



SERVICES CULTURE ÉDITIONS
RESSOURCES POUR
L'ÉDUCATION NATIONALE

**Ce document a été numérisé par le CRDP de Bordeaux pour la
Base Nationale des Sujets d'Examens de l'enseignement professionnel.**

Campagne 2013

**BREVET DE TECHNICIEN SUPÉRIEUR
INFORMATIQUE ET RÉSEAUX
POUR L'INDUSTRIE ET LES SERVICES TECHNIQUES**

ÉTUDE D'UN SYSTÈME INFORMATISÉ

Session 2013

Durée : 6 heures
Coefficient 5

Ce sujet comprend :

<i>Partie</i>	<i>Pagination</i>	<i>Couleur des feuilles</i>
Présentation du système et questionnement	pages 1 à 22	Rose
Annexes	pages 1 à 22	Vert
Document réponses À RENDRE OBLIGATOIREMENT, AGRAFÉ À UNE COPIE MODÈLE EN	pages 1 à 19	Blanc

Matériel autorisé :

L'usage de la calculatrice est autorisé (circulaire n°99-1 86 du 16-11-1999).

Tout autre document ou matériel est interdit.

Dès que le sujet vous est remis, assurez-vous qu'il est complet.

BTS INFORMATIQUE ET RÉSEAUX POUR L'INDUSTRIE ET LES SERVICES TECHNIQUES	Session 2013
ÉTUDE D'UN SYSTÈME INFORMATISÉ	Code : IRSES

BREVET DE TECHNICIEN SUPÉRIEUR INFORMATIQUE ET RÉSEAUX POUR L'INDUSTRIE ET LES SERVICES TECHNIQUES

ÉTUDE D'UN SYSTÈME INFORMATISÉ

Session 2013

—
Durée : 6 heures

Coefficient 5
—

SUJET

(22 PAGES)

Toutes les réponses aux questions sont à fournir sur le livret intitulé « document réponse », à l'exclusion de tout autre support. Ce document sera agrafé à une copie modèle EN.

Les réponses doivent être **exclusivement** situées dans les emplacements prévus à cet effet. Si nécessaire, le candidat peut rectifier ses réponses sur la page non imprimée **en regard**. Une réponse ne doit être justifiée que si la question le demande.

Pour des raisons de confidentialité certaines informations industrielles ont été modifiées.

Temps conseillés et barèmes indicatifs :

A. PRÉSENTATION DU SYSTÈME	30 mn	
B. ANALYSE DU CONTEXTE	45 mn	15 points
C. CONCEPTION DU SYSTÈME EMBARQUÉ	60 mn	21 points
D. ORGANISATION DES DONNÉES	10 mn	3 points
E. TEMPS REEL	45 mn	22 points
F. LES COMMUNICATIONS	65 mn	23 points
G. RÉSEAU	45 mn	16 points
Relecture	30 mn	

Matériel autorisé :

L'usage de la calculatrice est autorisé (circulaire n°99-186 du 16-11-1999) .

Tout autre document ou matériel est interdit.

Dès que le sujet vous est remis, assurez-vous qu'il est complet.

Étude d'un parc Éolien

A. PRÉSENTATION DU SYSTÈME.



photo 1 : Le parc éolien de Port Saint Louis.

L'énergie électrique éolienne est aujourd'hui en pleine expansion. La société Mistral Energie s'est développée pour apporter sa solution. Le système étudié est une unité qui a pour objectif à terme de produire 30 MW (Méga Watt).

Actuellement, 25 machines de 850 kW sont implantées sur une partie du site. Ces machines sont raccordées sur un réseau interne de 20kV. Un transformateur 20/63 kV permet de se raccorder au réseau RTE (Réseau de Transport d'Electricité).

La surveillance du parc se fait à plusieurs niveaux :

- Surveillance locale des paramètres du site :
Chaque éolienne est équipée d'un système embarqué. Les données machines, telles que la température du frein, la température de l'huile, ..., la vitesse de rotation des pales, la direction et la vitesse du vent, l'orientation de la machine, la puissance fournie ..., sont consultables via le site web local.
- Supervision du site :
Un système de supervision permet de gérer l'ensemble du parc éolien. Ce système est également gérable à distance via un site web sécurisé.
- Surveillance électrique :
Des équipements standards des sites de production électrique surveillent les différentes installations entre le 20kV et le 63 kV.

Les échanges de données s'effectuent sur plusieurs liaisons :

- réseau CIBUS
- réseau MODBUS
- réseau CANBUS
- réseau ARCNET
- réseau ETHERNET.

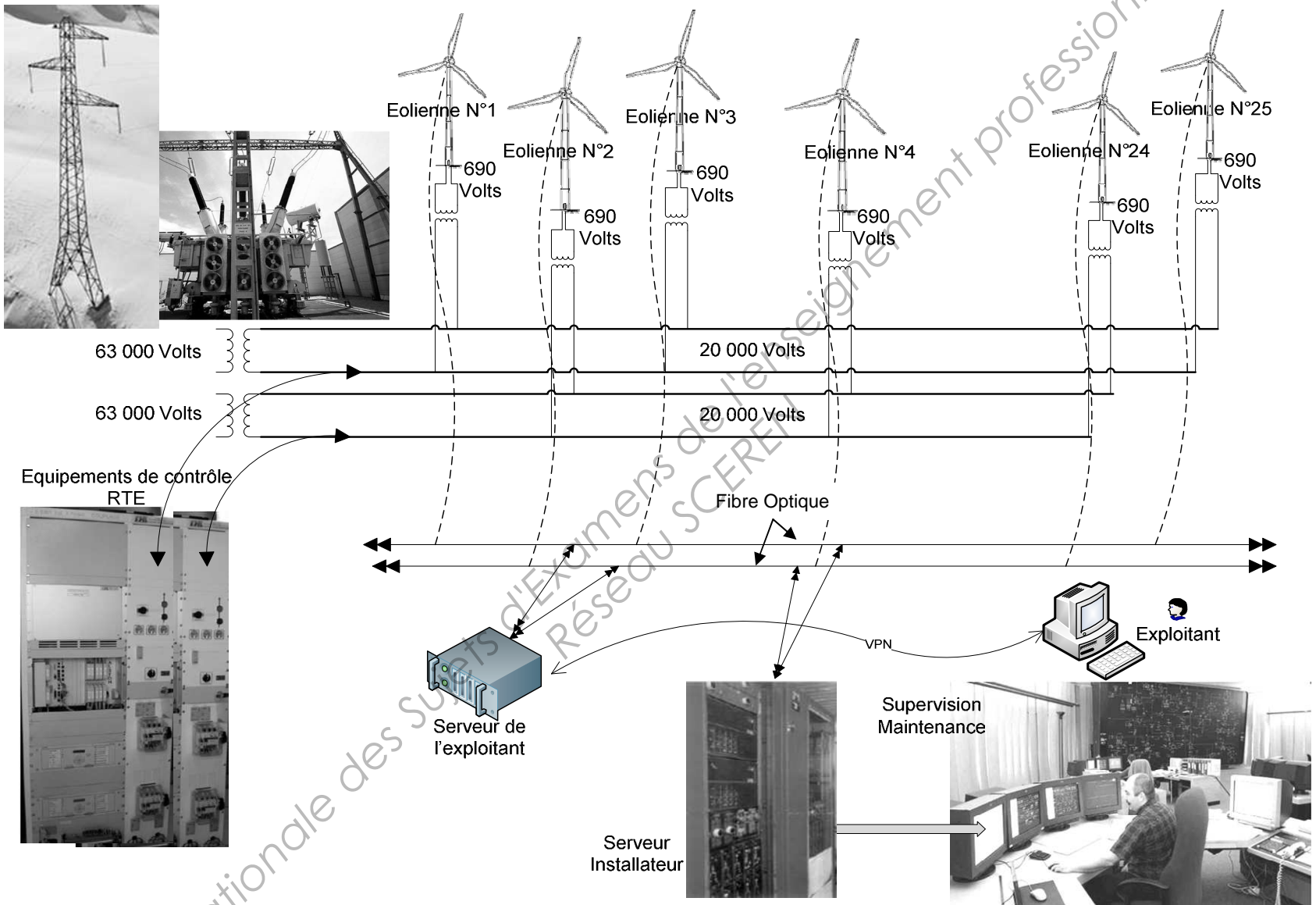


Figure 1 : Synoptique simplifié du système.

Architecture simplifiée :

L'architecture réseau du parc éolien (ou ferme éolienne) peut être décomposée d'un point de vue structurel en plusieurs niveaux :

- réseaux dans l'éolienne (ensemble nacelle et mât)
- réseau intranet
- réseau permettant de sortir sur l'extérieur.

Sur le plan fonctionnel, deux acteurs industriels interviennent sur le parc éolien :

- **L'exploitant** (la société Energie Mistral) produit puis revend de l'énergie électrique à la RTE, Réseau de Transport d'Electricité (filiale ERDF). L'exploitant pilote la mise en marche, l'arrêt des éoliennes, donne les consignes de production et obtient les informations de supervision.
Le **serveur exploitant** dialogue avec les calculateurs embarqués situés dans les nacelles des éoliennes.
- **L'installateur** (la société Vestas) réalise la maintenance du parc. Il accède aux différentes caractéristiques des éoliennes. Il intervient sur les paramètres de fonctionnement.

Diagramme des cas d'utilisation :

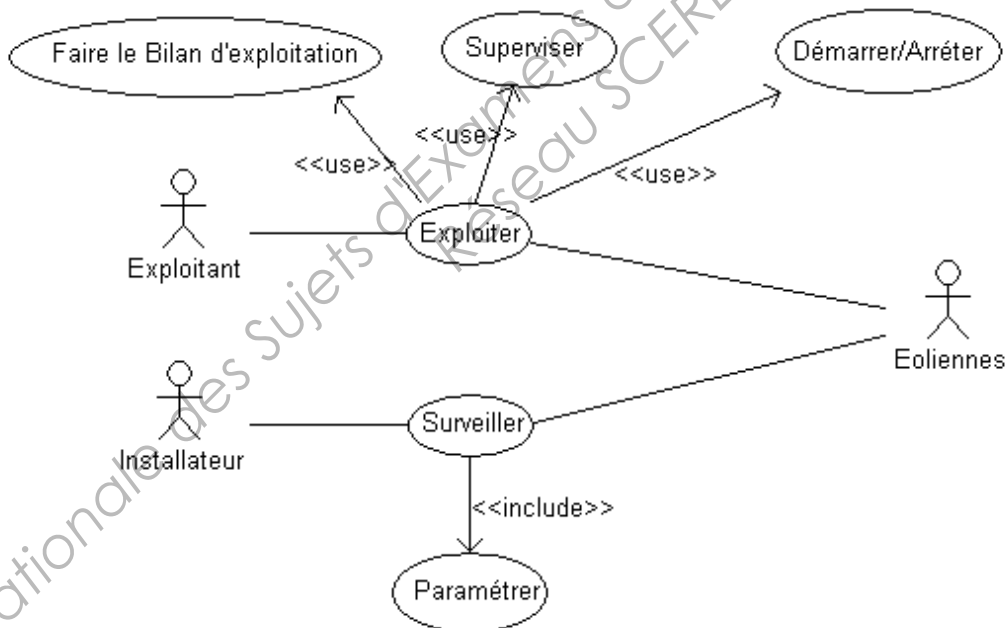


Diagramme 1 : L'offre de service du champ d'éolien.

La **figure 2** montre l'architecture simplifiée pour deux éoliennes (une paire et une impaire).

Cette configuration peut être étendue pour un nombre quelconque de machines.

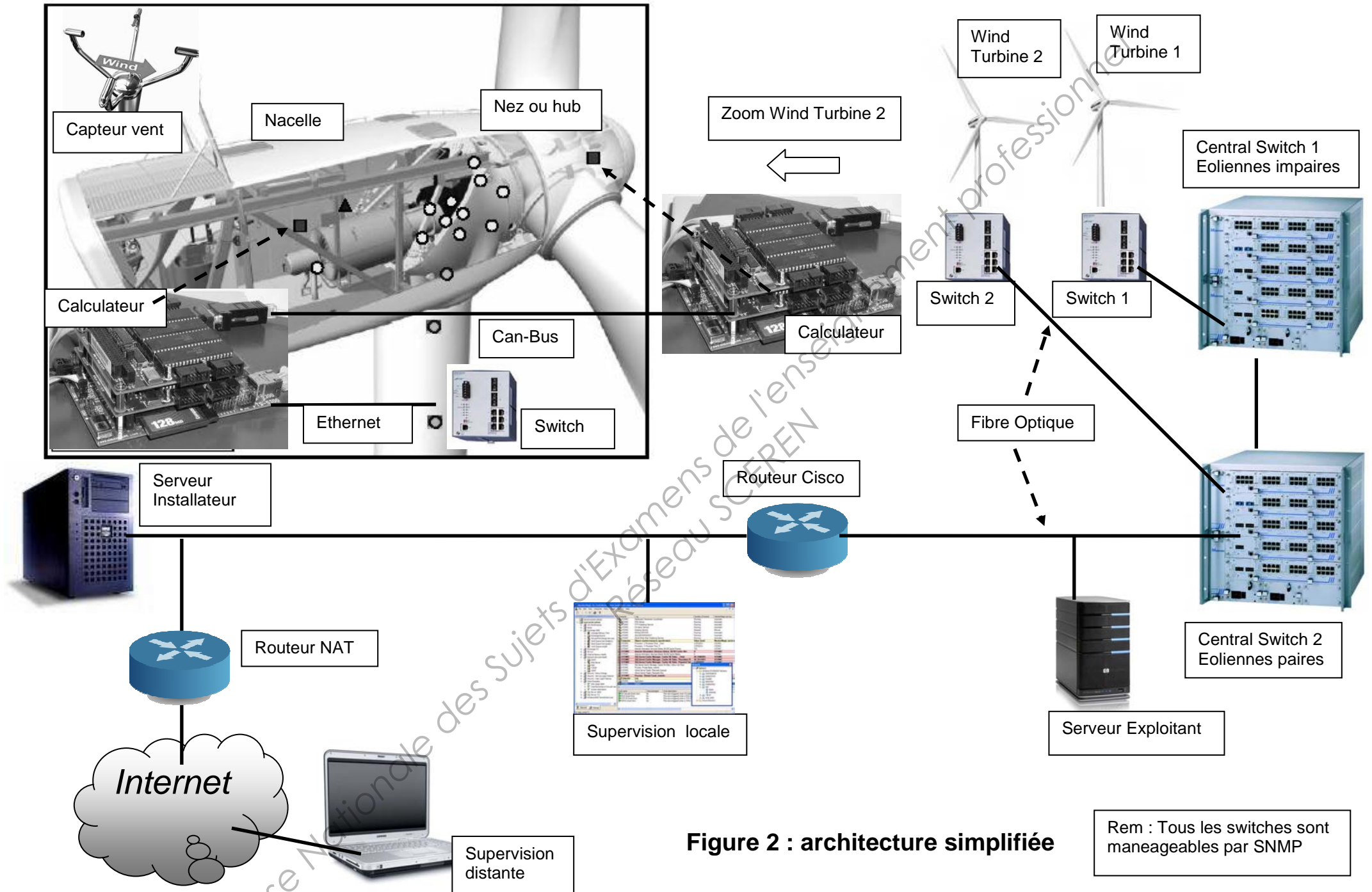


Figure 2 : architecture simplifiée

B. ANALYSE DU CONTEXTE.

Le parc éolien.

Prise de connaissance globale du système.

La société Mistral Energie a sélectionné des éoliennes V52.

Celles-ci sont implantées sur une bande de terre de 4500 m (le long d'un canal).

Question B1 (annexe 1) :

- Indiquez le nombre d'éoliennes nécessaire pour produire 30 MW.
- Combien d'éoliennes supplémentaires doit on implanter sur le site pour atteindre cette production ?

Question B2 (annexe 1) :

- Sachant que la distance entre deux éoliennes est au minimum de **2 fois le diamètre du rotor**, calculez le nombre maximal d'éoliennes de type V52 qu'il est possible d'implanter sur le site de Port Saint Louis.
- Quelle est alors la puissance maximum que peut espérer produire la société Energie Mistral ?

Le réseau électrique.

La production électrique doit se conformer au réseau de distribution d'énergie.

RTE est le transporteur en France de cette ressource et oblige la société à fournir son énergie sous une tension de 63kV. Le réseau interne de l'exploitant est de 20kV.

L'échauffement d'un câble s'exprime en degré / mm²

Question B3 (annexe 1) :

Le raccordement au réseau de distribution doit se faire en 63 kV.

- À l'aide du tableau 1 du document réponse, justifiez le choix de la tension 63kV pour transporter l'énergie.
- À l'aide des caractéristiques de la V52, donnez le rapport global de transformation pour atteindre la valeur de 63 kV.

Étude de l'orientation de l'éolienne.

Capteur de vent (annexe 2) :

Le capteur donne à la fois la direction et la vitesse du vent. La mesure s'effectue par le calcul du temps de déplacement du son (340 ms^{-1}) mis entre l'émetteur et le récepteur (distants de 0,2 m).

La vitesse et la direction du vent influent directement sur les temps de transmission entre l'émetteur et le récepteur.

La figure 3 suivante donne les temps mesurés par le capteur vent.

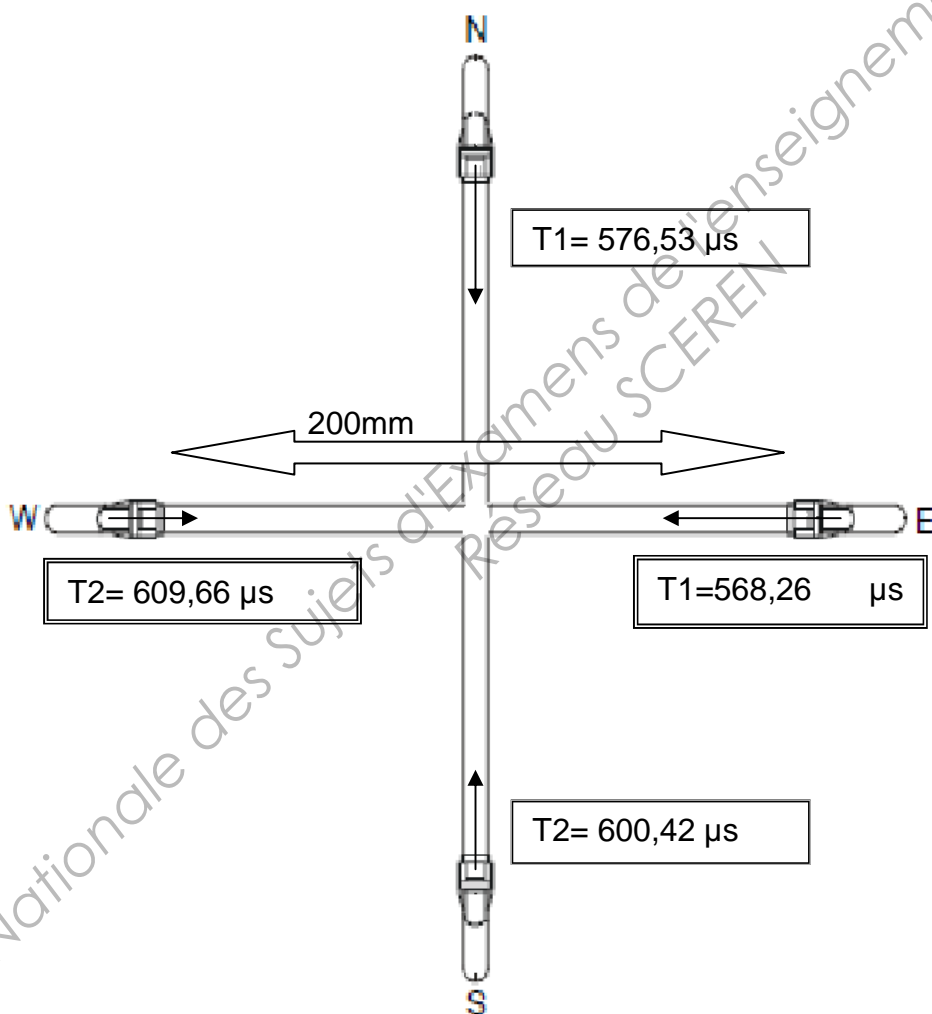


Figure 3 : La mesure de la vitesse et direction du vent.

Question B4 (annexe 2) :

- À partir de la figure 2 de l'annexe, sachant que les transducteurs du capteur vent sont distants de 0.2m, complétez le tableau 2.
- Représentation du résultat :
 - Sur la figure de la rose des vents :
 - tracez la vitesse obtenue dans la direction N-S
 - tracez la vitesse obtenue dans la direction E-W.
(Vous pouvez, pour le tracé, arrondir à la valeur entière supérieure les résultats obtenus dans le tableau 2.)
 - Tracez la vitesse résultante (le vent vient de la direction opposée).
- Le calculateur ne sait pas dessiner et procède à une résolution plus formelle :
 - calculez le module de la vitesse du vent en ms^{-1} , puis en km/h
 - calculez la direction du vent et exprimez-la en degré.

La capture d'une réponse du capteur vent donne la trame suivante :

<soh>	104	A	<stx>	C13	♥	30113	22165	31128	2136	101159	22165	0878	<eot>	6
-------	-----	---	-------	-----	---	-------	-------	-------	------	--------	-------	------	-------	---

Question B5 (annexe 2) :

À partir de cette trame de communication sous le protocole CIBus et sachant que « DD » représente la direction du vent en dizaine de degrés et « FFi » représente la force du vent en dixième de ms^{-1} , donnez la direction et la vitesse du vent pour les deux dernières minutes. Le dernier champ de cette trame représente la valeur du CRC.

C. CONCEPTION DES SYSTÈMES EMBARQUÉS.

Étude du système embarqué de la nacelle.
Analyse du suivi.

Question C1 :

À partir de l'architecture simplifiée (Figure 2 page 6 de ce dossier), complétez le diagramme de déploiement depuis le capteur de vent situé sur la nacelle d'une éolienne jusqu'à la *supervision locale*. En plus des cardinalités, vous ferez apparaître les différents changements de support pour transporter l'information.

Quand la vitesse du vent (**float vitesse**) est en dessous de 2 ms^{-1} , l'éolienne se met en arrêt, c'est-à-dire que les pales sont orientées parallèles au vent (la fonction Lire_direction_vent() retourne un entier exprimant la direction du vent en dizaines de degrés), et le contrôle sera repris pour des vitesses supérieures à 5 ms^{-1} .

Question C2 :

Donnez l'algorithme de la fonction
void Gestion_Vent_Faible(float vitesse) ;

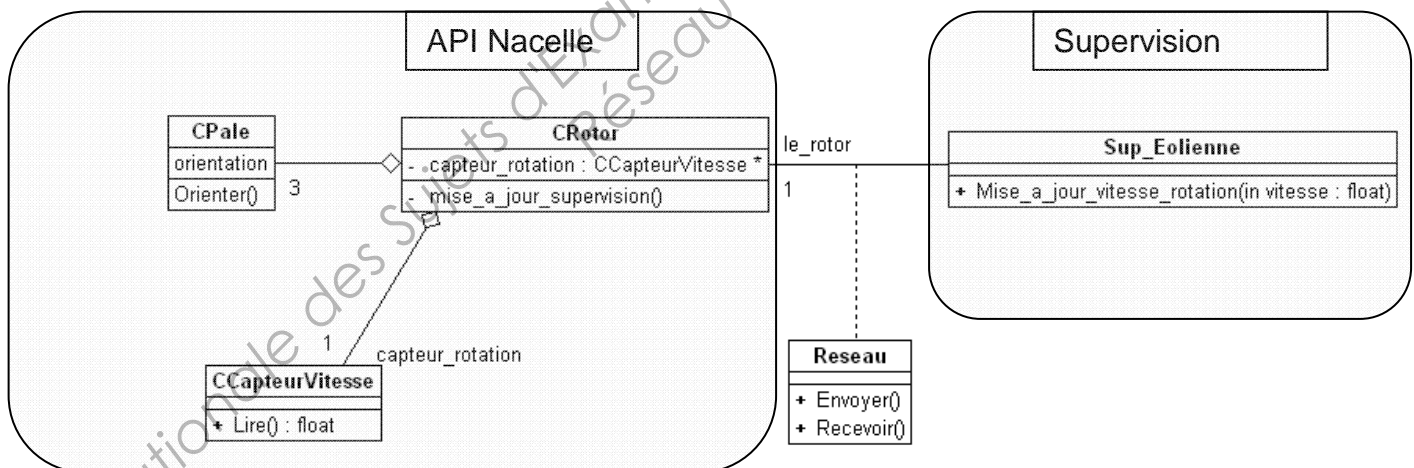


Diagramme 2 : Diagramme simplifié de la mise à jour supervision.

Question C3 :

À partir du diagramme 2 de classes ci-dessus :

- Que signifie le trait en pointillé vers la classe **Reseau** ?
- Donnez le diagramme de séquence du scénario permettant la mise à jour de la vitesse de rotation sur la supervision.

Détail de conception : on affine l'étude de la communication. Le diagramme de classe devient le suivant :

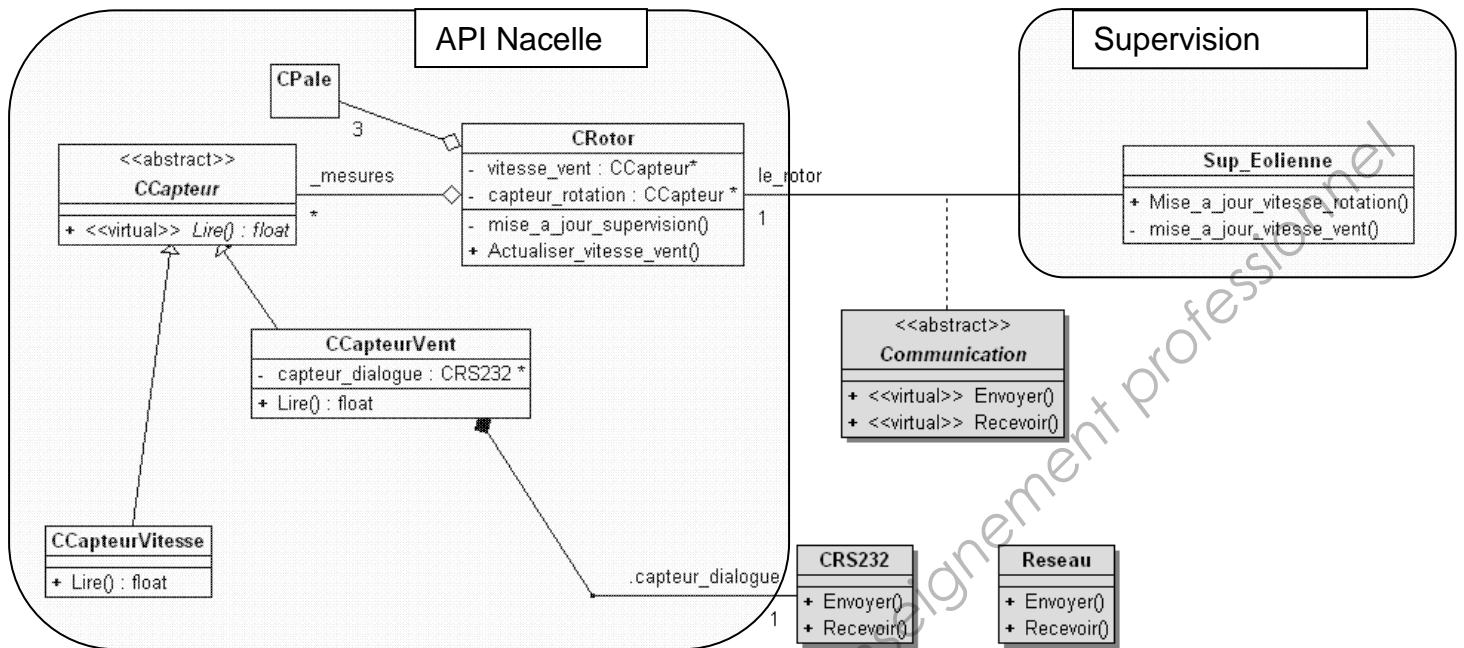


Diagramme 3 : Diagramme détaillé de la mise à jour supervision.

Question C4 :

Compléter le diagramme 3 sur le document réponse en précisant les relations qui peuvent exister entre les classes **CRS232**, **Communication** et **Reseau**.

Question C5 :

Expliquez pourquoi la méthode **CCapteur::Lire()** doit être virtuelle pure ?

Question C6 :

Qu'est-ce que cela implique au niveau des méthodes **Lire()** des classes dérivées ?

Codage.

Le calculateur de la nacelle de l'éolienne N°13 a reçu par la supervision l'ordre de s'arrêter. Le calculateur enregistre l'événement et les conditions d'arrêt dans un journal local. Les événements sont datés à la seconde près.

Question C7 :

Écrivez une structure de données pour une mise en arrêt de l'éolienne 13, à 14h57mn36s avec un vent de $1,9\text{m}\cdot\text{s}^{-1}$ par 30° 54. La vitesse de rotation de l'éolienne est de $0,3\text{trs}\cdot\text{s}^{-1}$.

Question C8 :

- Précisez la visibilité du constructeur `CCapteurVent()` du diagramme de classes (cf. diagramme 4).
- Répondez au QCM en complétant le tableau 3, en vous appuyant sur le listing écrit dans le document réponse.

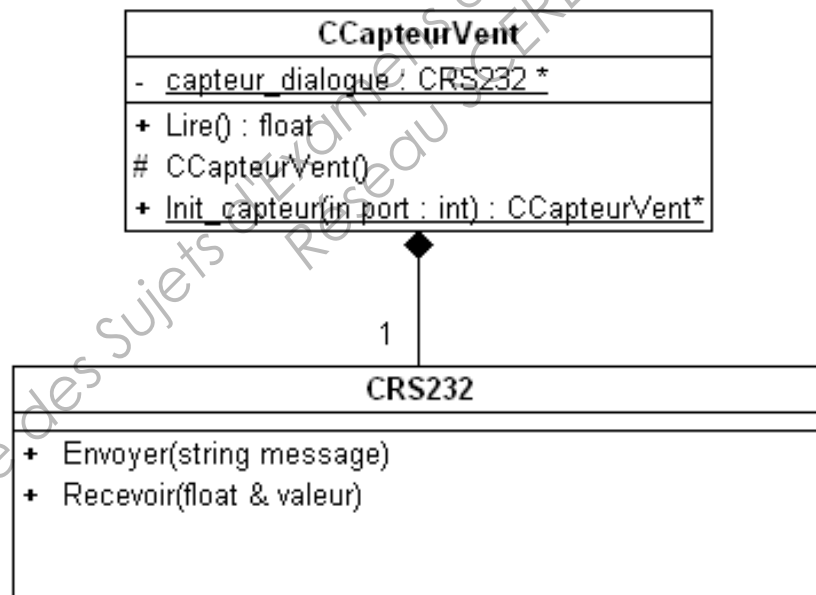


Diagramme 4 : Le capteur vent, détail.

D. ORGANISATION DES DONNÉES.

Les éoliennes peuvent se trouver dans un état "d'arrêt" suite à un évènement d'arrêt provoqué par la Supervision. Cet état signifie, non pas que l'éolienne ne tourne pas, mais tout simplement qu'elle ne produit pas d'énergie car les conditions atmosphériques ne le permettent pas (vent trop faible ou trop fort par exemple).

Dans ce cas, l'éolienne est orientée et positionnée d'une certaine façon afin de protéger les mécanismes (pale, nacelle) de cette dernière.

Tous les évènements d'arrêts journalier sont enregistrés dans un fichier XML et consultables. Dans une même journée, une éolienne peut être mise en état d'arrêt plusieurs fois. Un extrait d'un fichier XML **evt_arret.xml** est donné **en annexe 3**. Ce fichier contient notamment l'heure d'arrêt et les caractéristiques de positionnement.

Question D1 (annexe 3) :

- Indiquez le nom de la racine du fichier XML.
- Combien de nœuds "fils" comporte le fichier XML à partir du nœud racine ?

Base Nationale des Sujets d'Examens de l'enseignement professionnel
Réseau SCEREN

Présentation de l'ensemble nacelle-mât (Parties E et F)

Une éolienne est composée de pales montées sur le "nez" (appelé hub) et d'un calculateur « PC104 Hub » **référéncé CT316**. Celui-ci dialogue avec le boîtier "Top Control Panel" PC104 Nacelle **référéncé CT316**. La hauteur d'un mât peut varier de 30 à 100 mètres. La longueur d'une nacelle n'excède pas 15 mètres pour 3 à 4 mètres de large.

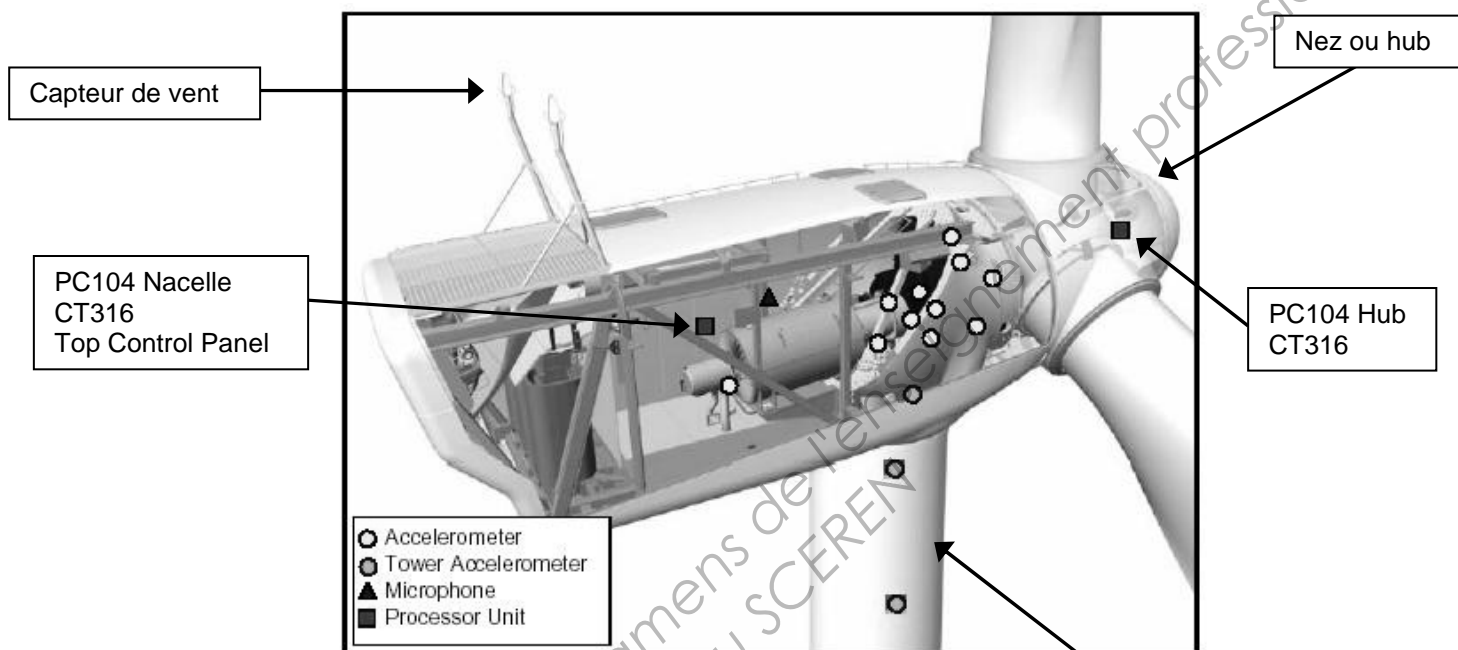
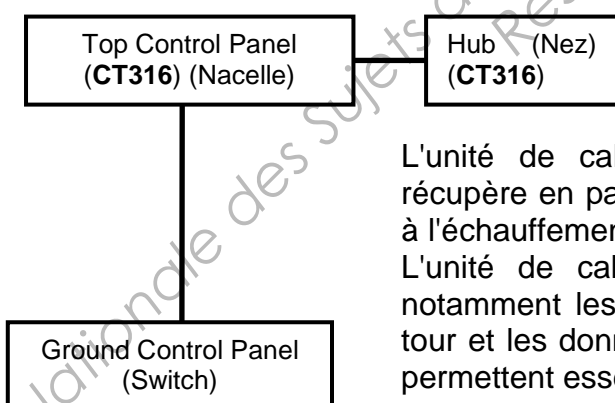


Schéma structurel :



L'unité de calcul (Processor Unit) placée dans le nez récupère en particulier des informations liées à la rotation et à l'échauffement du rotor.

L'unité de calcul se trouvant dans la nacelle récupère notamment les mesures des accéléromètres situés dans la tour et les données provenant du "nez". Les accéléromètres permettent essentiellement de mesurer les vibrations :

- dues au mouvement de rotation des pales. Elles se situent essentiellement dans l'ensemble Hub/nacelle
- provenant de la tour (mât) de l'éolienne et dues au vent.

Les unités de calcul sont constituées de PC104 (CT316).

E. TEMPS RÉEL

Dans ce chapitre nous allons nous intéresser à la mise en œuvre de la partie temps réel du système Hub / Nacelle.

Le PC104 "Top Control Panel" situé dans la nacelle :

Il est constitué :

- d'une carte mère Freescale EVOC 104-3511CD 2N PC/104 architecturée autour d'un microprocesseur MPC8245 avec un OS LINUX/FreeRTOS
- d'une carte de communication MODBUS
- d'une carte de communication CAN.

Le PC104 "Top Control Panel" doit :

- Surveiller en temps réel :
 - la vitesse et l'orientation du vent,
 - la température rotor/stator de l'ensemble alternateur,
 - les vibrations de la tour.
- Recevoir les paramètres de consigne de la supervision.
- Mettre à jour en temps réel sur la supervision les différentes valeurs des capteurs situés sur l'éolienne (nacelle + hub).
- Gérer la communication Ethernet avec le poste de supervision et la communication CAN avec le PC104 situé dans le hub.
- Signaler les défauts constatés sur les équipements.

Pour répondre à cette contrainte, la communication Ethernet utilisée est basée sur une architecture client / serveur.

Question E1 (annexe 4) :

- Indiquez sur quelle architecture est basée le processeur MPC8245 de la carte mère EVOC 104-3511CD 2N PC/104.
- Rappelez les caractéristiques principales de cette architecture.

Question E2 :

Le système d'exploitation retenu est un système dit préemptif.

Donnez une définition et rappelez le principe de fonctionnement d'un tel système ?

Question E3 :

Les concepteurs du système ont attribué le rôle de serveur au PC104 « Top Contrôle Panel » Justifier ce choix.

Le PC104 "Hub" situé dans le nez :

Il est constitué d'une carte mère Freescale EVOC 104-3511CD 2N PC/104 architecturée autour d'un microprocesseur MPC8245, son OS est LINUX/FreeRTOS.

Le PC104 "Hub" situé dans le nez est chargé :

- De surveiller :
 - la température du système d'orientation des pales,
 - la vitesse de rotation des pales,
 - les vibrations du nez.
- De dialoguer via une communication CAN avec la nacelle afin d'échanger des paramètres.
- De commander l'orientation des pales.

Question E4 :

- Combien de tâches allez-vous déclarer dans votre application ? Justifiez votre réponse.
- Donner un nom représentatif à chacune de vos tâches. Complétez le tableau 5.

Question E5 (annexe 4 – Listing 1) :

- Donnez les lignes de code correspondant à la déclaration de vos tâches.
- Donnez les lignes de code permettant de créer chacune de vos tâches. (Utilisez les paramètres par défaut)

Le PC104 "Top Control Panel" stocke dans une mémoire partagée l'ensemble des paramètres (Nacelle + hub) de fonctionnement relatif à l'éolienne. Ces données sont transmises au poste de supervision et/ou au service de maintenance.

La mémoire partagée est accessible en écriture par les tâches **Hub** et **Nacelle** et en lecture par les tâches **Supervision** et **Maintenance**. L'accès doit être exclusif pour chaque tâche. On prévoit de créer un sémaphore binaire pour en contrôler l'accès.

En première analyse on peut considérer :

- Une tâche en écriture **vEcritParamTask** qui bloquera le sémaphore le temps de scruter l'ensemble des paramètres de fonctionnement.
- Une tâche en lecture **vVisualisationTask** lancée sur une interruption du système de supervision. Elle lit les données contenues dans la mémoire partagée.

Question E6 (annexe 4 – Listing 2) :

- Pour **FreeRTOS**, donnez les différentes étapes de gestion du sémaphore, vous indiquerez les primitives.
- Expliquez le fonctionnement du sémaphore (code **listing 2**).

Question E7 :

- Répondez au QCM en complétant le tableau 6 dans le document réponse en vous appuyant sur les exemples précédents (listing 1 et listing 2).

F. LES COMMUNICATIONS.

Nous allons dans cette partie nous intéresser à certains réseaux de terrain présents dans l'ensemble nacelle-mât.

Deux réseaux MODBUS sont utilisés pour récupérer les informations liées aux accéléromètres :

- le premier récupère les informations des capteurs accéléromètres et du microphone situés dans la partie avant dans la nacelle.
- le second récupère les informations provenant d'une série de capteurs accéléromètres placés tout le long de la tour.

Pour des raisons de sécurité électrique, les deux lignes réseaux doivent être isolées.

Configuration de la communication MODBUS :

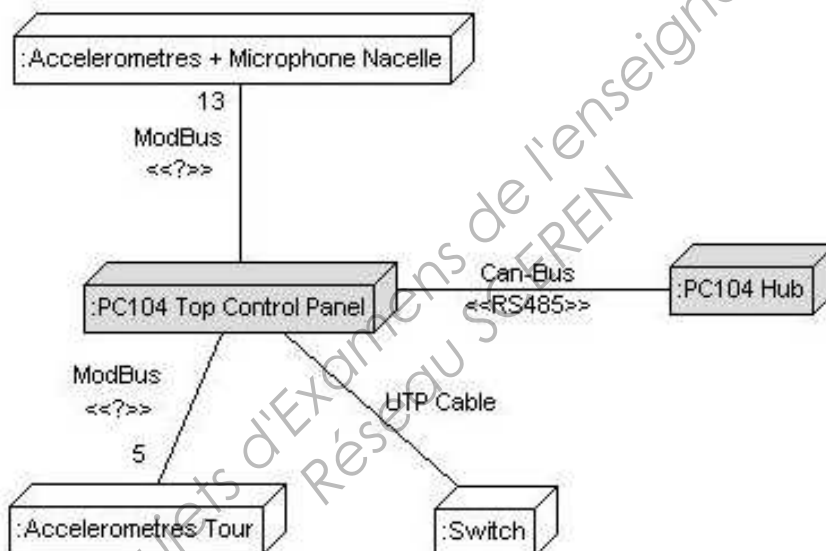


Diagramme 5 : Diagramme de déploiement

On a volontairement omis le stéréotype sur les liaisons MODBUS.

Question F1 (annexe 5) :

Complétez le tableau 7 en justifiant si la liaison série proposée est valide ou non pour les réseaux MODBUS présents.

Question F2 (annexe 6) :

Choisissez la carte de communication la mieux adaptée pour le PC104 "Top Control Panel" (PC104 de la nacelle) parmi les trois cartes proposées. Justifiez votre réponse.

Question F3 :

Complétez le diagramme 5 de déploiement du document réponse en indiquant la nature des liaisons <<?>> manquantes.

Les paramètres de la communication MODBUS sont les suivants :

19200 Bauds, sans parité, 8 bits de données, 1 bit stop

Le channel 1 (Base address 3F8h) est utilisé pour la ligne RS422 et le channel 2 (Base address 2F8h) est utilisé pour la ligne RS485.

Question F4 (annexe 7) : configuration de la carte PCM-3610.

- Quelle doit-être la valeur du diviseur ?
- Positionnez les cavaliers JP10, JP4 et JP5.
- Configurez les switches SW1 et SW2.
- Positionnez les cavaliers sur JP2 et JP3 pour les IRQ.
- Complétez le registre LCR pour la communication MODBUS.

Configuration de la communication CANBUS :

La communication Can-Bus s'effectue entre l'unité "PC104 Top Control Panel" et l'unité "PC104 Hub".

Du côté "PC104 Top Control Panel", la carte **PCM-3680** est enfichée sur le PC104. C'est une carte module d'interface Can-Bus. Seul le port 2 est utilisé.

Question F5 (annexe 5) :

Le réseau CANBUS est basé sur une architecture du type maître/esclave. Déterminez le rôle du "PC104 Top Control Panel" ?

Question F6 (annexe 8) : résistances de "terminaison".

Le réseau CANBUS utilise comme support physique la liaison RS485. La page 23 de l'annexe 8 (Carte PCM-3680) indique que l'on peut positionner des résistances de terminaison (Termination Resistor).

- Rappelez le rôle de cette résistance aussi appelée "Terminator". Quelle est sa valeur approximative ?
- Sachant que l'on utilise le port 2, indiquez les cavaliers (jumpers JP5, JP6) qui doivent-être utilisés sur la carte PCM-3680.

Le chip Can-Controller implanté sur la carte PCM-3680 est le SJA1000 de Philips. La carte PCM-3680 dispose de son propre quartz de fréquence **16Mhz**.

On se propose de vérifier que la vitesse de transmission soit bien de **100kbits/s**

Le concepteur a programmé la carte PCM-3680 en utilisant la bibliothèque Can841.lib

Il a saisi notamment les lignes de codes suivantes :

/* baud rate */

can2.btr0 = 04 ; /* BTR0 register */

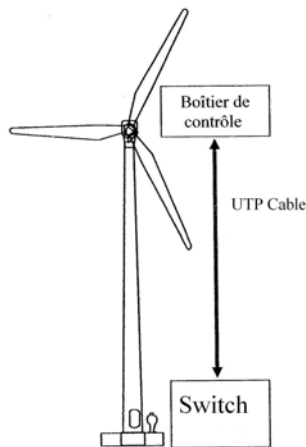
can2.btr1 = 0x3A ; /* BTR1 register */

Question F7 (annexe 9) : détermination du "bit rate".

- Complétez les registres BTR0 et BTR1.
- Déterminez la valeur de t_{scl} (paragraphe 6.5.1.1).
- À l'aide du contenu des registres BTR0 et BTR1, exprimez la valeur du "Nominal Bit Time" = $n * t_{scl} = t_{SYNCSEG} + t_{SEG1} + t_{SEG2}$.
Reportez-vous au début du paragraphe 6.5.2.2 pour déterminer les valeurs de $t_{SYNCSEG}$, t_{SEG1} et de t_{SEG2} . Il est conseillé d'exprimer ces trois valeurs en fonction de t_{scl} .
- En déduire la vitesse de transmission du Can-Bus: bit rate = 1 / Nominal bit time.

Étude de la communication entre le "PC104 top control panel" et le switch :

La liaison physique entre la nacelle et le switch est assurée par un câble de type paire torsadée.

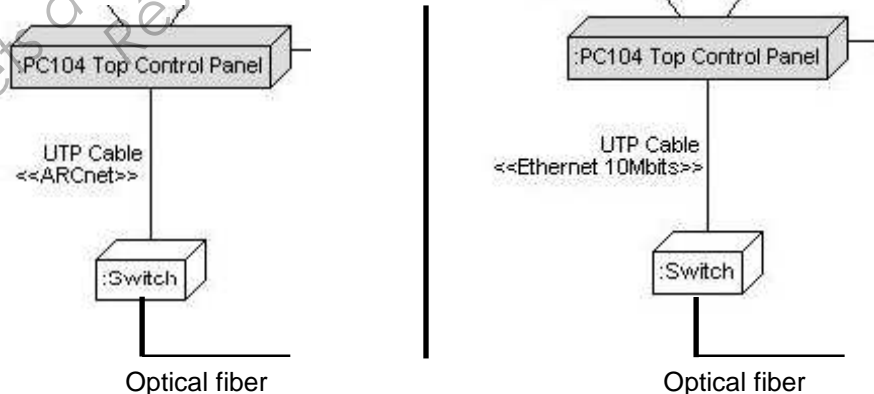


Deux protocoles réseaux peuvent être envisagés :

- réseau Ethernet 10/100 Mbits
- réseau ARCnet.

Les deux solutions ne peuvent pas cohabiter. Dans tous les cas, le "PC104 Top Control Panel" devra être équipé soit d'une carte interface réseau module Ethernet 10/100 Mbits ou d'une carte interface réseau module ARCnet. Les deux familles de protocoles supportent IPv6.

Les deux architectures sont alors les suivantes :



Question F8 :

On souhaite comparer les caractéristiques des deux architectures réseaux. Complétez le tableau 8.

Question F9 :

Rappelez brièvement la particularité d'un réseau déterministe ?

G. PARTIE RÉSEAU.

Dans cette dernière partie, l'étude va porter sur le réseau Intranet du parc éolien et sur l'accès depuis Internet.

Le réseau exploitant est le réseau qui permet de piloter la mise en marche, l'arrêt des éoliennes, de donner les consignes de production et d'obtenir les informations de supervision.

Le réseau installateur, permet à l'installateur d'honorer son contrat de maintenance. Il peut donc accéder aux éoliennes et avoir des informations sur différentes caractéristiques de fonctionnement et éventuellement d'intervenir sur des paramètres de fonctionnement. Ce réseau n'est accessible que par l'installateur via internet. Les accès sont sécurisés. (Voir figure 4, page 21).

Le plan d'adressage est articulé autour d'une adresse réseau **172.16.0.0**

Le réseau est découpé en sous réseaux. Le masque de sous réseau choisi est codé sur **20 bits**.

Question G1 :

Exprimez sous forme décimale le masque de sous réseau.

Question G2 :

Avec un masque de 20 bits, combien de sous réseaux valides peut-on avoir au maximum ?

Question G3 :

Avec un masque de 20 bits, combien de machines maximum peuvent avoir une adresse valide par sous réseau ?

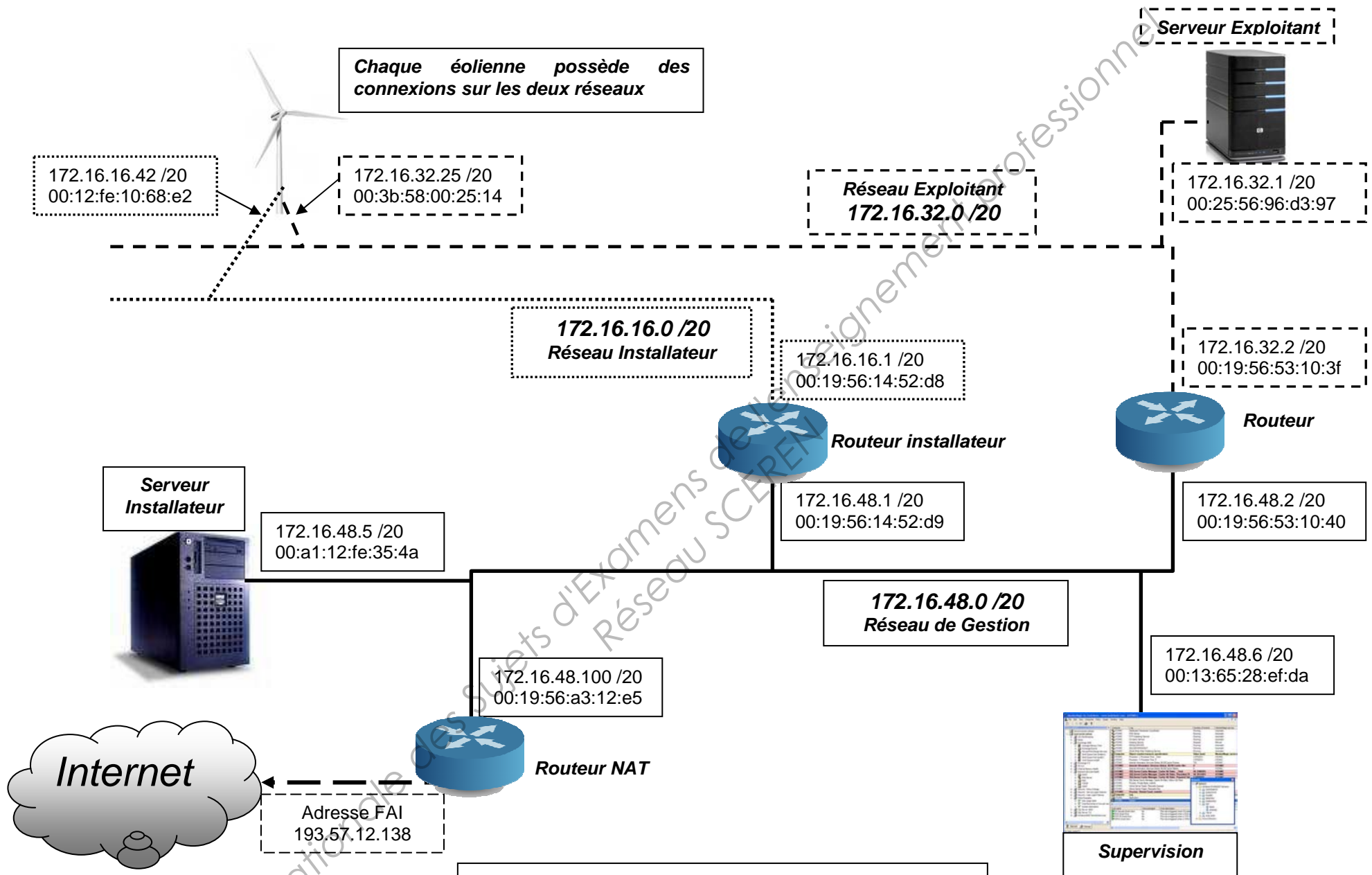
Question G4:

Pour le réseau 172.16.16.0 /20 indiquez :

l'adresse de la première machine possible

l'adresse de la dernière machine possible

l'adresse de l'adresse de diffusion du sous réseau.



Le routeur de l'installateur a pour rôle de séparer les domaines de diffusion.
Le routage est activé entre les deux réseaux 172.16.16.0/20 et 172.16.48.0/20

Question G5 :

Dans le cas d'une communication entre l'éolienne d'adresse **176.16.16.42** et le serveur Installateur, indiquez les différentes adresses que contiennent les trames au cours de la communication.

Un routeur Nat a été installé pour permettre l'accès à internet et autoriser l'accès aux serveurs Installateur et la Supervision.

Question G6 :

Expliquez le rôle et le principe de fonctionnement d'un routeur Nat.
Dans le cas où il y a plus (+) d'adresses privées que publiques, comment la translation est-elle réalisée (Nat dynamique) ?

Question G7 :

Dans le cas d'une communication entre la supervision et une machine située sur internet (**113.65.13.241**), indiquez les différentes adresses que contiennent les trames au cours de la communication.

Question G8 : Nat/Pat

Le serveur de l'installateur possède un accès https sur son port 443. Il doit être accessible de l'extérieur sur internet sur le port 443.

Le serveur de supervision possède un accès https sur son port 443. Il doit être accessible de l'extérieur sur internet via le port utilisateur 50443

Donnez dans le tableau 9 la configuration du routeur NAT afin d'accéder au serveur installateur et la supervision depuis internet.

**BREVET DE TECHNICIEN SUPÉRIEUR
INFORMATIQUE ET RÉSEAUX
POUR L'INDUSTRIE ET LES SERVICES TECHNIQUES**

ÉTUDE D'UN SYSTÈME INFORMATISÉ

Session 2013

—————
DURÉE : 6 HEURES
Coefficient 5
—————

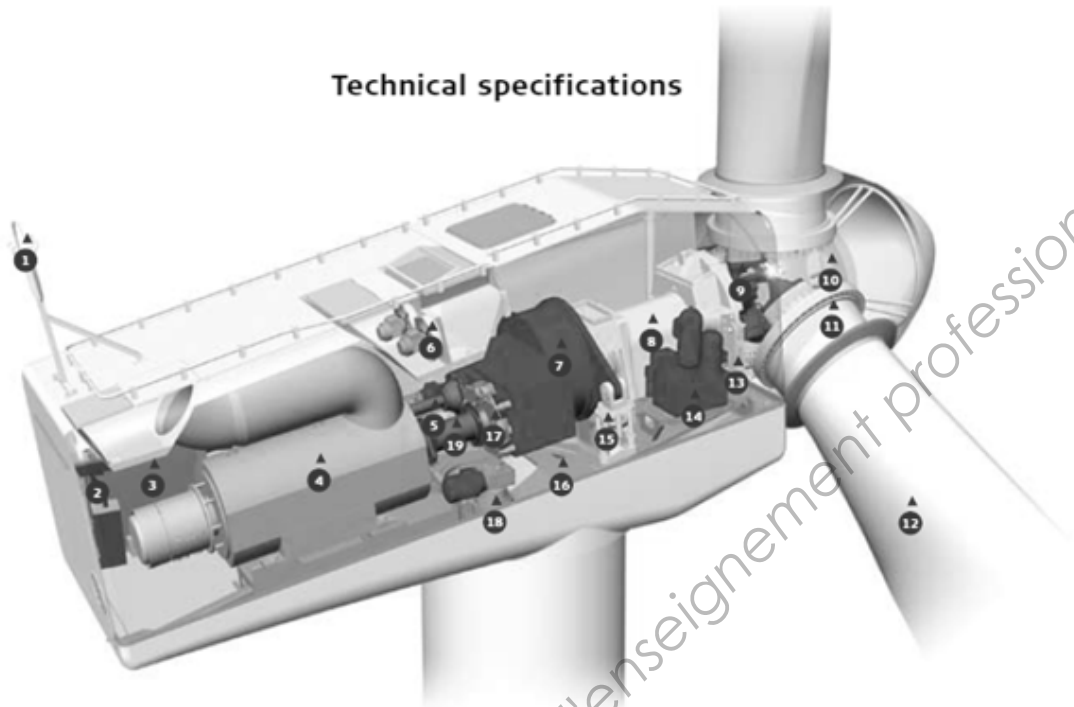
ANNEXES

(22 PAGES)

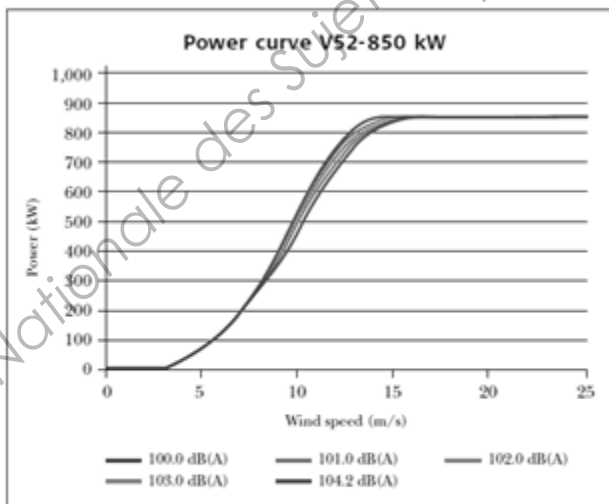
Annexe 1:	Éolienne VESTAS V52 (2 pages)	2
Annexe 2:	Capteur Vent (4 pages).....	4
Annexe 3:	XML, Parser TinyXML, STL (1 page).....	8
Annexe 4:	PC 104 - Carte mère et FreeRTOS (5 pages).....	9
Annexe 5:	Architecture réseau nacelle (2 pages).....	14
Annexe 6:	PC 104 – Modules de communication (1 page)	16
Annexe 7:	Carte Module PCM-3610 (extrait 2 pages).....	17
Annexe 8	Carte Module PCM-3680 (extrait 2 pages).....	19
Annexe 9:	Controller Can-Bus SJA 1000 (extrait 2 pages).....	21

Annexe 1 : Éolienne Vestas V52

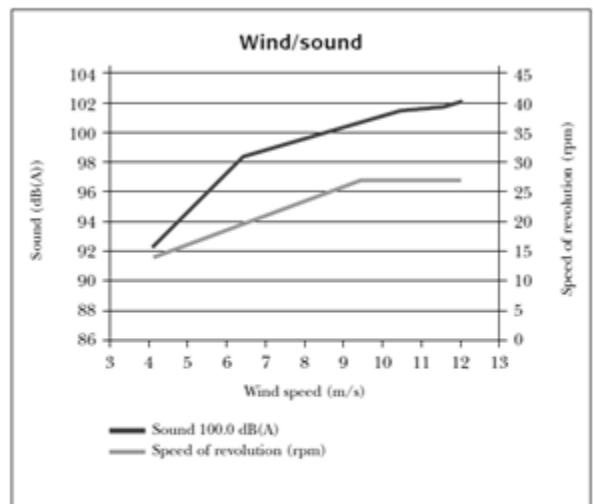
Technical specifications



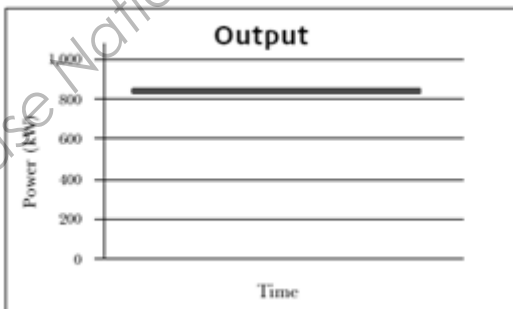
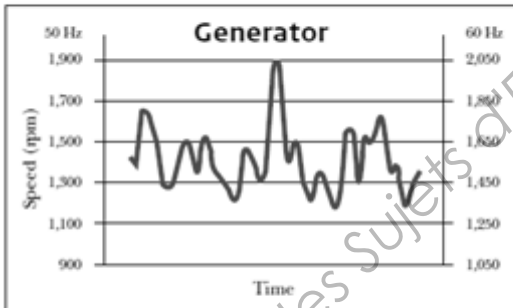
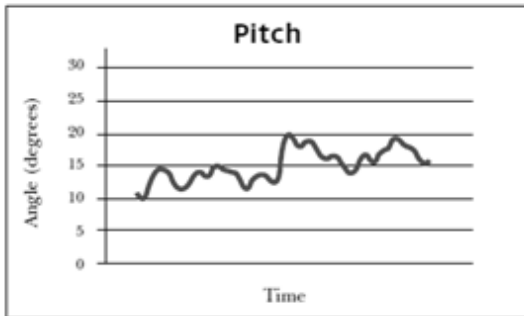
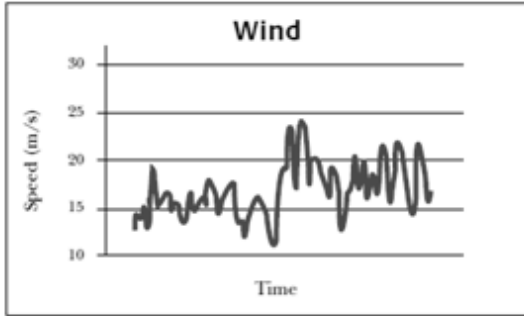
- | | | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> 1 Ultrasonic wind sensor 2 Service crane 3 VMP-Top controller with converter 4 OptiSpeed® generator 5 Pitch cylinder | <ul style="list-style-type: none"> 6 Oil and water coolers 7 Gearbox 8 Main shaft 9 Pitch system 10 Blade hub | <ul style="list-style-type: none"> 11 Blade bearing 12 Blade 13 Rotor lock system 14 Hydraulic unit 15 Torque arm | <ul style="list-style-type: none"> 16 Machine foundation 17 Mechanical disc brake 18 Yaw gear 19 Composite disc coupling |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|



The figure above illustrates the power curves at different sound levels for the V52-850 kW turbine, which is equipped with OptiSpeed®.



The sound output level can be adjusted by varying the revolution speed of the turbine as illustrated in the figure above. It clearly shows the sound level advantages of lower speeds of revolution because the sound level is approximately 7 dB(A) lower at 4 m/s than at 8 m/s. For other sound levels, the benefit can be as much as 10 dB(A). Please note that a decrease of 3 dB(A) represents a halving of the sound level.



Rotor

Diameter: 52 m
 Area swept: 2,124 m²
 Nominal revolutions: 26 rpm
 Operational interval: 14.0-31.4 rpm
 Number of blades: 3
 Power regulation: Pitch/OptiSpeed®
 Air brake: Full blade pitch

Tower

Hub height: 40 m, 44 m, 49 m, 55 m,
 60 m, 65 m, 74 m, 86 m

Operational data

Cut-in wind speed: 4 m/s
 Nominal wind speed: 16 m/s
 Cut-out wind speed: 25 m/s

Generator

Type: Asynchronous with OptiSpeed®
 Nominal output: 850 kW
 Operational data: 50 Hz/60 Hz
 690 V

Gearbox

Type: 1 planet step/2-step
 parallel axle gears

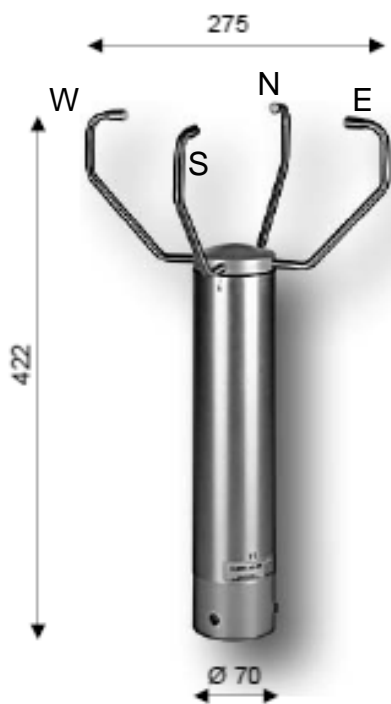
Control

Type: Microprocessor-based monitoring of all turbine functions as well as OptiSpeed® output regulation and OptiTip® pitch regulation of the blades.

Weight

Nacelle: 22 t
 Rotor: 10 t

Annexe 2 : Capteur Vent



Les capteurs adaptés au givrage utilisent un anémomètre ultrasonique pouvant se réchauffer facilement. Ces équipements sont principalement destinés à être installés en montagne.

Les capteurs réalisent des mesures de vitesse et de direction du vent 2 fois par secondes et effectuent les calculs nécessaires à l'exploitation du vent conformément aux règles de l'OMM* et de l'OACI**.

Principaux calculs effectués :

Vitesse maximum du vent et sa direction associée, calculées sur une et 10 minutes

Variation calculée sur 10 minutes de la vitesse et de la direction du vent

Vitesse moyenne du vent et l'argument du vecteur vent moyen sur 2 et 10 minutes

Vent passé sur une minute

Écart type de la vitesse et de la direction du vent sur 10 minutes

Indicateur de grain.

Les capteurs intelligents gèrent le protocole Cibus et transmettent les données à un afficheur pour visualiser les valeurs instantanées du vent

Câble : longueur 10m

Cette chaîne de mesure du vent est utilisée dans le réseau national de Météo-France

Caractéristiques du capteur :

Vitesse :

Étendue de mesure : 0-65m/s
Précision : ± 0.1m/s jusqu'à 5m/s
± 2% au-delà de 5m/s

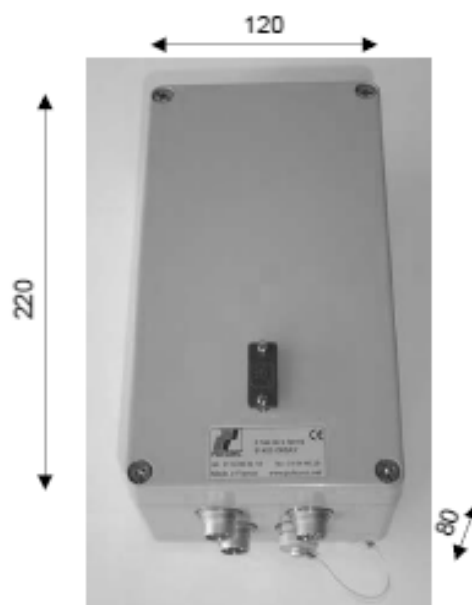
Direction :

Étendue de mesure : 0-360°
Précision : ± 1°

Domaine d'utilisation : -40°C70°C

Réchauffage : 24V - 70W

Fixation sur tube 1^{1/2}" (40/49 en mm)



*OMM Organisation Météorologique Mondiale
**OACI Organisation de l'Aviation Civile Internationale



Figure 1: Le capteur vent : vitesse et direction.

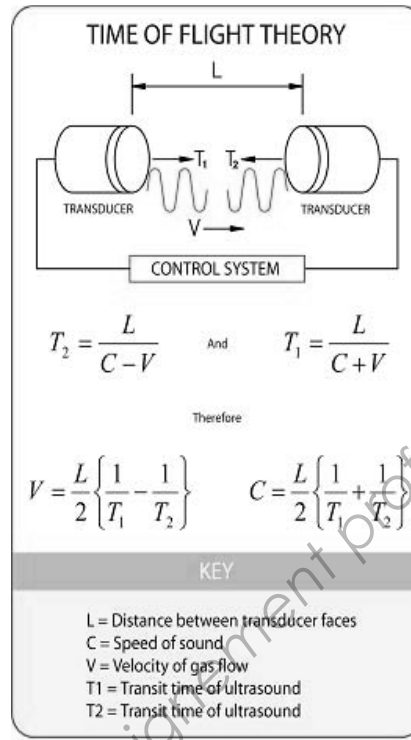


Figure 2 : Calcul de la vitesse et direction du vent.

Extrait du manuel d'utilisation du capteur.

Communication

Protocole CIBus :

Le protocole CIBus est décrit dans le

Cahier des Spécifications Techniques

YB8 1400 B 7400

Protocole d'échange réduit CIBUS entre capteurs interactifs et systèmes d'acquisition

Requête du centralisateur

<ENQ> N <SOH> 0 00 A <EOT><CRC>

Le caractère N prend successivement les valeurs .0. .1,9. .A. B, .M. .0....

4.4.1.2. Réponse transmise par le capteur, CIBus version 3

<SOH>Raa A<STX>Cx VSS DFFF1 DFFF2 DFFF3 dddd fffffff DFFF4 VVVV<EOT><CRC>

Raa Adresse physique du capteur dans le réseau CIBUS

A Identification de message de données minutes

Cx Identification logique de la mesure de vent

VSS Version du message du capteur vent/status public/status privé (V= 3)

DFFF1 Vent moyen 10 min

DFFF2 Vent instantané : maximum dans les 10 min précédentes

DFFF3 Vent moyen 2 min

dddd Vent moyen 3 s : domaine de variation des directions dans les 10 min précédentes

ffffff Vent moyen 3 s : domaine de variation des vitesses dans les 10 min précédentes

DFFF4 Vent instantané : maximum dans la minute précédente ;

VVVV Vent passé dans la minute précédente.

4.4.1.3. Réponse transmise par le capteur, CIBus version 4

<SOH>Raa A<STX>Cx VSS DFFF1 DFFF2 DFFF3 dddd fffffff DFFF4 G sfff sdd<EOT><CRC>

Raa Adresse physique du capteur dans le réseau CIBUS

A Identification de message de données minutes

Cx Identification logique de la mesure de vent

VSS Version du message du capteur vent/status public/status privé (V=4)
DDFF1 Vent moyen 10 min
DDFF2 Vent instantané : maximum dans les 10 min précédentes
DDFF3 Vent moyen 2 min
ddDD Vent moyen 3 s : domaine de variation des directions dans les 10 min précédentes
fffFFF Vent moyen 3 s : domaine de variation des vitesses dans les 10 min précédentes
DDFF4 Vent instantané : maximum dans la minute précédente ;
G Indicateur de grain, codé 1 si présent, 0 sinon
sdff Ecart type de la force du vent exprimé en 1/100ème de m/s
sdd Ecart type de la direction en °

Codage status public

Codage	Désignation
B1B0 = 00	Capteur HS
B1B0 = 01	DD HS
B1B0 = 10	FF HS
B1B0 = 11	Capteur OK
B2 = 0	EEPROM_CI non programmée ou HS.
B3 = 1	Forcé à 1
B4 = 0	Alimentation locale < 6V
B5 = 1	Forcé à 1
B6 = 1	Forcé à 1
B7 = 1	Non utilisé, forcé à 1

Codage status privé

Codage	Désignation
B0 = 0	Courant consommé > 6mA
B1 = 1	Forcé à 1
B2 = 1	Forcé à 1
B3 = 1	Forcé à 1
B4 = 1	Forcé à 1
B5 = 1	Forcé à 1
B6 = 1	Forcé à 1
B7 = 0	Alerte grain

Diffusion de données sur le 2ème canal

Ce 2ème canal permet de connecter des afficheurs autonomes.

L'unité centrale émet spontanément toutes les 500 ms une trame dont le format est le suivant :

<SOH>RaaTCiDATAcxDATA<EOT>CRC

R numéro de réseau.

aa adresse dans le réseau.

T identificateur de trame.

Ci,Cx identificateur de données vents (Ci : données instantanées, Cx : autres données).

DATA données.

La direction DD ou DDi est exprimée en dizaine de degrés (2 caractères ASCII),

une direction invalide ou non calculée est codée : //.,

une direction non significative ou dite de vent calme est codée.00

La force FFF ou FFi est exprimée en dixième de m/s (3 caractères ASCII),

une force invalide ou non calculée est codée : ///

Le vent passé VVVV est exprimé en mètres (caractère ASCII),
un vent passé non calculé ou invalide est codé : ///

L'émission de données s'effectue par l'émission de 8 trames consécutives.

Trame n°1 : trame d'identification avec les arguments suivants :

<SOH>Raa I CiDDFFi CI QFU<EOT>CRC

DD direction instantanée.

FFi force instantanée.

QFU nom du site exprimé sur 4 caractères ASCII.

Trame 2 : trame de status avec les arguments suivants :

<SOH>Raa S CiDDFFi CS VSS<EOT>CRC

DD direction instantanée.

FFi force instantanée.

V version logicielle du capteur (3 ou 4)

SS status public et status privé.

Trame n°3 à 8 : trame des mesures avec les arguments suivants :

<SOH>Raa M CiDDFFi Cx DATA<EOT>CRC

DD direction instantanée.

FFi force instantanée.

X et **DATA** suivant le tableau suivant.

X	DATA
A	DDFFF (vent moyen 2')
B	DDFFF vent moyen 10')
C	ddDD (secteur de variabilité direction moyenne 3 » sur 10')
D	fffff (secteur de variabilité force et moyenne 3 » sur 10')
E	DDFFF (max vent instantané. 1')
F	VVVV (vent passé sur la minute)

Les paramètres force et direction instantanés transmis dans chaque trame ne sont pas affectés par la condition dite de vent calme.

CRC = CRC sur 16 bits avec polynôme générateur du type

$$X^{16} + X^{12} + X^5 + 1.$$

Le CRC est calculé du caractère SOH au caractère EOT inclus.

Caractéristiques:

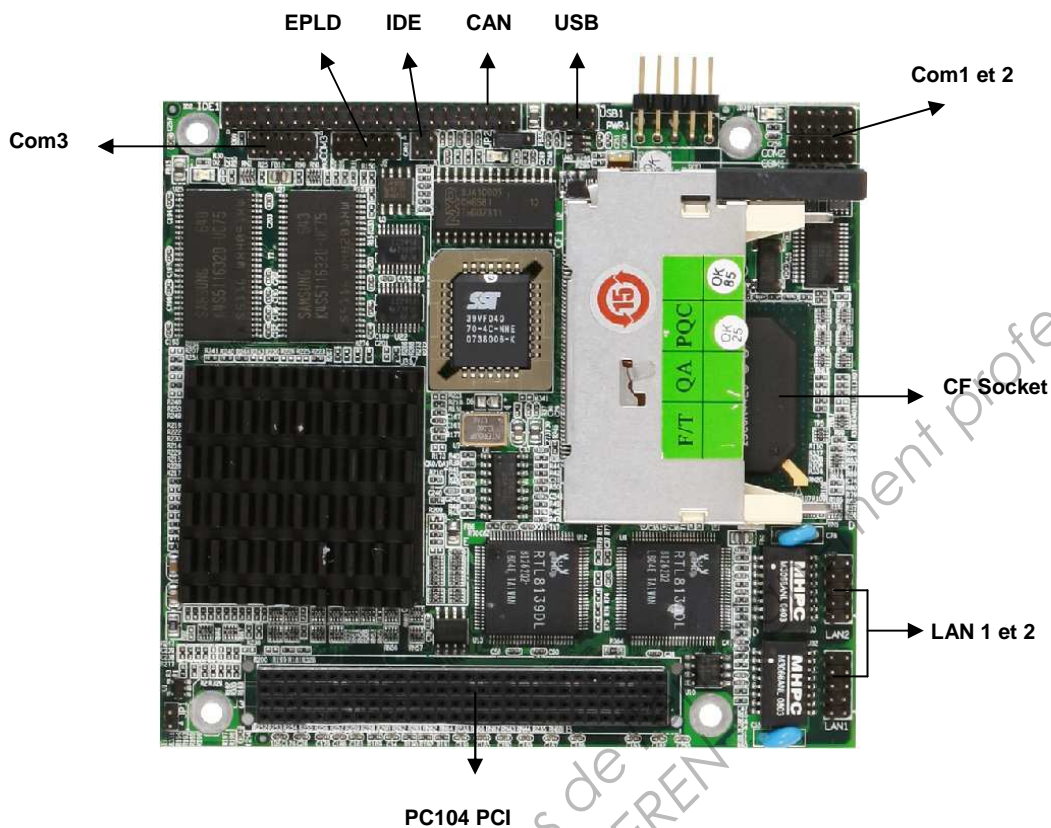
Modulation FSK-V23 centrée à 1700 Hz avec déplacement de fréquence de ± 400 Hz. Au repos, la fréquence de 1300 Hz est transmise en permanence. Un boîtier supplémentaire permet d'adapter cette modulation à une liaison série RS232. Les paramètres de transmission sont : 1200 bauds, 8 bits, sans parité, 1 stop.

Annexe 3 : Fichier XML, Parser TinyXML, Librairie STL

Extrait du fichier XML: evt_arret.xml

```
<?xml version="1.0" encoding="iso-8859-1" standalone="no"?>
<EVT_ARRET>
  <EOLIENNE NUM_EOLIENNE="1">
    <DATE>14:57:36</DATE>
    <VITESSE_VENT>4.8</VITESSE_VENT>
    <ORIENTATION>31:54</ORIENTATION>
    <VITESSE_ORIENTATION>0.3</VITESSE_ORIENTATION>
  </EOLIENNE>
  <EOLIENNE NUM_EOLIENNE="9">
    <DATE>14:59:37</DATE>
    <VITESSE_VENT>4.2</VITESSE_VENT>
    <ORIENTATION>28:60</ORIENTATION>
    <VITESSE_ORIENTATION>0.5</VITESSE_ORIENTATION>
  </EOLIENNE>
  <EOLIENNE NUM_EOLIENNE="6">
    <DATE>15:29:13</DATE>
    <VITESSE_VENT>5.6</VITESSE_VENT>
    <ORIENTATION>34:12</ORIENTATION>
    <VITESSE_ORIENTATION>1.3</VITESSE_ORIENTATION>
  </EOLIENNE>
  <EOLIENNE NUM_EOLIENNE="2">
    <DATE>16:14:18</DATE>
    <VITESSE_VENT>5.3</VITESSE_VENT>
    <ORIENTATION>29:35</ORIENTATION>
    <VITESSE_ORIENTATION>0.6</VITESSE_ORIENTATION>
  </EOLIENNE>
  <EOLIENNE NUM_EOLIENNE="7">
    <DATE>16:35:54</DATE>
    <VITESSE_VENT>5.8</VITESSE_VENT>
    <ORIENTATION>18:36</ORIENTATION>
    <VITESSE_ORIENTATION>0.1</VITESSE_ORIENTATION>
  </EOLIENNE>
  <EOLIENNE NUM_EOLIENNE="1">
    <DATE>17:18:04</DATE>
    <VITESSE_VENT>3.8</VITESSE_VENT>
    <ORIENTATION>25:21</ORIENTATION>
    <VITESSE_ORIENTATION>1.7</VITESSE_ORIENTATION>
  </EOLIENNE>
  <EOLIENNE NUM_EOLIENNE="9">
    <DATE>18:03:45</DATE>
    <VITESSE_VENT>4.5</VITESSE_VENT>
    <ORIENTATION>13:37</ORIENTATION>
    <VITESSE_ORIENTATION>0.8</VITESSE_ORIENTATION>
  </EOLIENNE>
</EVT_ARRET>
```


Annexe 4 : PC104 – Carte Mère 104-3511CD2N et FreeRTOS



Features

- Onboard MOTOROLA MPC8245 350MHz processor - **RISC design**
- Onboard 256MB system memory, 64MB Flash ROM
- Two 10/100Mbps Ethernet controllers
- One PCI-104 connector, one CAN bus connector
- Three serial ports, two USB1.0 ports
- +5V, +12V power supply

Operating system

Embedded OS Vxworks/Linux patch FreeRTOS

Exemple d'applications FreeRTOS:

Une application FreeRTOS aura généralement la forme suivante :

- Initialisation sommaire du matériel (assez pour les deux étapes suivantes)
- Copie de la section .data et initialisation de la section .bss
- Exécution du programme principal (à partir de l'étape suivante)
- Fin de l'initialisation du matériel (horloges supplémentaires, wait states, etc.)
- Initialisation des périphériques
- Création des structures FreeRTOS (tâches et structures de contrôle)
- Démarrage de l'ordonnanceur

Ce premier exemple listing 1 est une application simple mettant en œuvre deux tâches indépendantes **fonctionnant concurremment. Le but est de montrer comment à l'installation d'un environnement multitâche, on définit et créer les tâches, lancer un ordonnanceur de préemption, aussi bien que** montrer quelques fonctions de contrôle de temps de tâche.

Dans le premier exemple il y a deux tâches indépendantes La première appelée LEDS fait juste clignoter des leds à une fréquence fixe. La seconde appelée 'la TEMPERATURE' demande périodiquement au capteur externe de température une mesure, pour l'afficher sur un PC via une liaison série. Les deux tâches fonctionnent avec la même priorité et l'ordonnanceur peut anticiper les tâches pour faire exécuter l'autre

La tâche **vLEDTask** configure d'abord le port de sortie matériel et initialise une variable PRINCIPALE elle entre alors dans la boucle infinie où elle active les LEDS selon la variable d'état, incrémente cette variable, demande ensuite à l'ordonnanceur de bloquer la tâche pour un temps donné.

La tâche **vTempTask** initialise d'abord la communication UART avec de capteur de température, Le capteur température demande un échantillonnage continu et entre dans la boucle infinie. Dans la boucle, la tâche lit la température du capteur, l'envoie sur le port série puis attend un temps donné.

La fonction principale du programme doit ajouter les tâches à l'ordonnanceur et le commencer. Ceci est fait utilisant les fonctions de vTaskStartScheduler et le xTaskCreate. Voici le code complet du fichier de main.c.

```
/**
 * \file main.c
 */
#include "Projet.h"

/* Hardware initialization */
static void prvSetupHardware( void );
static void vLEDTask(void* pvParameters);
static void vTempTask(void* pvParameters);

/**
 * putchar function required by stdio.h module to be able to use printf()
 */
int putchar(int c)
{
    return uart0_putchar(c);
}

/* The main function. */
int main( void )
{
    /* Setup the hardware. */
    prvSetupHardware();

    /* Add the two tasks to the scheduler */
    xTaskCreate(vLEDTask, "LED", configMINIMAL_STACK_SIZE, NULL, 1, NULL );
    xTaskCreate(vTempTask, "Temperature", configMINIMAL_STACK_SIZE, NULL, 1, NULL );
```

```

/* Start the scheduler. */
vTaskStartScheduler();

/* As the scheduler has been started we should never get here! */
return 0;
}

/* Initialize the main hardware parameters. */
static void prvSetupHardware( void )
{
.....

/* Rôle de la tâche*/
.....

}

/* The LED task function.It increments a variable and displays its value on the 3 LEDs, then waits for a given
delay and start over again.*/
* \param pvParameter NULL is passed as parameter.
*/
static void vLEDTask(void* pvParameters)
{
.....

/* Rôle de la tâche*/
.....
}

/* The temperature measurement task function. It reads the temperature from the sensor and print it on the serial
port.*/
* \param pvParameters NULL is passed, unused here.
*/
static void vTempTask(void* pvParameters)
{
.....

/* Rôle de la tâche*/
.....
}

```

Listing 1

Task synchronization

Ce deuxième exemple est une application simple montrant une tâche qui se synchronise par l'intermédiaire d'un sémaphore avec une routine de service s'interrompant..

La solution utilisée est un sémaphore binaire.

Sémaphores binaires sont utilisées à la fois pour l'exclusion mutuelle et à des fins de synchronisation.

Sémaphores binaires et les mutex sont très similaires, mais ont quelques différences subtiles: Mutex inclue un mécanisme d'héritage de priorité, les sémaphores binaires n'en ont pas. Cela rend les sémaphores binaires le meilleur choix pour la mise en œuvre de la synchronisation (entre les tâches ou entre les tâches et une interruption) et des mutex le meilleur choix pour la mise en œuvre simple, l'exclusion mutuelle. La description de la façon dont un mutex peut être utilisé comme un mécanisme d'exclusion mutuelle détient également des sémaphores binaires

Les Fonctions API de sémaphores permettent un temps de blocage à préciser. Le temps de blocage indique le nombre maximal de «tiques» que la tâche doit attendre dans l'état bloqué lorsque l'on tente de «prendre» un sémaphore, le sémaphore ne doit pas être immédiatement disponible. Si une tâche sur le même sémaphore que la tâche en cours tente d'accéder à la même zone alors la tâche avec la plus haute priorité sera la tâche qui sera servie la prochaine fois que le sémaphore devient disponible.

Les macros pdTRUE indique que le sémaphore est occupé et pdFALSE que le sémaphore est libre

Le paramètre xBlockTime de la première macro est le nombre maximal de ticks pour attendre que le sémaphore soit disponible. Il correspond au même paramètre avec les fonctions de file d'attente.

La fonction xSemaphoreGiveFromISR libère le sémaphore en présence d'une interruption.

Voici le code complet.

```
#include "Projet.h"

/* Function Prototypes */
static void prvSetupHardware( void );
static void vLEDTask(void* pvParameters);
static void rx_char_cb(uint8_t c);

int putchar(int c)
{
    return uart0_putchar(c);
}

/* Global Variables */
xSemaphoreHandle xSemaphore;

/* The main function. */
int main( void )
{
    /* Setup the hardware. */
    prvSetupHardware();

    /* Create the Semaphore for synchronization between UART and LED task */
    vSemaphoreCreateBinary( xSemaphore)

    /* Add the only task to the scheduler */
    xTaskCreate(vLEDTask, "LED", configMINIMAL_STACK_SIZE, NULL, 1, NULL );

    /* Start the scheduler. */
    vTaskStartScheduler();
}
```

```

    /* As the scheduler has been started we should never get here! */
    return 0;
}
/* Initialize the main hardware parameters. */
static void prvSetupHardware( void )
{
    /* Stop the watchdog timer. */
    WDTCTL = WDTPW + WDTHOLD;

    /* Setup MCLK 8MHz and SMCLK 1MHz */
    set_mcu_speed_xt2_mclk_8MHz_smclk_1MHz();

    /* Configure the UART module for serial communication */
    uart0_init(UART0_CONFIG_1MHZ_115200);
    uart0_register_callback(rx_char_cb);
    printf("type any char to update the LEDs\r\n");

    /* Enable Interrupts */
    eint();
}

/* The LEDs task function. It waits the semaphore is given, then takes it and update the LEDs
 * \param pvParameters NULL is passed, unused here.
 */
static void vLEDTask(void* pvParameters)
{
    uint16_t leds_state = 0;

    /* Initialize the LEDs */
    LEDS_INIT();

    /* Infinite loop */
    while(1)
    {
        /* Increment the LED state */
        leds_state ++;
        /* Block until the semaphore is given */
        xSemaphoreTake(xSemaphore, portMAX_DELAY);
        /* update the LEDs and loop */
        LEDS_SET(leds_state);
    }
}

/* The callback function called when a char is received. It gives the semaphore.
 * \param c the received char
 */
static void rx_char_cb(uint8_t c)
{
    static portBASE_TYPE xHigherPriorityTaskWoken;
    /* Give the semaphore */
    xSemaphoreGiveFromISR(xSemaphore, &xHigherPriorityTaskWoken);
}

```

Listing 2

Annexe 5 : Architecture réseau Nacelle.

Definition of Terms

CT316	The processor dedicated to VCMS
CT317	Connection box for the RS485 network.
CT5200	High sensitive accelerometer for VCMS
CT5210	Microphone for VCMS
CT5221	Tower accelerometer
CT5230	Low sensitive accelerometer for VCMS
VCMP	See CT316
VCMS	Vestas Condition Monitoring System
VCPM	Vestas Converter Processor Module. Variable speed processor.
VMP	Vestas Multi Processor controller

Processor

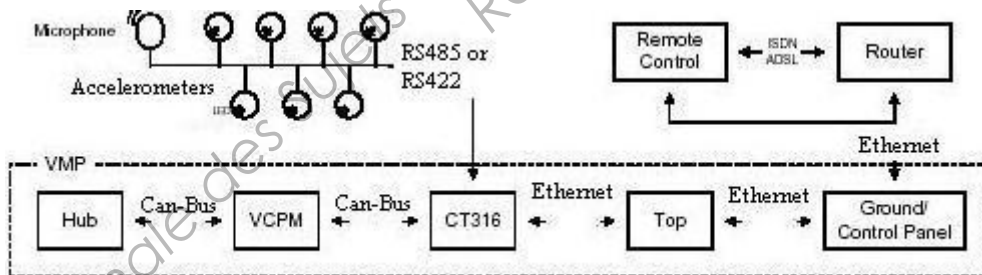
unit:

The hardware consists of 2 CT316 (processor unit), 3 CT317, 6 high sensitive accelerometers (CT5200), two low sensitive accelerometers (CT5230) and a microphone (CT5210).

The CT316 is similar to the other processors in the turbine placed on the ARCnet or Ethernet, which it uses to communicate possible warnings and alarms and receive operational data from the central turbine monitor (VMP).

The CT316 handles all calculations in connection with VCMS and this has the advantage that the significantly large number of calculations does not burden the rest of the processors in the VMP. The processor contains 16 MB RAM and 32 MB flash disc.

Therefore, a second CT316 is placed in the hub in order to collect and process data from these sensors. This is only a slave processor which only is used to sample data from the nodes and transfer them to the master CT316 in the top control panel.



RS485 or RS422 network:

The network between the accelerometers, the microphone and the CT316 has been chosen especially to improve signal reliability in the noisy environment of the turbine. In order to ensure that the network is free of reflections, the sensors are connected in series and terminated in both ends. However the network will still be functional if e.g. the power is disconnected from one of the sensors. The reason is that there is a direct physical connection through the accelerometer and the transceiver circuit is connected parallel to the network.

By measuring the voltage before every network command broken cables can be found. The CT317 is a connection box, in which it is possible to disable the connection to one or more sensors.

SPECIFICATIONS		RS232	RS423	RS422	RS485
Mode of Operation		SINGLE -ENDED	SINGLE -ENDED	DIFFERENTIAL	DIFFERENTIAL
Total Number of Drivers and Receivers on One Line (One driver active at a time for RS485 networks)		1 DRIVER 1 RECVR	1 DRIVER 10 RECVR	1 DRIVER 10 RECVR	32 DRIVER 32 RECVR
Maximum Cable Length		50 FT.	4000 FT.	4000 FT.	4000 FT.
Maximum Data Rate (40ft. - 4000ft. for RS422/RS485)		20kb/s	100kb/s	10Mb/s-100Kb/s	10Mb/s-100Kb/s
Maximum Driver Output Voltage		+/-25V	+/-6V	-0.25V to +6V	-7V to +12V
Driver Output Signal Level (Loaded Min.)	Loaded	+/-5V to +/-15V	+/-3.6V	+/-2.0V	+/-1.5V
Driver Output Signal Level (Unloaded Max)	Unloaded	+/-25V	+/-6V	+/-6V	+/-6V
Driver Load Impedance (Ohms)		3k to 7k	>=450	100	54
Max. Driver Current in High Z State	Power On	N/A	N/A	N/A	+/-100uA
Max. Driver Current in High Z State	Power Off	+/-6mA @ +/-2v	+/-100uA	+/-100uA	+/-100uA
Slew Rate (Max.)		30V/uS	Adjustable	N/A	N/A
Receiver Input Voltage Range		+/-15V	+/-12V	-10V to +10V	-7V to +12V
Receiver Input Sensitivity		+/-3V	+/-200mV	+/-200mV	+/-200mV
Receiver Input Resistance (Ohms), (1 Standard Load for RS485)		3k to 7k	4k min.	4k min.	>=12k

PCM-36 10 PCM-36 12 PCM-36 14

RS-232/422/485 Module with Isolation Protection

2-port RS-422/485 Module

4-port RS-422/485 High-speed Module



Features

- High speed transmission rate
- Automatic RS-485 data flow control
- Jumper selectable interrupt level
- Supports Windows® 98/2000/XP
- Supports WinCE 3.0, 4.2, 5.0
- Powerful and easy-to-use utility (ICOM Tools)

Specifications

General

- Card Interface PC/104
- Certifications CE
- Connectors 2 x DB9-M
- Ports 2
- Power Consumption +5 V @ 400 mA typical
±12 V @ 950 mA max

Communications

- Channel 1 RS-232, 422, or 485
- Channel 2 RS-422, or RS-485
- Character Length 5, 6, 7, or 8 bits
- IRQ 3, 4, 5, 6, 7, 9
- Parity Even, odd, or none
- Speed 50 bps ~ 115.2 kbps
- Stop Bit 1, 1.5, or 2

Protection

- Isolation Protection 1,000 V_{DC}

Environment

- Humidity (Operating) 0 ~ 90 % RH
- Operating Temperature 0 ~ 65° C (32 ~ 149° F)
- Storage Temperature -40 ~ 85° C (-40 ~ 185° F)

Ordering Information

- PCM-3610-B Isolated RS-232/422/485 Module

Features

- Long distance communication
- Automatic RS-485 data flow control
- Jumper selectable interrupt level
- Supports Windows® 98/2000/XP
- Supports WinCE 3.0, 4.2, 5.0
- Powerful and easy-to-use utility (ICOM Tools)

Specifications

General

- Card Interface PC/104
- Certifications CE
- Connectors 2 x DB9-M
- Indicators Red LED for TX
Green LED for RX
- Ports 2
- Power Consumption +5 V @ 400 mA typical

Communications

- Channel 1 and 2 RS-422, or RS-485
- Character Length 5, 6, 7, or 8 bits
- IRQ 3, 4, 5, 6, 7, 9, 10, 11, 12 or 15
- Parity Even, odd, or none
- Speed 50 bps ~ 115.2 kbps
- Stop Bit 1, 1.5, or 2

Environment

- Humidity (Operating) 0 ~ 90 % RH
- Operating Temperature 0 ~ 65° C (32 ~ 149° F)
- Storage Temperature -40 ~ 85° C (-40 ~ 185° F)

Ordering Information

- PCM-3612-A Dual Port RS-422/485 Module

Features

- Automatic RS-485 data flow control
- Shared IRQ settings for each ports
- LED indicators: TX, RX
- Standard PC ports: COM1, COM2, COM3, COM4 compatible
- Supports Windows® 98/2000/XP
- Supports WinCE 4.2, 5.0
- Powerful and easy-to-use utility (ICOM Tools)

Specifications

General

- Card Interface PC/104
- Certifications CE
- Connectors 4 x DB9-M
- Ports 4
- Power Consumption +5 V @ 450 mA

Communications

- Data Bits 5, 6, 7, 8
- I/O Address Range 0 x 000 ~ 0 x 3F8
- IRQ 3, 4, 5, 6, 7, 9, 10, 11, 12, or 15
- Parity Even, odd, or none
- Data Signals RS-422: TxD+, TxD-, RxD+, RxD-, CTS+, CTS-, RTS+ and RTS-
RS-485: DATA+, DATA-, CTS+, CTS-
- Speed 50 bps ~ 921.6 kbps
- Stop Bits 1, 1.5, 2

Protection

- EFT Protection 1000 V_{DC}
- Termination Resistor 120 W

Environment

- Humidity (Operating) 0 ~ 90 % RH
- Operating Temperature 0 ~ 65° C (32 ~ 149° F)
- Storage Temperature -40 ~ 85° C (-40 ~ 185° F)

Ordering Information

- PCM-3614-A 4-port RS-422/485 High-speed Module

PCM-3610 PC/104 Isolated Dual-port RS-232 and RS-485/422 Module



Initial inspection

We carefully inspected the PCM-3610 both mechanically and electrically before we shipped it. It should be free of marks and scratches and in perfect electrical order on receipt.

Handle the board only by its edges. The static charge on your body may damage its integrated circuits. Keep the card in its anti-static package whenever it is not installed. You can use this package to return the card if it should need repair.

Switches and jumpers

The following chart shows the switches and jumpers corresponding to each serial interface channel:

Ch. 1	Ch. 2	Function
SW1	SW2	I/O base address
JP2	JP3	Interrupt level
JP4	JP5	RS-485 or RS-422
JP10	N/A	RS-232 or RS-485/422

Switch locations appear in the figure below.

RS-232/422/485 selection (JP4, 5, 10)

Channel 1

- RS-232 JP10 up to 232
- RS-422 JP10 down to 485/422
JP4 right to RS-422
- RS-485 JP10 down to 485/422
JP4 left to RS-485

Note: You must connect channel 1's external signal cable to JP7 for RS-232 or JP8 for RS-422/485.

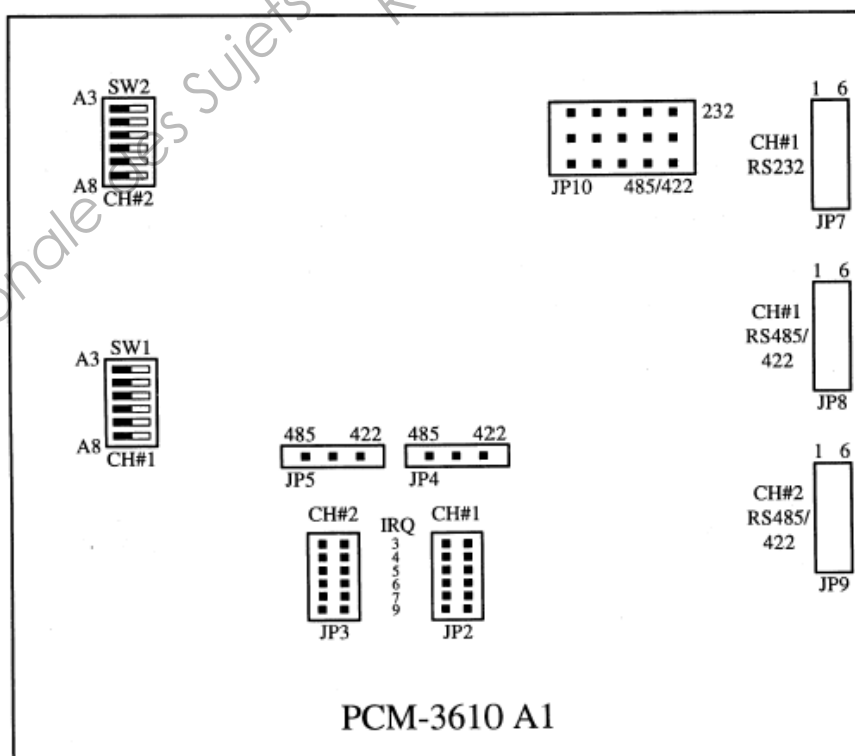
Channel 2

- RS-422 JP5 right to RS-422
- RS-485 JP5 left to RS-485

Base address (SW1 and SW2)

Two 6-position DIP switches select the I/O port base address for each channel. SW1 controls Ch. 1, and SW2 controls Ch. 2. See the figure below for switch locations.

You can set the base address anywhere from hex 200 to 3F8. The default settings are 3F8 (COM1) for Channel 1 and 2F8 (COM2) for Channel 2. The following table shows switch settings for various base addresses:



Module I/O addresses (SW1 and SW2)						
Range (hex)	Switch position					
	1	2	3	4	5	6
200 - 207	○	○	○	○	○	○
208 - 20F	●	○	○	○	○	○
!						
2E8 - 2EF	●	○	●	●	●	○
* 2F8 - 2FF (COM2)	●	●	●	●	●	○
3E8 - 3EF	●	○	●	●	●	●
* 3F8 - 3FF (COM1)	●	●	●	●	●	●

○ = On ● = Off * = defaults

NOTE: Switches 1-6 control the PC bus address lines as follows:

Switch	1	2	3	4	5	6
Line	A3	A4	A5	A6	A7	A8

Interrupt level-IRQ (JP2 and JP3)

You can set each port for any interrupt level from 3 to 9, except 8. Jumper JP2 controls Ch. 1, and JP3 controls Ch. 2. Simply short the pins on the jumper corresponding to the interrupt level (see figure below).

JP2 (Ch 1, COM1) IRQ JP3 (Ch 2, COM2) IRQ



If you use the ports as standard COM1 and COM2, you will need to set Channel 1 to IRQ 4 and Channel 2 to IRQ 3.

Baud rate	Divisor
50	2304
75	1536
110	1047
133.5	857
150	768
300	384
600	192
1200	96
1800	64
2000	58
2400	48
3600	32
4800	24
7200	16
9600	12
19200	6
38400	3
56000	2
115200	1

LCR : line control register

LCR : line control register				
Bit	Value		Description	
0,1	Bit 1	Bit 0	Data word length	
	0	0	5 bits	
	0	1	6 bits	
	1	0	7 bits	
	1	1	8 bits	
2	0		1 stop bit	
	1		1.5 stop bits (5 bits word)	
	1		2 stop bits (6, 7 or 8 bits word)	
3,4,5	Bit 5	Bit 4	Bit 3	No parity
	x	x	0	Odd parity
	0	0	1	Even parity
	0	1	1	High parity (stick)
	1	0	1	Low parity (stick)
6	0		Break signal disabled	
	1		Break signal enabled	
7	0		DLAB : RBR, THR and IER accessible	
	1		DLAB : DLL and DLM accessible	

PCM-3680 PC/104 Dual Port CAN Interface Module



PCM-3680 PC/104 Dual Port CAN Interface Module

Jumper Setting

The PCM-3680 is a special purpose communication card that brings the Control Area Network to your PC. With the built-in CAN controller, the PCM-3680 provides bus arbitration and error detection with automatic transmission repeat function. This drastically avoids data loss and ensures system reliability. The on-board CAN controllers are located at different positions in the memory. You can run both CAN controllers at the same time, independently. The PCM-3680 operates at baud rates up to 1 Mbps and can be installed directly into the expansion slot of your PC.

Control Area Network

The CAN (Control Area Network) is a serial bus system especially suited for networking "intelligent" I/O devices as well as sensors and actuators within a machine or plant. Characterized by its multi-master protocol, real-time capability, error correction, high noise immunity, and the existence of many different silicon components, the CAN serial bus system, originally developed by Bosch for use in automobiles, is increasingly being used in industrial automation.

Direct Memory Mapping

The PCM-3680 is assigned with memory address, which allows direct access to the CAN controller. This is the simplest and fastest way of programming any board in a PC because the board is regarded as standard RAM.

Optical Isolation Protection

On-board optical isolators protect your PC and equipment against damage from ground loops, increasing system reliability in harsh environments.

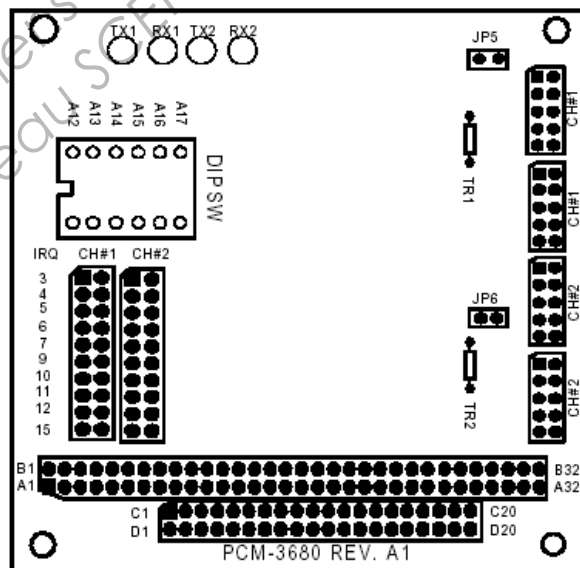
Specifications

- Ports: 2
- CAN controller: SJA-1000
- CAN transceiver: 82C250
- Signal support: CAN-L, CAN-H
- Memory address: From C800H to EF00H
- IRQ: 3, 4, 5, 6, 7, 9, 10, 11, 12, 15
- Isolation voltage: 1000 V_{DC}
- Power consumption: +5 V @ 400 mA typical, 950 mA max.
- Connectors: Dual DB-9 male connectors
- Operating temperature: 32 to 122° F (0 to 50° C)
- PC/104 form factor: 3.6" x 3.8" (90 mm x 96 mm)
- Shipping weight: 0.9 lb (0.4 kg)

Features

- Operates 2 separate CAN networks at the same time
- High speed transmission up to 500 kbps
- 16 MHz CAN controller frequency
- Takes a 4 KB address space, 40 base address adjustable in steps from C800H up to EF00H
- Optical isolation protection of 1000 V_{DC} ensures system reliability
- Wide IRQ selection for each port includes: IRQ 3, 4, 5, 6, 7, 9, 10, 11, 12, 15
- LED indicates Transmit/Receive status on each port
- Direct memory mapping enables speedy access to the CAN controllers
- C library and examples included

Jumper & Switch Locations



Wiring

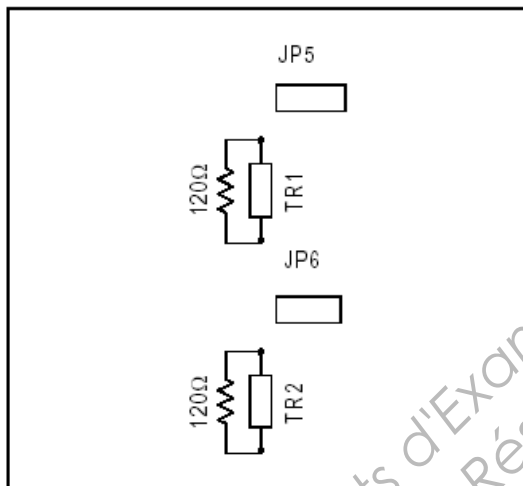
Pin Assignments

The following figure shows the pin assignments for the card's DB-9 connectors.

Termination Resistor Setup

Termination resistors are factory-installed to allow for impedance matching. These resistors can be enabled by using jumpers number 5 and 6 (shown below). Jumper 5 enables the terminal resistor for port 1, while jumper 6 enables the terminal resistor for port 2. The value of the resistor equals the characteristic impedance of the signal wires (approximately 120 Ω)

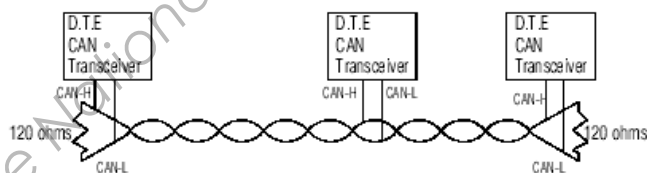
The following figure shows resistor placements. Port TR1 enables jumper JP5, and TR2 enables jumper JP6



CAN termination resistor installation

CAN signal wiring

The CAN standard supports half-duplex communication. This means that just two-wires are used to transmit and receive data.



Wiring topography

Wiring connections are as follows:

PCM-3860	DTE (male DB-9)	Terminal DTE
Pin	Signal	Signal
7	CAN-H	CAN-H
3	GND	GND
2	CAN-L	CAN-L

Register Structure

This appendix gives a brief description of the CAN controller registers. For more detailed information, please refer to the *Stand-alone CAN-controller Data Book* from Philips Semiconductors Microcontroller Products. (You may also find the information on the enclosed disk under the "Manual" directory, in the Word 6.0 file: REGISTER.DOC)

CAN Controller Address Allocation

Philips PCX82C200 CAN Controller
ADDRESS

Register address map

0	CONTROL	control segment
1	COMMAND	
2	STATUS	
3	INTERRUPT	
4	ACCEPTANCE CODE	
5	ACCEPTANCE MASK	
6	BUS TIMING 0	
7	BUS TIMING 1	
8	OUTPUT CONTROL	
9	TEST	

10	IDENTIFIER	descriptor	transmit buffer
11	RTR BIT, DATA LENGTH CODE		
12	BYTE 1	data field	
13	BYTE 2		
14	BYTE 3		
15	BYTE 4		
16	BYTE 5		
17	BYTE 6		
18	BYTE 7		
19	BYTE 8		

20	IDENTIFIER	descriptor	receive buffers
21	RTR BIT, DATA LENGTH CODE		
22	BYTE 1	data field	
23	BYTE 2		
24	BYTE 3		
25	BYTE 4		
26	BYTE 5		
27	BYTE 6		
28	BYTE 7		
29	BYTE 8		

Stand-alone CAN controller

SJA1000

6.5 Common registers

6.5.1 BUS TIMING REGISTER 0 (BTR0)

The contents of the bus timing register 0 defines the values of the Baud Rate Prescaler (BRP) and the Synchronization Jump Width (SJW). This register can be accessed (read/write) if the reset mode is active.

In operating mode this register is read only, if the PeliCAN mode is selected. In BasicCAN mode a 'FFH' is reflected.

Table 44 Bit interpretation of bus timing register 0 (BTR0); CAN address 6

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SJW.1	SJW.0	BRP.5	BRP.4	BRP.3	BRP.2	BRP.1	BRP.0

6.5.1.1 Baud Rate Prescaler (BRP)

The period of the CAN system clock t_{sc1} is programmable and determines the individual bit timing. The CAN system clock is calculated using the following equation:

$$t_{sc1} = 2 \times t_{CLK} \times (32 \times BRP.5 + 16 \times BRP.4 + 8 \times BRP.3 + 4 \times BRP.2 + 2 \times BRP.1 + BRP.0 + 1)$$

where t_{CLK} = time period of the XTAL frequency = $\frac{1}{f_{XTAL}}$

6.5.1.2 Synchronization Jump Width (SJW)

To compensate for phase shifts between clock oscillators of different bus controllers, any bus controller must re-synchronize on any relevant signal edge of the current transmission. The synchronization jump width defines the maximum number of clock cycles a bit period may be shortened or lengthened by one re-synchronization:

$$t_{SJW} = t_{sd} \times (2 \times SJW.1 + SJW.0 + 1)$$

6.5.2 BUS TIMING REGISTER 1 (BTR1)

The contents of bus timing register 1 defines the length of the bit period, the location of the sample point and the number of samples to be taken at each sample point. This register can be accessed (read/write) if the reset mode is active.

In operating mode, this register is read only, if the PeliCAN mode is selected. In BasicCAN mode a 'FFH' is reflected.

Table 45 Bit interpretation of bus timing register 1 (BTR1); CAN address 7

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SAM	TSEG2.2	TSEG2.1	TSEG2.0	TSEG1.3	TSEG1.2	TSEG1.1	TSEG1.0

6.5.2.1 Sampling (SAM)

BIT	VALUE	FUNCTION
SAM	1	triple; the bus is sampled three times; recommended for low/medium speed buses (class A and B) where filtering spikes on the bus line is beneficial
	0	single; the bus is sampled once; recommended for high speed buses (SAE class C)

Stand-alone CAN controller

SJA1000

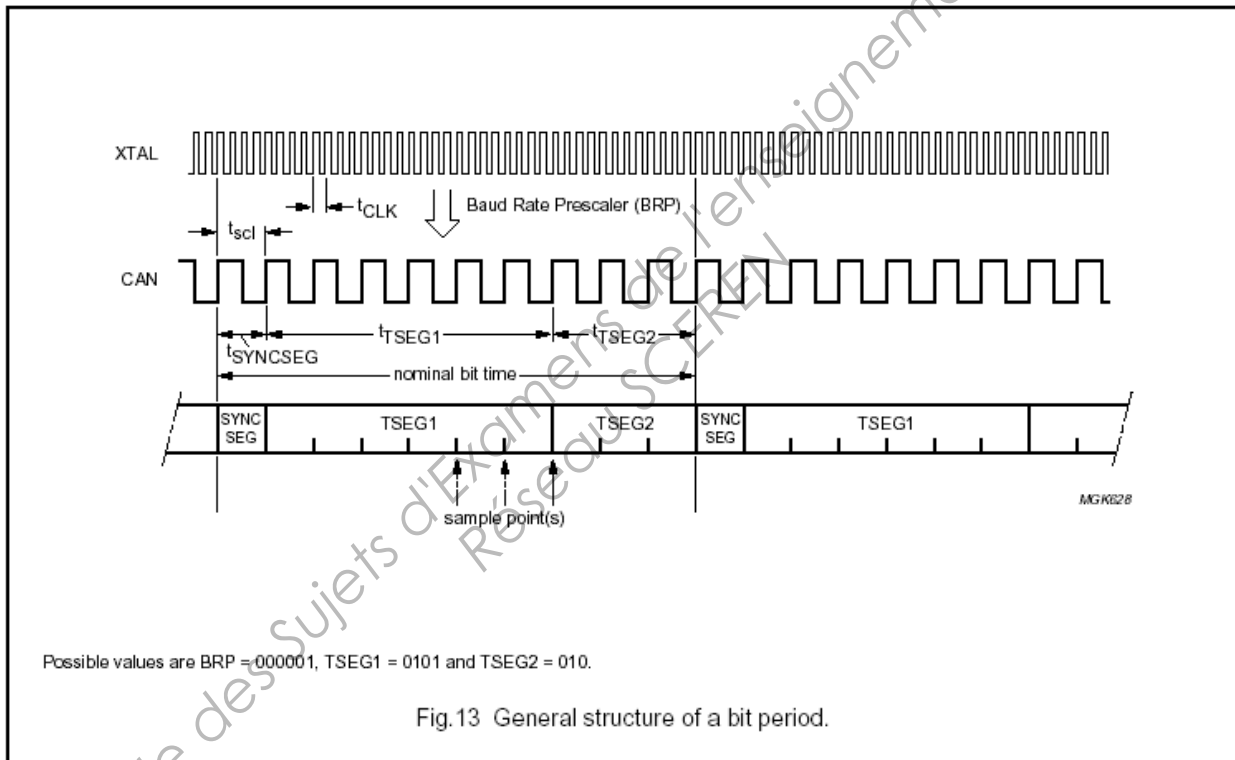
6.5.2.2 Time Segment 1 (TSEG1) and Time Segment 2 (TSEG2)

TSEG1 and TSEG2 determine the number of clock cycles per bit period and the location of the sample point, where:

$$t_{\text{SYNCSEG}} = 1 \times t_{\text{scl}}$$

$$t_{\text{TSEG1}} = t_{\text{scl}} \times (8 \times \text{TSEG1.3} + 4 \times \text{TSEG1.2} + 2 \times \text{TSEG1.1} + \text{TSEG1.0} + 1)$$

$$t_{\text{TSEG2}} = t_{\text{scl}} \times (4 \times \text{TSEG2.2} + 2 \times \text{TSEG2.1} + \text{TSEG2.0} + 1)$$



6.5.3 OUTPUT CONTROL REGISTER (OCR)

The output control register allows the set-up of different output driver configurations under software control.

This register may be accessed (read/write) if the reset mode is active. In operating mode, this register is read only, if the PeliCAN mode is selected. In BasicCAN mode a 'FFH' is reflected.

Table 46 Bit interpretation of the output control register (OCR); CAN address 8

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
OCTP1	OCTN1	OCPOL1	OCTP0	OCTN0	OCPOL0	OCMODE1	OCMODE0

**BREVET DE TECHNICIEN SUPÉRIEUR
INFORMATIQUE ET RÉSEAUX
POUR L'INDUSTRIE ET LES SERVICES TECHNIQUES**

ÉTUDE D'UN SYSTÈME INFORMATISÉ

SESSION 2013

Durée : 6 heures
Coefficient 5

DOCUMENT RÉPONSES

(19 PAGES)

À RENDRE IMPÉRATIVEMENT À LA FIN DE L'ÉPREUVE,
AGRAFÉ À UNE COPIE MODÈLE EN.

B. ANALYSE DU CONTEXTE

B1 :

Nombre d'éoliennes pour une production de 30 MW :

Nombre d'éoliennes supplémentaires:

B2 :

Nombre maximal d'éoliennes:

Puissance maximum du parc? :

B3 :

Variation de température:

Tension (en V)	Courant (en A)	Echauffement linéaire du câble (en °C) pour une section de 1 mm ²
220	136 363	744 000
690	43 478	75000
20000	1 500	90
63000	476	9

Tableau 1

Justification de la tension 63000 Volts? :

Rapport de transformation :

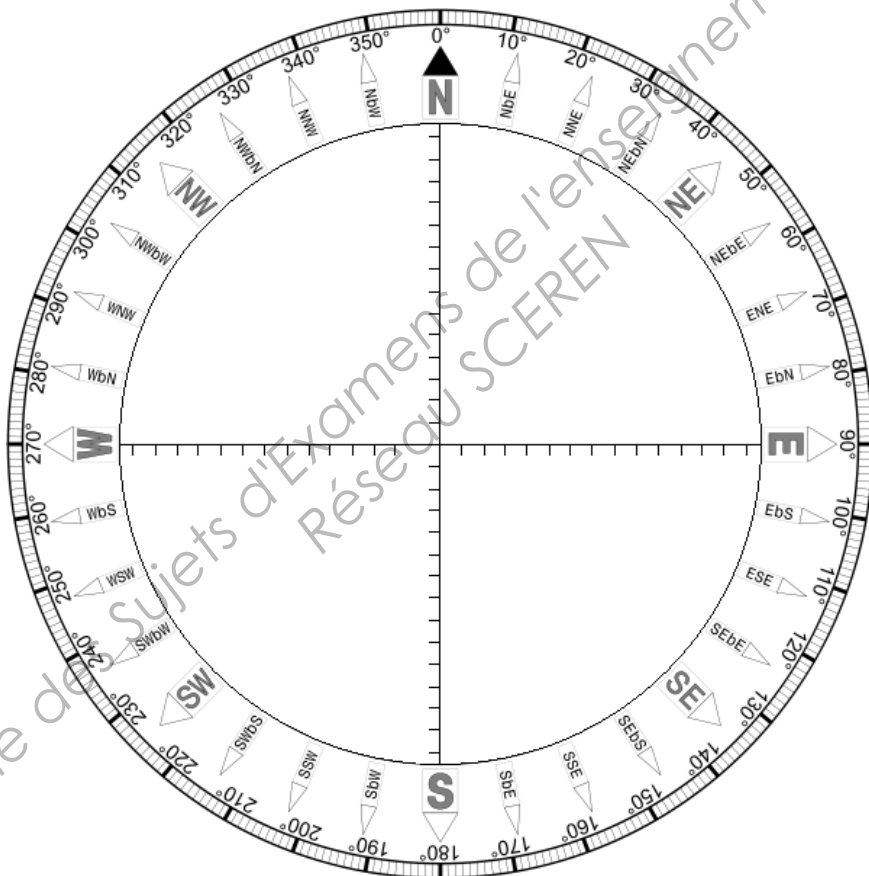
B4:

Vitesse du vent :

Direction du vent	T1 (en μs)	T2 (en μs)	Vitesse des composantes du vent (ms^{-1})
Axe N-S	576,53	600,42	
Axe E-W	568,26	609,66	

Tableau 2

Rose des vents et vitesse résultante :



Résolution formelle :

Calcul du module de vitesse :

Direction du vent :

B5 : Lecture des données du capteur

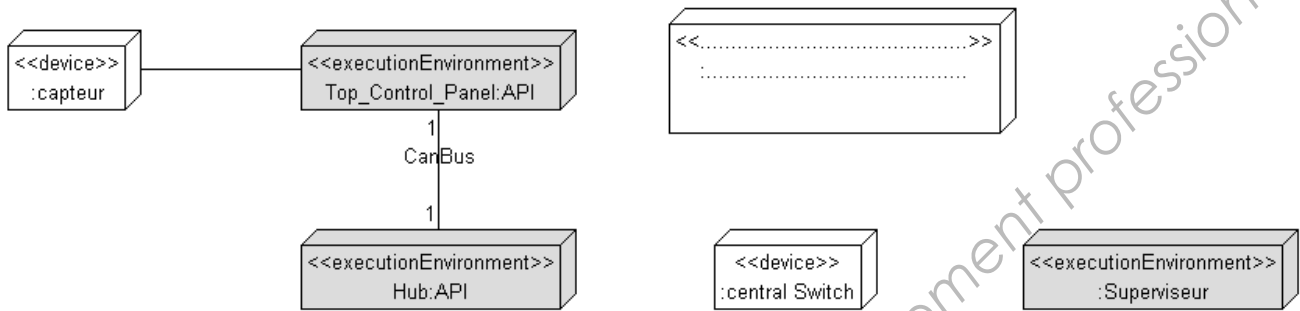
<soh>	104	A	<stx>	C13	♥	30113	22165	31128	2136	101159	22165	0878	<eot>	6
-------	-----	---	-------	-----	---	-------	-------	-------	------	--------	-------	------	-------	---

Remarque codage ASCII ♥ : 0x03
♦ : 0x04

Base Nationale des Sujets d'Examens de l'enseignement professionnel
Réseau SCEREN

C. CONCEPTION DES SYSTÈMES EMBARQUÉS

C1 : Complétez le diagramme de déploiement:



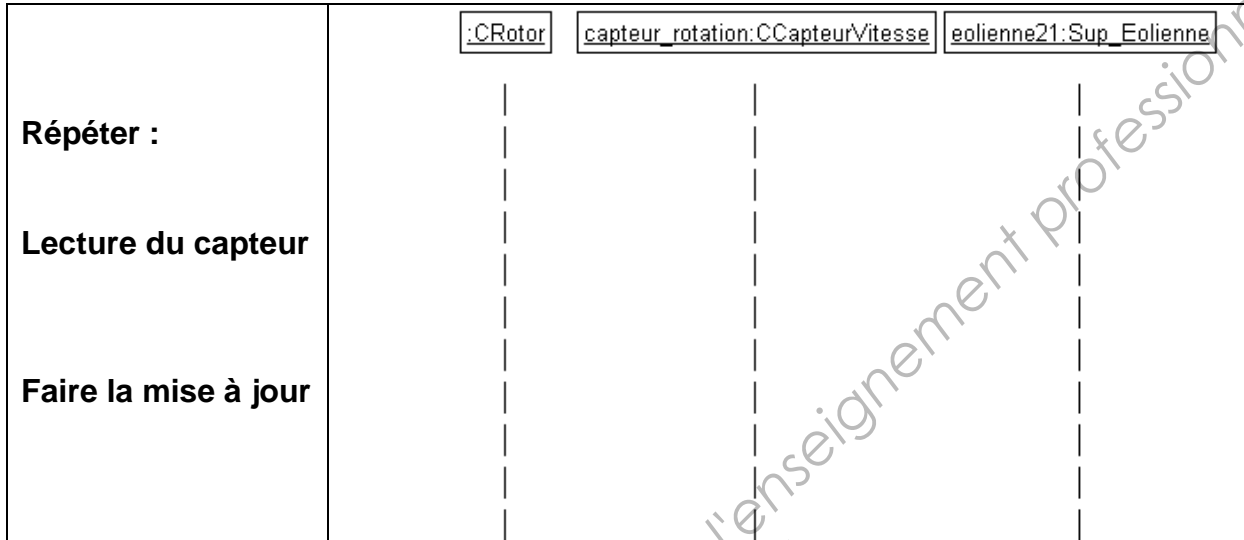
C2 :
void Gestion_Vent_Faible(float vitesse) ;
DEBUT

FIN

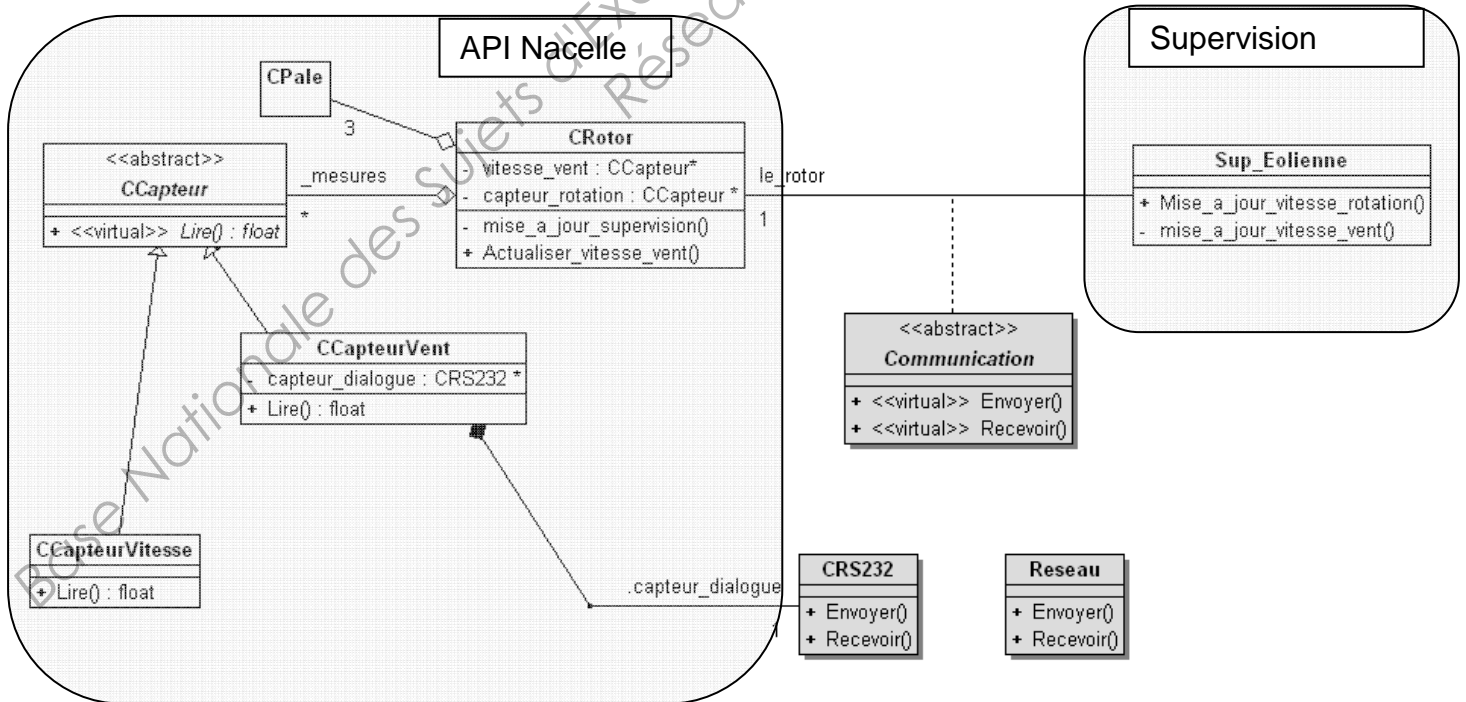
C3 :
Signification du trait en pointillé:

Diagramme de séquence à compléter:

Scénario :



C4 : Diagramme 3 à compléter :



C5 : Méthode virtuelle :

C6 : Conséquence au niveau des classes dérivées :

C7 : La structure :

Base Nationale des Sujets d'Examens de l'enseignement professionnel
Réseau SCEREN

C8 :

Visibilité du constructeur :

Listing du QCM :

```
1  CCapteurVent * CCapteurVent :: Init_capteur (int port)
2  {      CCapteurVent *capteur = NULL ;
3          int erreur = -1 ;
4          float  valeur_capteur;
5          capteur_dialogue = new CRS232(1200,8,port) ; // écrit dans la doc du capteur
6          // vérification de la présence du capteur physique
7          if (capteur_dialogue!= NULL) // le port COM est ouvert
8              {
9                  capteur_dialogue ->Envoyer(message_init_capteur) ;
10                 erreur = capteur_dialogue ->Recevoir(&valeur_capteur) ;
11                 if (erreur !=-1) // le capteur physique existe
12                     capteur = new CCapteurVent() ;
13                 else
14                     delete capteur_dialogue ;
15             }
16         return capteur ;
17     }
```

Le QCM : Cochez la bonne réponse :

Ligne	Propositions :	VRAI	FAUX
2	Initialisation du pointeur capteur		
5	Etablissement de l'association		
12	Après l'exécution de cette ligne, capteur n'est jamais NULL		
14	Cette destruction est inutile, elle est réalisée automatiquement à la fin de la méthode		
17	capteur peut avoir pour valeur : NULL		
	Un objet de type CCapteurVent est toujours créé.		

Tableau 3

D. ORGANISATION DES DONNÉES

D1 :

Nom de la racine :

Nombre de nœuds "fils" :

Base Nationale des Sujets d'Examens de l'enseignement professionnel
Réseau SCEREN

E. TEMPS REEL

E1 :

Architecture du processeur MPC8245 :

Caractéristiques principales :

E2 : Système Préemptif :

Base Nationale des Sujets d'Examens de l'enseignement professionnel
Réseau SCEREN

E3 : Architecture client - serveur :

Justification du rôle de serveur : -----

E4 : Nombre de tâches et nom :

Nombre de tâches : -----

Justifications : -----

Nom de la tâche	Signification

Tableau 5

E5 : Déclaration et création des tâches :

```
#include "Projet.h"
```

```
// zone de déclaration
```

```
static void prvSetupHardware( void );
```

```
-----  
-----  
-----  
-----  
-----
```

```
// programme principal
```

```
int main( void )
```

```
{
```

```
  prvSetupHardware();
```

```
-----  
-----  
-----  
-----  
-----
```

```
  return 0;
```

```
}
```

E6 : Fonctionnement du sémaphore :

Étape de gestion du sémaphore :

Principe de fonctionnement :

E7 : Le QCM :

Cochez la bonne réponse :

Propositions :	VRAI	FAUX
Le sémaphore utilisé est un sémaphore binaire.		
La tâche vVisualisationTask n'est pas lancée par la supervision.		
Les tâches vEcritParamTask et vVisualisationTask accèdent simultanément à la mémoire partagée.		
L'utilisation du sémaphore dans la tâche vLEDTask (Listing 2, annexe 4) sera similaire pour la tâche vEcritParamTask .		
La tâche vEcritParamTask a pris le sémaphore et empêche la supervision d'accéder à la mémoire partagée.		
L'utilisation d'un sémaphore tel que xSemaphore (listing 2) sera bloquante pour la tâche vEcritParamTask .		

Tableau 6

F Les communications

F1 :

(Oui ou non)	RS232	RS422	RS485	Justifications
Réseau ModBus Nacelle				
Réseau ModBus Tour				

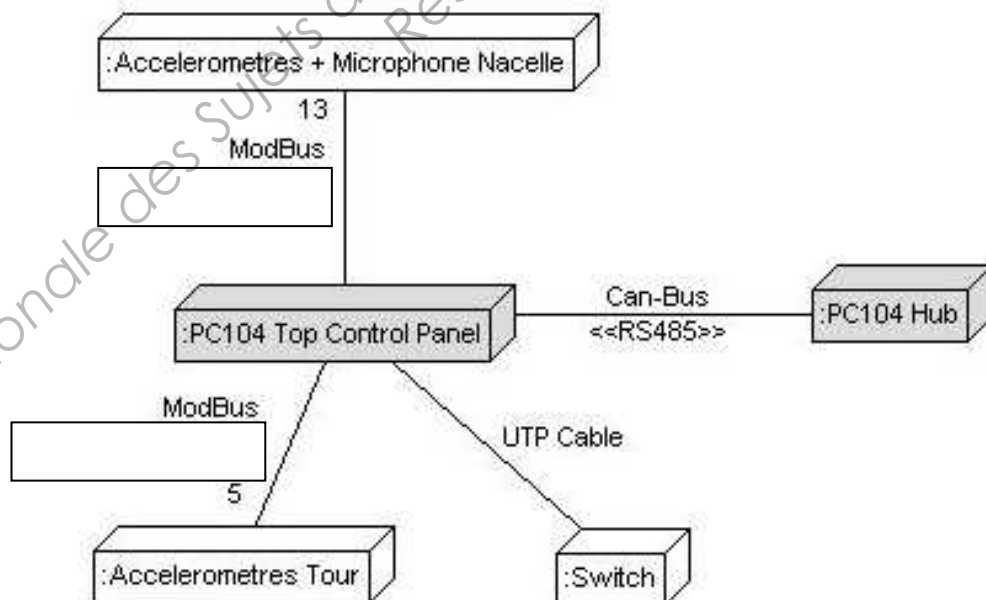
Tableau 7

F2 : Choix carte de communication pour les réseaux ModBus :

Carte : _____

Justification :

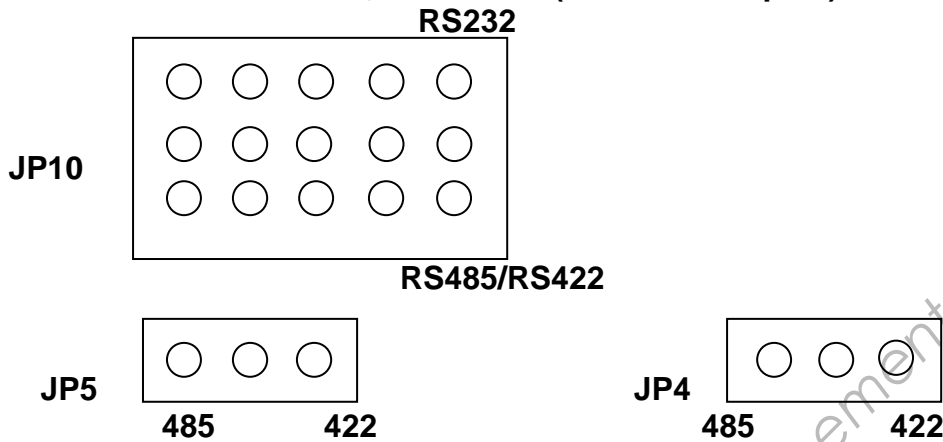
F3 : Diagramme de déploiement à compléter (les 2 rectangles vides) :



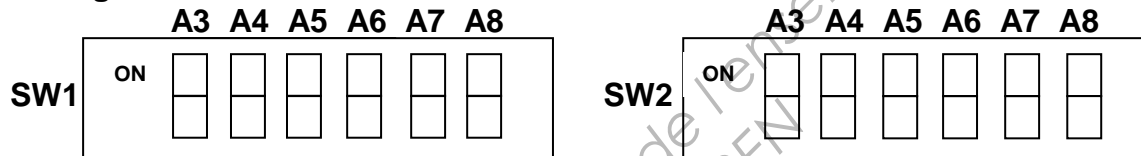
F4 : Configuration de la carte PCM-3610

Valeur du diviseur : -----

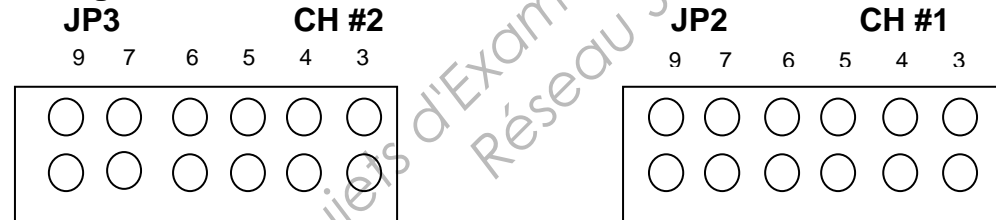
Position des cavaliers JP10, JP4 et JP5 (Entourez les pins) :



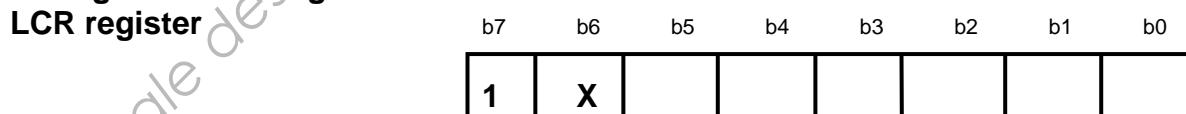
Configuration des switches :



Configuration des IRQ :



Configuration du register LCR :



F5 : Justification du role du maître "PC104 Top Control Panel" :

F6 : Résistance de terminaison :

Valeur: -----

Rôle : -----

Positionnement des cavaliers JP5 et JP6 :

F7 : Détermination du "bit rate" :

(A compléter)

BTR0 Register :

b7	b6	BPR5	BPR4	BPR3	BPR2	BPR1	BPR0
X	X						

BTR1 Register :

b7	TSEG2.2	TSEG2.1	TSEG2.0	TSEG1.3	TSEG1.2	TSEG1.1	TSEG1.0
X							

Calcul de t_{scl} :

Calcul du "Nominal bit time" :

$t_{SYNCSEG} =$

$t_{SEG1} =$

$t_{SEG2} =$

Nominal bit time =

Calcul du bit rate :

F8 : Comparatif réseau à compléter :

	Couche(s) du modèle OSI concerné(s)	Technique d'accès au réseau	Réseau déterministe ? (oui ou non)	IPv6 supporté (oui ou non)
ARCnet	Physique Liaison	Token-Bus		
Ethernet				

Tableau 8

F9 : Réseau déterministe :

G PARTIE RESEAU :

G1 : Exprimez sous **forme décimale** le masque de sous réseau.

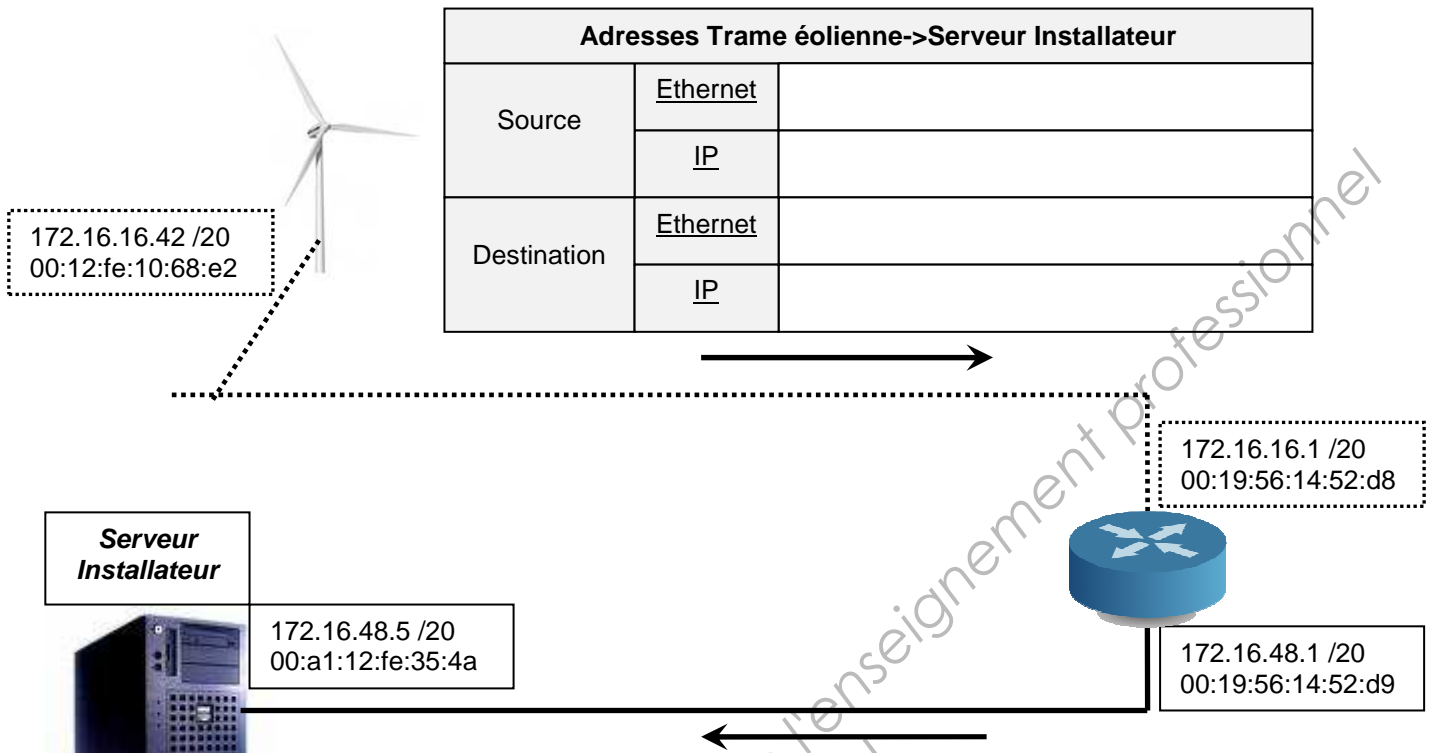
G2 : Avec un masque de **20 bits**, combien de sous réseaux valident peut-on avoir au maximum ? (détailler votre calcul)

G3 : Avec un masque de **20 bits**, combien de machines maximum peuvent avoir une adresse valide ? (détaillez votre calcul)

G4 : Pour le réseau **172.16.16.0 /20** indiquez:
l'adresse de la **première machine possible**
l'adresse de la **dernière machine possible**
l'adresse de l'adresse de **diffusion du sous réseau**.

Première Adresse :
Dernière Adresse :
Adresse de diffusion :

G5 : Trame éolienne → Serveur Installateur



Adresses Trame éolienne->Serveur Installateur		
Source	<u>E</u> thernet	
	<u>I</u> P	
Destination	<u>E</u> thernet	
	<u>I</u> P	

Adresses Trame éolienne->Serveur Installateur		
Source	<u>E</u> thernet	
	<u>I</u> P	
Destination	<u>E</u> thernet	
	<u>I</u> P	

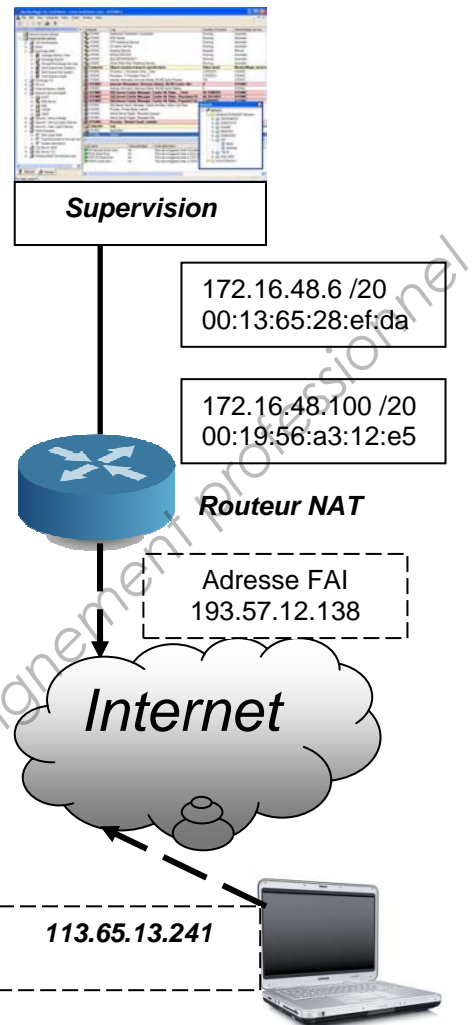
G6 : Routeur NAT : Principe :

Nat dynamique :

G7 : Trame Supervision → 113.65.13.241

Adresses Trame Supervision → 113.65.13.241		
Source	Ethernet	
	IP	
Destination	Ethernet	
	IP	

Adresses Trame Supervision → 113.65.13.241		
Source	IP	
Destination	IP	



G8 : Table de translation du routeur NAT

Table de translation du Routeur Nat		
Serveur Installateur	Port Public	
	Protocole (couche transport)	
	Port privé	
	Ip Privée	
Supervision	Port Public	
	Protocole (couche transport)	
	Port privé	
	Ip Privée	

Tableau 9