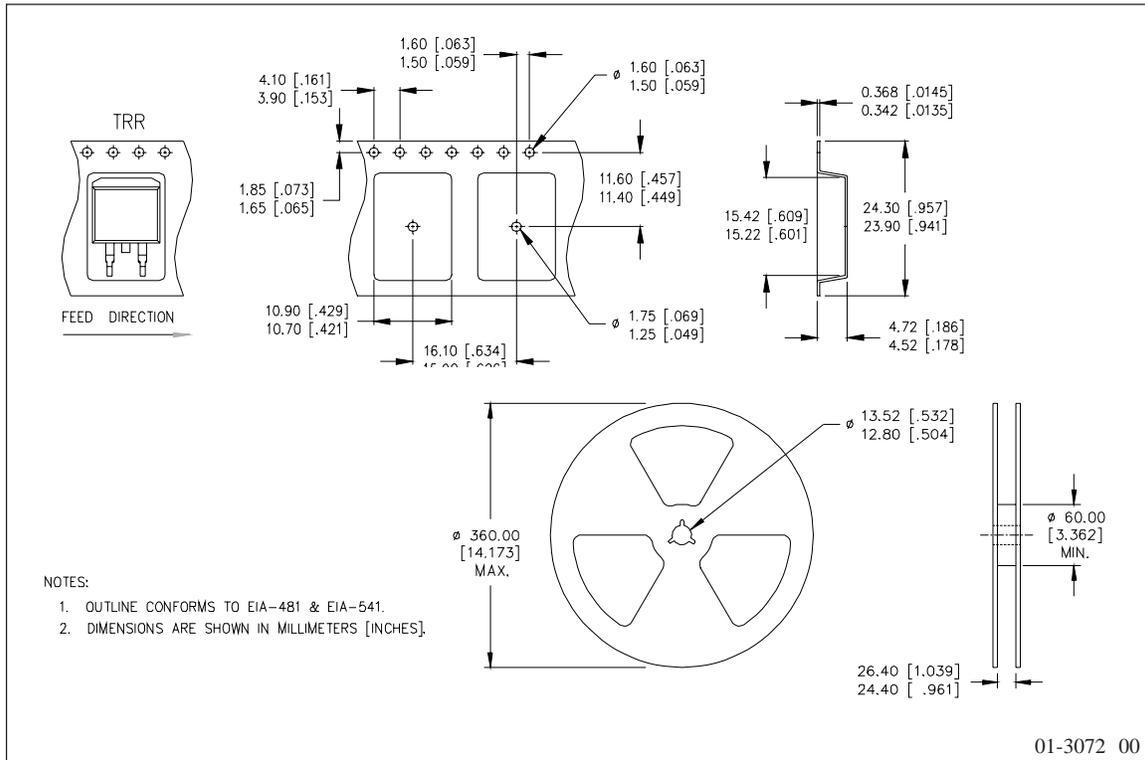


SYMBOL	DIMENSIONS				NOTES
	MILLIMETERS		INCHES		
	MIN.	MAX.	MIN.	MAX.	
A	4.06	4.83	.160	.190	4
A1		0.127		.005	
b	0.51	0.99	.020	.039	
b1	0.51	0.89	.020	.035	
b2	1.14	1.40	.045	.055	4
c	0.38	0.74	.015	.029	
c1	0.43	0.63	.017	.025	3
c2	1.14	1.40	.045	.055	
D	8.51	9.65	.335	.380	3
D1	5.33		.210		
E	9.65	10.67	.380	.420	3
E1	6.22		.245		
e	2.54 BSC		.100 BSC		
L	14.61	15.88	.575	.625	
L1	1.78	2.79	.070	.110	
L2		1.65		.065	
L3	1.27	1.78	.050	.070	
L4	0.25 BSC		.010 BSC		
m	17.78		.700		
m1	8.89		.350		
n	11.43		.450		
o	2.08		.082		
p	3.81		.150		
θ	90°	93°	90°	93°	

IPS0151(S)

International
IR Rectifier

Tape & Reel - D²PAK (SMD220)



International
IR Rectifier

IR WORLD HEADQUARTERS: 233 Kansas St., El Segundo, California 90245 Tel: (310) 252-7105
Data and specifications subject to change without notice. 11/13/2001