

**EDE NUM 1**

**SESSION 2023**

# Section : SCIENCES INDUSTRIELLES DE L’INGÉNIEUR Option : INGÉNIERIE INFORMATIQUE

CAPET

CONCOURS EXTERNE ET CAFEP CORRESPONDANT ET TROISIEME CONCOURS

**ÉPREUVE ÉCRITE DISCIPLINAIRE**

Durée : 5 heures

*Calculatrice autorisée selon les modalités de la circulaire du 17 juin 2021 publiée au BOEN du 29 juillet 2021.*

*L’usage de tout ouvrage de référence, de tout dictionnaire et de tout autre matériel électronique est rigoureusement interdit.*

*Il appartient au candidat de vérifier qu’il a reçu un sujet complet et correspondant à l’épreuve à laquelle il se présente.*

*Si vous repérez ce qui vous semble être une erreur d’énoncé, vous devez le signaler très lisiblement sur votre copie, en proposer la correction et poursuivre l’épreuve en conséquence. De même, si cela vous conduit à formuler une ou plusieurs hypothèses, vous devez la (ou les) mentionner explicitement.*

**NB : Conformément au principe d’anonymat, votre copie ne doit comporter aucun signe distinctif, tel que nom, signature, origine, etc. Si le travail qui vous est demandé consiste notamment en la rédaction d’un projet ou d’une note, vous devrez impérativement vous abstenir de la signer ou de l’identifier.**

**Le fait de rendre une copie blanche est éliminatoire.**

## Tournez la page S.V.P.

A

**INFORMATION AUX CANDIDATS**

Vous trouverez ci-après les codes nécessaires vous permettant de compléter les rubriques figurant en en-tête de votre copie

Ces codes doivent être reportés sur chacune des copies que vous remettrez.

* **Concours externe du CAPET de l’enseignement public** :

|  |  |  |  |
| --- | --- | --- | --- |
| Concours | Section/option | Epreuve | Matière |
| EDE | 1413E | 101 | 9311 |

* **Concours externe du CAFEP/CAPET de l’enseignement privé** :

|  |  |  |  |
| --- | --- | --- | --- |
| Concours | Section/option | Epreuve | Matière |
| EDF | 1413E | 101 | 9311 |

**►Troisième concours externe du CAPET de l’enseignement public :**

|  |  |  |  |
| --- | --- | --- | --- |
| Concours | Section/option | Epreuve | Matière |
| EDV | 1413E | 101 | 9311 |

# COMPOSITION DU SUJET

## SUJET :

* Partie 1 : Description générale Page 1
* Partie 2 : Étude du déplacement dans l’environnement Page 3
* Partie 3 : Détection d’obstacles Page 7
* Partie 4 : Étude la collecte et de l’analyse des évènements Page 12
* Partie 5 : Mise à disposition des informations utilisateurs Page 14
* Partie 6 : Conclusion Page 15

## DOCUMENTS :

* Dossier technique (DT1 à DT 17) Page 16
* Documents réponses (DR1 à DR 8)
* DR1 à DR4 Page 42
* DR5 et DR6 Page 43
* DR7 et DR8 page 44

*I M P R I M E R I E N A T I O N A L E* – 23 0223 – D’après documents fournis

EDE NUM 1

## Etude d’un moyen de déplacement interne industriel Partie 1 - Description générale

*Objectif : s’approprier le contexte de l’étude et l’intérêt de la mise en place d’une flotte de miniporteurs.*

## Conditions de travail liées à une activité de stockage

Avec le développement des activités de vente en ligne, les besoins en stockage ont considérablement évolué (plus de 1200 plateformes en France couvrent plus de 20000 m2). Une entreprise de logistique vient d’agrandir ses entrepôts en passant d’une surface de 10000 m2 à 15000 m2.

Les employés de l’entreprise rencontrent un certain nombre de difficultés dans leurs conditions de travail : temps de déplacements entre les différents espaces pour les chefs d’équipe, pénibilité du transport des charges pour les agents de maintenance et d’inventaire, insécurité des déplacements dans des entrepôts partagés entre les piétons et les véhicules.

L’espace est conçu pour optimiser le flux des marchandises (zonage des produits).

## Entrepôt d’origine : 10000 m2 (200 x 50)

Arrivée marchandises



Zone de stockage (extension)

Préparation commandes

Zone de stockage

Départ commandes



**Extension : 5000 m2 (100 x 50)**

L’entreprise a mené une campagne interne sur les troubles TMS (les troubles musculosquelettiques représentent 95 % des maladies professionnelles dans le secteur du Transport et de la Logistique et 15 % des accidents du travail sont liés au mal de dos d’après l’Assurance Maladie). Elle souhaite accompagner son extension d’une amélioration des conditions de travail pour les métiers impactés, notamment sur la durée des déplacements et le transport de charges.

Un chef d’équipe parcourt pour ses activités environ 8 km par jour pour couvrir l’entrepôt avant l’agrandissement.

1

B

## Tournez la page S.V.P.

**Question 1 :** Avec une activité similaire, calculer le temps quotidien nécessaire à cet employé pour couvrir l’entrepôt agrandi. On prendra comme hypothèse une vitesse de marche de 4,5 km·h-1 pour un piéton.

Une caisse à outils de maintenance peut peser de 10 à 20 kg, et une servante d’inventaire peut rapidement servir à déplacer 100 kg de marchandises. Pour assister au déplacement de ces charges, l’entreprise souhaite arrêter d’utiliser les véhicules type chariot de manutention, surdimensionnés (ils sont prévus pour 1 tonne et l’essentiel de l’énergie sert à leur propre déplacement) et peu maniables pour des déplacements rapides.

Afin améliorer efficacement les conditions de travail, l’entreprise a choisi de se doter de miniporteurs.

## Miniporteur Hublex

La société Hublex conçoit des outils de co-botique d’assistance à l’homme. Le miniporteur professionnel Hublex est un outil de déplacement et de transport de petites charges.



Il permet à un employé de se déplacer sans fatigue sur de longues distances, et de transporter avec lui son matériel ou des quantités limitées de marchandises.

L’entreprise en a acquis trois, afin d’assurer une disponibilité permanente entre les différentes personnes amenées à s’en servir, dans un fonctionnement en horaire de travail en 2 x 8.

La présente étude porte sur l’intérêt de l’acquisition de ces miniporteurs dans le cadre d’une amélioration efficace des conditions de travail.

Le document DT 1 détaille le diagramme des exigences partiel du système étudié.

**Question 2 :** A partir de ce document, quel serait le gain maximal en temps pour le chef d’équipe évoqué ci-dessus ? Donnez l’exigence qui répond au besoin spécifique d’un agent d’inventaire.

**Question 3 :** Justifier l’exigence 1.3.1 au regard du besoin d’améliorer les conditions de

travail avec efficacité.

**Question 4 :** A l’aide des deux diagrammes de définition de blocs (BDD) du DT2, expliquer sommairement la différence de fonctionnement dans les deux cas exposés.

## Partie 2 - Etude du déplacement dans l’environnement

*Objectif : déplacement autonome vers la zone de charge.*

Afin de minimiser le temps et les efforts pour les utilisateurs, ceux-ci laissent le miniporteur après utilisation dans une zone de dépose identifiée par des marquages au sol (rectangle vert sur la figure 1)



**Figure 1 : zone de dépose d’un miniporteur**

La zone est reliée à une ligne guide bicolore (rouge et bleue). Un système autonome d’analyse de l’environnement et de guidage ramène alors le miniporteur dans sa zone de parking et de recharge de la batterie.

La caméra OpenMV Cam H7 embarquée sur le miniporteur permet d’acquérir les images nécessaires à l’analyse de l’environnement, à l’aide d’un logiciel en MicroPython qui implémente une version allégée de Python 3.4 et des modules spécifiques à la caméra.

## Analyse de l’environnement

Pour identifier le type de zone où il se trouve, le miniporteur utilise un système de repères visuels appelé « AprilTag ». Ce système se compose de cibles carrées à imprimer et d’un logiciel de détection sous licence libre.

**Question 5 :** Indiquer ce que permet de faire ce type de licence.

Le document DT3 AprilTag présente les principes du système de repérage et le DT4 donne les cibles utilisées par le miniporteur.

**Question 6 :** Justifier l’intérêt d’utiliser ce système et donner la signification de la cible en photo sur la figure 1

Le module cam\_analysing.py réalise entre autres la détection des tags April et des blobs (ensemble de pixels connectés par des seuils de couleur, soit ici la ligne guide bicolore). Il effectue notamment les importations suivantes :



**Question 7 :** Donner la famille AprilTag utilisée par le programme et ses caractéristiques.

Le miniporteur déposé entame son retour autonome vers la zone de charge grâce au guidage de de ligne. Le prochain tag rencontré est le tag ci-contre.

**Question 8 :** Sachant qu’un zéro est codé en noir, expliquer le principe d’encodage de l’information dans ce type de tag, et en déduire la valeur hexadécimale que le programme va utiliser pour reconnaître le tag.

Une distance de Hamming *d* entre deux mots binaires signifie qu’au minimum

*d* bits varient entre les deux mots.

Le motif ci-contre entre dans le champ de vision de la caméra.

**Question 9 :** Expliquer si ce tag sera détecté comme valide.

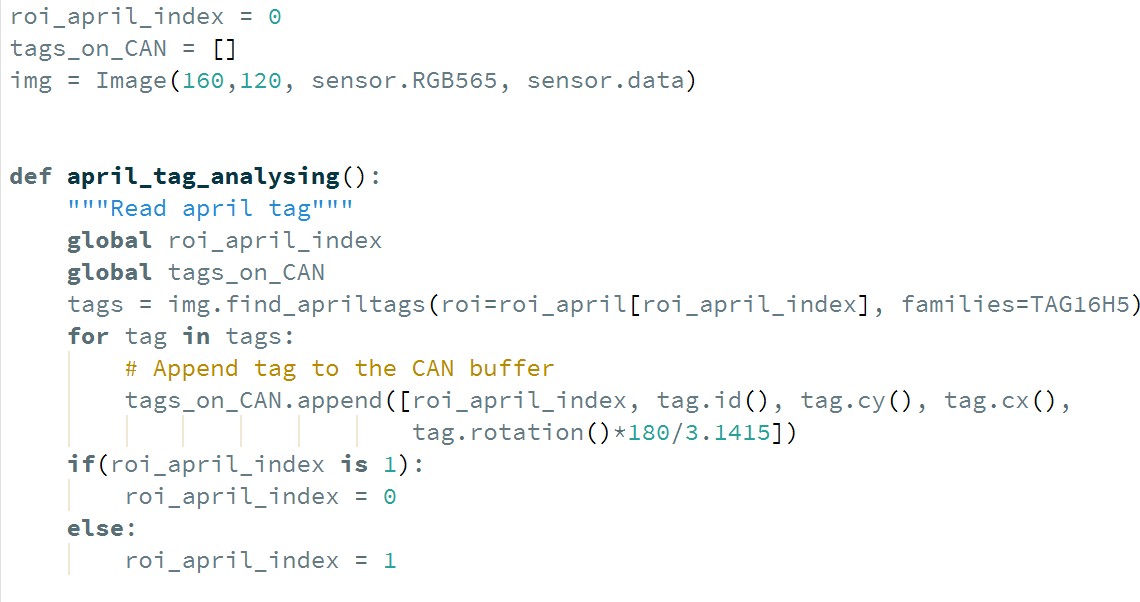
Le document DT5 présente l’organisation logicielle partielle simplifiée de l’analyse d’image.

**Question 10 :** Donner le nom du schéma UML présenté dans le document technique DT5,

ainsi que le type d’association entre les rectangles « Image » et « AprilTag ».

La méthode find\_apriltags() est utilisée pour identifier les tags qui se situent dans l’image. Elle retourne une liste d’objets de la classe apriltag, identifiés dans la zone d’intérêt de l’image (roi) passée en argument. Cette zone est identifiée par un index qui spécifie si c’est la zone gauche (roi\_april\_index vaut 0) ou droite (roi\_april\_index vaut 1) qui est analysée.

Extrait du module cam\_analysing.py :

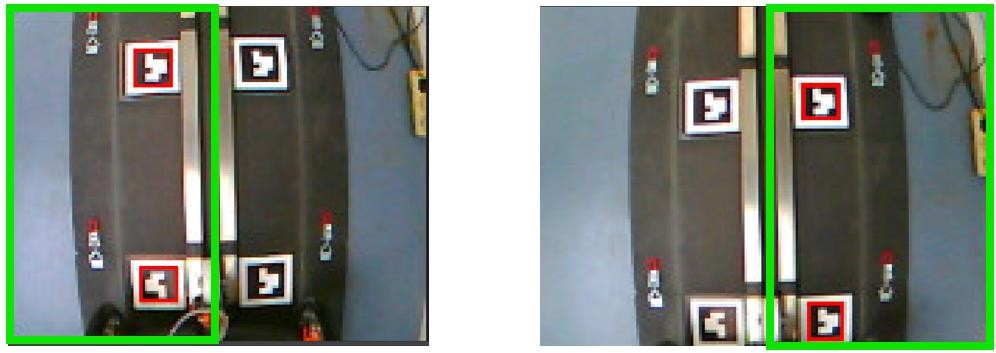


**Question 11 :** Expliquer le rôle du mot-clé « global » et indiquer pourquoi il n’est pas

appliqué à la variable img.

**Question 12 :** A l’aide du DT5, proposer une implémentation en python du constructeur de la classe Image.

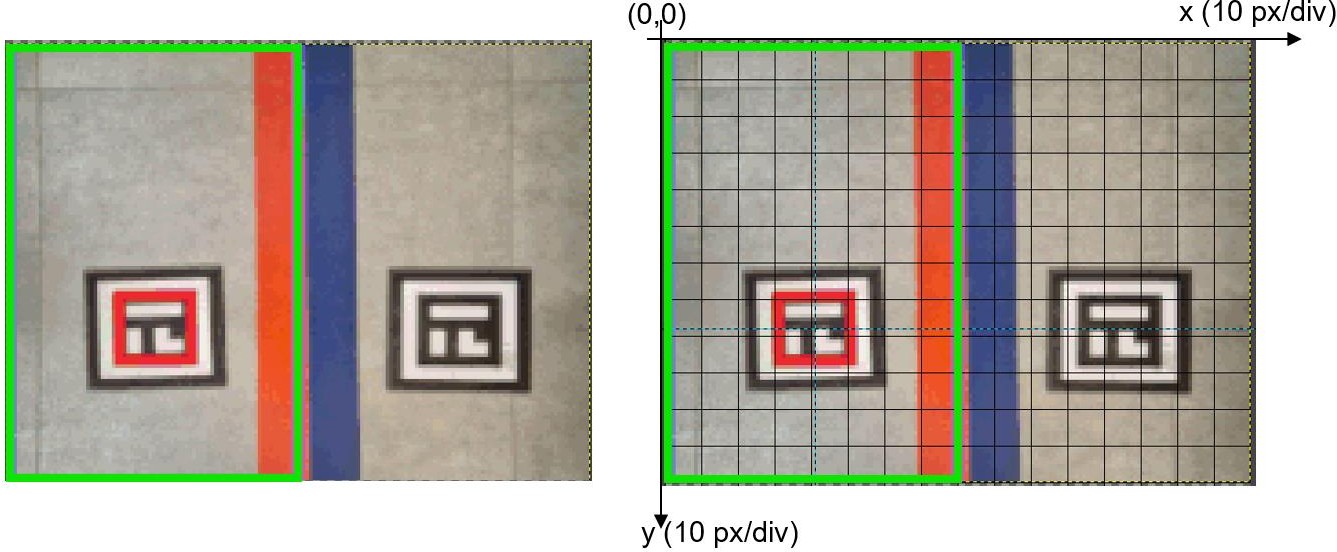
Sur chacune des vues ci-dessous, un rectangle vert indique la zone d’intérêt et un rectangle rouge indique un tag April détecté par le programme lors de l’appel à april\_tag\_analysing().



**Figure 2 : analyse A analyse B**

**Question 13 :** Déterminer ce que renvoie len(tags) après l’analyse A. Expliquer précisément ce que la variable tags contient.

La figure 3 présente une nouvelle image capturée ainsi que l’échelle de mesure associée :



**Figure 3 : image analysée et grille de mesure**

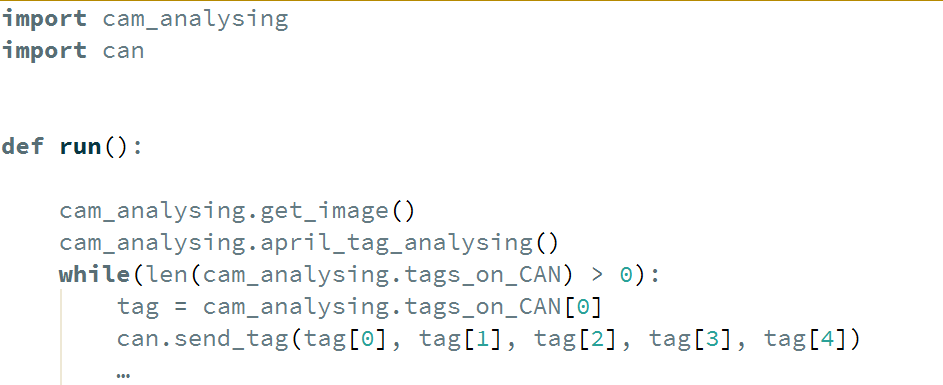
**Question 14 :** Dans le cas de la figure 3, donner le contenu de tags\_on\_CAN après l’appel

de april\_tag\_analysing().

On souhaite faire dessiner sur l’image chaque rectangle rouge autour des Tags April détectés.

**Question 15 :** Proposer une instruction pour l’implémenter et indiquer son emplacement dans le code de cam\_analysing.py

Dans le code du programme principal, on trouve les lignes suivantes :



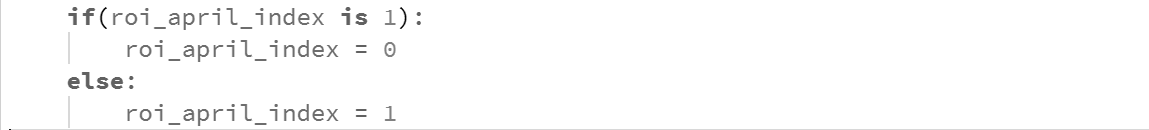
cam\_analysing.getimage() récupère une image de la caméra.

can.send\_tag() envoie un message dans le format prévu sur la liaison CAN entre la carte du guidon et la carte mère du miniporteur.

**Question 16 :** Expliquer et déterminer la condition de la boucle while. Rédiger la dernière ligne de la fonction run() sur le DR1.

La fonction run() est appelée en boucle par le programme d’acquisition du miniporteur.

On redonne les dernières lignes de april\_tag\_analysing() :



**Question 17 :** Expliquer l’objectif de ces lignes

**Question 18 :** Conclure sur la réalisation de l’exigence 1.3.1.1

## Partie 3 - Détection d’obstacles

*Objectif : étudier la solution retenue pour la gestion des obstacles lors d’un retour en autonomie du miniporteur.*

Afin de détecter de potentiels obstacles lors d’un déplacement, le miniporteur est muni de deux capteurs VL53L0X, un pointant vers l’avant, l’autre vers le bas.

## Etude du capteur

La documentation technique partielle du VL53L0X est donnée dans le DT6.

**Question 19 :** Donner la technologie employée par ces capteurs et son mode de fonctionnement.

**Question 20 :** Donner le type de signal lumineux employé. Déterminer sa longueur d’onde. A l’aide du DT7, définir la gamme de lumière dans laquelle il situe et s’il est visible par l’œil humain. Déterminer sa dangerosité pour l’œil humain.

**Question 21 :** Quels sont les trois types de mémoires présents sur le capteur ? Donner la signification des acronymes. Dans quelle(s) mémoire(s) pourrait être stockée l’adresse du composant ? Pourquoi ?

## Mise en œuvre des capteurs

La transmission des données provenant des capteurs vers la carte caméra s’effectue par le

protocole I²C.

L’adresse usine d’un capteur est 0x52.

**Question 22 :** L’adresse usine correspond à une écriture. Déterminer quelle serait alors

l’adresse en lecture. Justifier la réponse.

La vitesse de transmission sur le bus I²C utilisé est définie ainsi : i2c.init(I2C.MASTER, baudrate=400000)

**Question 23 :** Calculer dans ce cas la durée de transmission d’un bit. Calculer le temps de transmission d’un octet.

La gestion des capteurs s’effectue à l’aide d’un programme développé sous Python nommé

« obstacle.py ».

Son header est donné ci-dessous :

1...

2...

3...

4...

5...

6...

**Question 24 :** Commenter les lignes n°1, 3 et 4. Expliquer l’intérêt de préciser les classes

au lieu de faire import \*.

Les capteurs sont connectés au microcontrôleur par l’intermédiaire de connecteurs suivant le schéma du DT8. Le capteur bas est relié au connecteur P5 et le capteur avant au connecteur P4.

Leur utilisation nécessite de préciser la connexion des broches XSHUT et leur adresse de départ afin d’éviter tout problème d’adressage dans le cas de la réutilisation de capteurs d’un autre miniporteur.

Le programme de gestion des capteurs utilise entre autres la classe VL53L0X donnée partiellement en DT9.

**Question 25 :** D’après la définition de la classe VL53L0X, définir l’adresse d’initialisation

par défaut des capteurs si celle-ci n’est pas précisée spécifiquement.

**Question 26 :** Sur le document DR2, compléter les informations manquantes :

* afin d’expliciter les broches utilisées pour le signal XSHUT de chaque capteur ;
* afin d’initialiser les deux capteurs à une même adresse connue, l’adresse dite

« usine ».

Le miniporteur utilisant deux capteurs VL53L0X, il est nécessaire d’attribuer une adresse différente à chacun d’eux.

Pour ce faire, il faut désactiver un des capteurs afin d’initialiser l’autre. Cette désactivation s’effectue à l’aide de la broche XSHUT qui permet d’éteindre complètement le VL53L0X. Le capteur bas est le premier à être initialisé selon la procédure suivante :



**Question 27 :** A l’aide du DT10, expliciter chacune des 3 lignes précédentes.

**Question 28 :** Donner la commande permettant de désactiver le capteur avant.

On souhaite modifier l’adresse du capteur bas en 0x30. Le registre (index) d’adressage du capteur est le suivant : I2C\_SLAVE\_DEVICE\_ADDRESS = 0x8A

**Question 29 :** A l’aide du DT9 et des initialisations présentes sur le DR2, donner la commande permettant cette modification. Décrire les différents éléments de cette commande.

**Question 30 :** A l’aide du DT6, déterminer les commandes en hexadécimales à envoyer sur la liaison I²C du capteur bas.

**Question 31 :** Compléter le document DR3 en nommant les deux signaux et en

représentant la trame de modification d’adresse du capteur bas.

## Transmission des informations

Les informations récupérées par le microcontrôleur sont ensuite transmises par bus CAN à une carte intermédiaire gérant notamment le servomoteur pour la direction, puis en CANOpen jusqu’à la carte mère gérant les moteurs du miniporteur.

**Question 32 :** Le bus CAN comporte deux signaux, CANH et CANL, dont la lecture s’effectue en mode différentiel et la transmission par paire torsadée. Expliquer l’intérêt de ces technologies.

**Question 33 :** A l’aide du DT11, donner la taille d’un identificateur en mode standard ainsi

que le nombre maximum de valeurs d’identificateurs utilisables.

**Question 34 :** Déterminer la taille maximum que peut prendre le champ de données.

Différents nœuds peuvent accéder simultanément au bus, il y a donc nécessité d’un

arbitrage.

L’arbitrage consiste à déterminer la priorité d’une trame par rapport à une autre. Le champ pendant lequel s’effectue l’arbitrage est constitué des bits de l’identifiant ainsi que du bit RTR. Pendant le champ d’arbitrage, l’interface CAN compare les bits transmis et reçus. Le nœud qui émet un bit dominant (‘0’) gagne l’arbitrage face à un bit récessif (‘1’) et continue à émettre alors que l’autre nœud s’arrête.

**Question 35 :** Etant donné le fonctionnement du bit RTR, quelle trame sera prioritaire entre une trame de requête et une trame de données ?

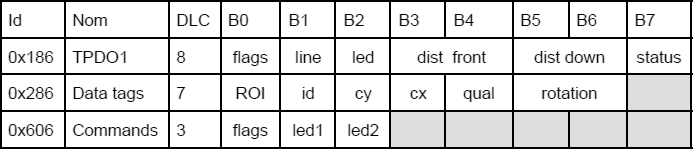
Les trois identifiants suivants sont initialisés pour la communication par bus CAN des informations provenant de la caméra :

#IDs

\_COMMANDS\_CANID = const(0x606)

\_DATA1\_CANID = const(0x186)

\_TAG\_CANID = const(0x286)

Le tableau ci-dessous présente les données transférées.

**Question 36 :** Ces 3 nœuds émettent une trame de données au même moment. Compléter le DR4 en traçant la trame résultante finalement présente sur le bus. Quelle trame est prioritaire ? Cela semble-t-il cohérent ?

**Question 37 :** Commenter la valeur du champ d’arbitrage de la trame prioritaire par rapport

aux deux autres. En déduire comment les transmissions peuvent être priorisées.

Le bus CAN étant une liaison asynchrone, il faut qu’un front soit régulièrement présent pour

resynchroniser le récepteur.

L’ajout systématique d’un bit de niveau logique inverse au bout de 5 bits de même niveau

permet de résoudre le problème.

L’émetteur ajoute donc ces bits dits de « stuffing », le récepteur ne les considèrera pas

comme de l’information utile.

**Question 38 :** Expliciter le terme asynchrone.

**Question 39 :** Le récepteur se synchronisant sur les fronts descendants, déterminer le nombre maximum de bits entre chaque resynchronisation.

Les contrôleurs moteur Hublex disposent d’une interface CAN implémentant le protocole CANOpen présenté succinctement dans le DT12.

Pour une commande moteur, le CANOpen id est 0x2000.

Cette commande change de signification selon le mode de contrôle actif au moment de la réception.

Les éléments de la trame sont les suivants :

***Sous-index*** *: Non applicable Argument selon le mode actif :*

* *Mode= Boucle Ouverte : commande en tension +/-100% Ubus, soit [-32768 ; +32767] sur 16 bits signés ;*
* *Mode = Vitesse : consigne vitesse en rpm [-200 ; +200] sur 16 bits signés ;*
* *Mode = Position : consigne position encodeur sur 32 bits signés ;*
* *Mode = Courant : consigne courant (couple) +/-13N.m ⇔ [-23920 ; +23920] sur 16 bits signés ;*

***Réponse*** *: Aucune*

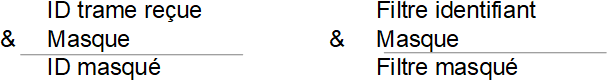
Lorsque qu’un des capteurs VL53L0X envoie une valeur indiquant une distance inférieure à 50 cm, une commande de freinage est envoyée. Elle correspond à une commande

« reverse » pendant 150 ms puis rien pendant 100 ms.

Le mode « reverse » est l’envoi d’une consigne en vitesse de – 100 rpm sur un message SDO de commande.

**Question 40 :** Compléter la trame du DR5 pour une telle consigne sur le nœud 6.

Chaque message possède un identificateur qui n'indique pas la destination du message mais la signification des données du message. Ainsi tous les nœuds reçoivent le message, et chacun est capable de savoir, grâce à un système de filtrage et de masquage, si ce dernier lui est destiné ou non.

ET

logique

Lors de la réception d’une trame, si l’ID (header) masqué est identique au filtre masqué, le

contrôleur accepte la trame.

Exemple : Filtre identifiant : 0x2CA = 010 1100 1010(2)

& Masque : 0x7CF = 111 1100 1111(2)

Filtre masqué : 0x2CA = 010 1100 1010(2)

Les trames acceptées seront 0x2CA, 0x2DA, 0x2EA et 0x2FA.

Dans le cas du miniporteur, le filtre masqué pour le nœud 6 est 0x006.

Les 3 trames rattachées au nœud 6 sont celles présentées précédemment : 0x186, 0x286, 0x606.

**Question 41 :** Déterminer si le masque 0x00F permet l’acceptation de toutes ces trames.

Déterminer quelle partie du masque permet de s’assurer que le nœud 6 n’accepte que les trames lui étant destinées mais pas celles destinées aux autres nœuds.

Les trames de données CAN et les trames distantes contiennent une protection basée sur un polynôme CRC : l'émetteur calcule une somme de contrôle à partir des bits transmis et fournit le résultat dans la trame dans le champ CRC. Les récepteurs utilisent le même polynôme pour calculer la somme de contrôle à partir des bits vus sur les lignes de bus. La somme de contrôle auto-calculée est comparée à la somme reçue. Si elle correspond, la trame est considérée comme correctement reçue et le nœud récepteur transmet un état dominant dans le bit du slot ACK, écrasant l'état récessif de l'émetteur. En cas de non- concordance, le nœud récepteur envoie une trame d'erreur après le délimiteur ACK.

Le document DT 13 présente la manière dont est calculé le CRC.

**Question 42 :** Recopier puis compléter l’algorithme suivant permettant ce calcul : CRC\_RG = 0; // initialize shift register

REPEAT

CRCNXT = NXTBIT EXOR CRC\_RG(14);

…………………………………………………. // shift left by

…………………………………………………. // 1 position IF CRCNXT THEN

…………………………………………………..

ENDIF

UNTIL (CRC SEQUENCE starts or there is an ERROR condition)

**Question 43 :** Conclure sur la réalisation de l’exigence 1.3.1.2 concernant la détection et

l’évitement des obstacles.

## Partie 4 - Etude de la collecte et de l’analyse des évènements

*Objectif : Traitement des évènements en prévention des pannes et accidents.*

## Analyse de la solution existante

Les miniporteurs enregistrent lors de leur fonctionnement un certain nombre d’évènements,

qui sont utilisés à titre de supervision et de maintenance.

Ces données sont stockées dans des fichiers de type csv, comme par exemple la liste des

alarmes déclenchées sur un miniporteur pendant une période d’utilisation.

Ces fichiers sont ensuite récoltés sur chaque équipement pour une visualisation et un traitement des données via un tableur.

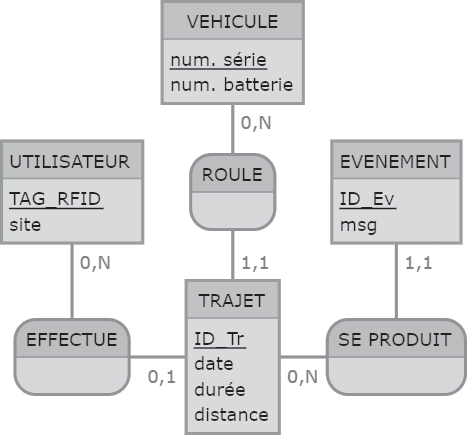
**Question 44 :** Expliquer ce que sont ces fichiers : donner leur structure et leur usage en général.

## Passage à une base de données

Le fournisseur de miniporteurs propose un service de maintenance basé sur ces données, et souhaite passer à une structure de stockage sous forme de base de données relationnelle des informations consolidées de tous les miniporteurs, et d’une gestion de cette base.

**Question 45 :** Expliquer les avantages à passer dans un modèle de type base de données relationnelle avec un système de gestion associé.

La figure 4 ci-dessous donne le modèle association-entité simplifié de la base. En particulier, un trajet est défini par une utilisation sans interruption d’un miniporteur.



?, ?

Figure 4 : modèle association-entité

**Question 46 :** Expliquer ce que signifie l’association EFFECTUE, et justifier la cardinalité (0,1) proche de l’entité trajet.

**Question 47 :** Proposer et justifier une cardinalité proche de l’entité utilisateur.

La translation du modèle conceptuel en modèle relationnel est donnée partiellement ci- dessous :

VEHICULE ( num. série : int, num. batterie : int ) UTILISATEUR ( TAG\_RFID : string, site : string ) EVENEMENT ( ID\_Ev : int, msg : string, #ID\_Tr : string)

**Question 48 :** Expliquer ce que représente ID\_Tr dans la relation Evenement.

**Question 49 :** Rédiger la dernière relation de la translation du modèle conceptuel en modèle relationnel.

La base de données est une base interrogeable en langage SQL. Pour analyser l’utilisation

des miniporteurs, on souhaite extraire un certain nombre de statistiques.

**Question 50 :** Donner l’instruction SQL pour afficher la distance moyenne des trajets sur

l’ensemble des résultats.

Le résultat que cette requête donne sur tous les enregistrements de la base de données est de 270 m.

On souhaite savoir si certains trajets n’ont pas pu être effectués en raison de problèmes

matériels.

**Question 51 :** Proposer une requête qui liste les numéros et date des trajets qui n’ont pas

abouti à un déplacement.

**Question 52 :** Donner le résultat de cette requête à partir des tables du DT14.

**Question 53 :** Proposer une requête qui liste les messages et les RFID des Miniporteurs

associés aux trajets qui n’ont pas abouti à un déplacement, en utilisant une jointure.

**Question 54 :** Expliquer les conditions nécessaires pour pouvoir supprimer un

enregistrement de la table utilisateur sans provoquer un problème d’intégrité.

**Question 55 :** Conclure sur l’intérêt de collecter les informations des trajets dans un objectif de maintenance

## Partie 5 - Mise à disposition des informations utilisateurs

*Objectif : permettre l’accès aux données et à la visualisation des évènements d’un site pour un client.*

## Etude du serveur web

La solution initiale avec des fichiers CSV est exploitée par le fournisseur Hublex au travers

d’une plateforme de gestion de site web no code, Google Sites.

Le document DT15 est une capture de l’outil Wireshark, qui est un analyseur de réseau. Il représente une sélection des trames capturées sur l’interface réseau de la machine client, lorsque la société de stockage se connecte sur le site dédié mis à sa disposition par le fournisseur avec l’url : https://sites.google.com/hublex.fr/client-societe-stockage

**Question 56 :** Expliquer ce que représentent les trames 1 et 2 et détailler le fonctionnement de ce protocole.

Les trames 3 à 11 montrent la synchronisation au niveau TCP puis l’initialisation de la session TLS et le protocole « poignée de main » entre le client et le serveur. Le client envoie sa requête pour la page web à la trame 12.

**Question 57 :** Expliquer pourquoi on ne peut pas lire en en clair le mot clé de la requête GET dans cette trame.

**Question 58 :** Etablir le modèle de réseau en couches OSI et y situer les protocoles mis en

œuvre dans le DT 15.

**Question 59 :** Compléter le diagramme de séquence du DR6 en faisant l’hypothèse que le serveur répond favorablement à la demande de page web du client.

La future évolution pourra s’intégrer dans un serveur sous la maîtrise complète du

fournisseur de maintenance du miniporteur.

Les données récoltées sont transférées via le réseau internet sur un serveur qui les stocke dans la base de données. Celle-ci est exploitée au travers d’une architecture logicielle de type LAMP (Linux, Apache, MySQL, PHP) pour permettre l’accès au site web pour le client. Il peut :

* visualiser les rapports de suivi d’utilisation (durée d’utilisation par véhicule, plages horaires d’utilisation, taux d’alarme batterie faible…) ;
* soumettre une requête au support ;
* demander un devis.

Pour assurer la confidentialité des données, le client doit s’identifier avant d’accéder à la page des rapports de suivi d’utilisation.

Le code simplifié de la page index.html qui contient un formulaire de connexion, est présenté dans le DT16.

**Question 60 :** Expliquer le rôle des trois fichiers auxquels fait appel cette page et indiquer

leur emplacement d’exécution (client ou serveur).

Le DR7 donne le contenu simplifié du fichier script.js qui s’assure de la saisie de tous les

champs par l’utilisateur.

**Question 61 :** Compléter les lignes 4, 6 et 20 du code du DR7 et justifier comment l’un des

cas du switch permet la soumission du formulaire.

Le DT17 donne le code partiel du fichier rapports.php. Ce fichier est exécuté lorsque le client s’est correctement identifié. Il utilise la classe PDO pour se connecter à la base de données, en instanciant un objet PDO.

**Question 62 :** Expliquer le rôle du bloc entre les lignes 10 et 16.

**Question 63 :** Décrire ce que fait la ligne 21 (echo"<h2> durée des trajets dans la base".DBNAME." </h2>";).

La méthode fetchAll() a été paramétrée pour retourner les résultats de la requête sous la

forme d’un tableau de tableau associatif (ou encore dictionnaire).

**Question 64 :** Compléter la boucle du DR 8 qui affiche le numéro du trajet sur une première ligne, puis la durée (en italique) et la date, sur une seconde ligne.

**Question 65 :** Conclure sur l’intérêt de pouvoir suivre les informations de type durée

d’utilisation pour la société de stockage

## Partie 6 - Conclusion

**Question 66 :** Conclure sur l’utilité et la gestion d’une flotte de miniporteurs au sein d’un

environnement de type entrepôt.

EDE NUM 1

## DOSSIER TECHNIQUE

DT 1 : Diagramme d’exigences partiel

DT 2 : BDD

DT 3 : AprilTag

DT 4 : Tags April reconnus par le miniporteur

DT 5 : Organisation logicielle partielle simplifiée de l’analyse d’image

DT 6 : Documentation partielle du capteur VL53L0X DT 7 : Longueurs d’ondes et rayonnement Laser DT 8 : Shield VL53L0X

DT 9 : Fichier partiel de la classe VL53L0X

DT 10 : Extrait de la documentation de la classe PIN DT 11 : Format d’une trame CAN

DT 12 : Documentation CANopen DT 13 : Calcul du CRC

DT 14 : Extraits de tables intermédiaires DT 15 : Capture Wireshark

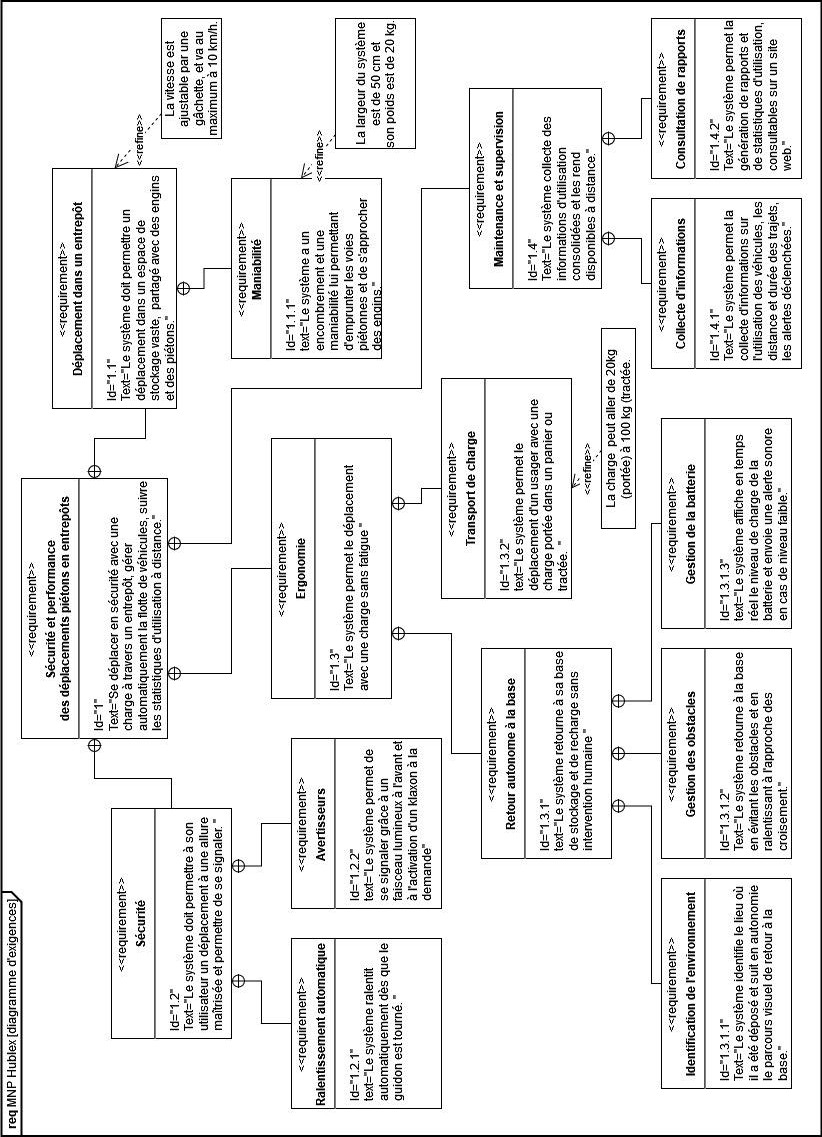
DT 16 : Fichier simplifié index.html DT 17 : Fichier partiel rapports.php

16

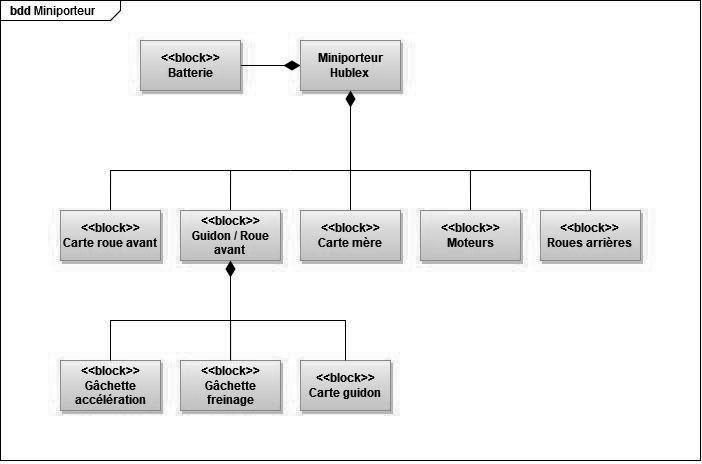
C

## Tournez la page S.V.P.

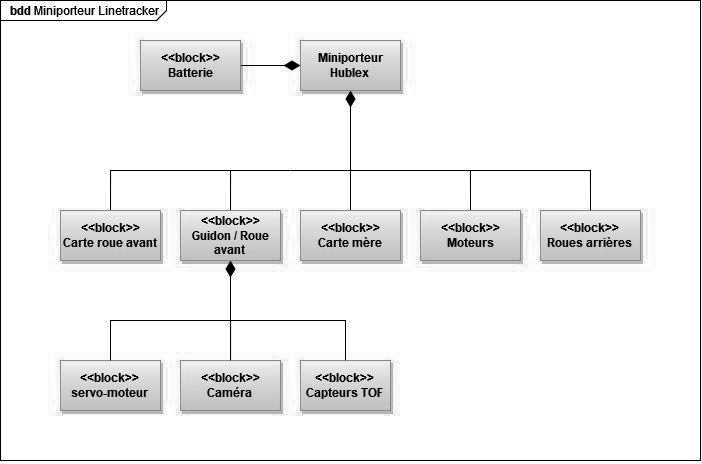
**DT 1 : Diagramme d’exigences partiel**



**DT 2 : BDD**



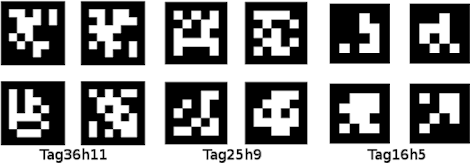
Fonctionnement avec utilisateur



Fonctionnement en autonomie

## DT 3 : AprilTag

AprilTag est un système de repères de balises visuels, utile pour une grande variété de tâches, notamment la réalité augmentée, la robotique et l’étalonnage de caméras.



## Echantillons de différentes familles d’AprilTag

Les cibles peuvent être créées à partir d’une imprimante ordinaire et le logiciel de détection AprilTag calcule la position 3D précise, l’orientation et l’identité des balises par rapport à la caméra. Il est conçu pour être facilement inclus dans d’autres applications, ainsi que pour être portable sur des appareils embarqués. Des performances en temps réel peuvent être obtenues même sur des processeurs de qualité téléphone portable.

La conception des repères et le système de codage sont basés sur un système de codage lexicographique presque optimal, et le logiciel de détection est robuste aux conditions d’éclairage et à l’angle de vue. Les cibles AprilTag sont conceptuellement similaires aux QR codes, en ce sens qu’ils sont un type de code à barres bidimensionnel. Cependant, ils sont conçus pour coder des charges utiles de données beaucoup plus petites (entre 4 et 12 bits), ce qui leur permet d’être détectés de manière plus robuste et à partir de portées plus longues. De plus, ils sont conçus pour une précision de localisation élevée : on peut calculer la position 3D précise de l’AprilTag par rapport à la caméra.

*(Licence BSD - extrait de « APRIL Robotics Laboratory » - University of Michigan)*

Famille

Une famille d’AprilTag se définit par le nombre de bits utilisés pour l’encodage et la distance de Hamming utilisée. Ainsi, TAG36h11 signifie que 36 bits (6x6) sont utilisés pour coder et que la distance de Hamming vaut 11. La zone codée est entourée d’un carré noir.

Distance de Hamming

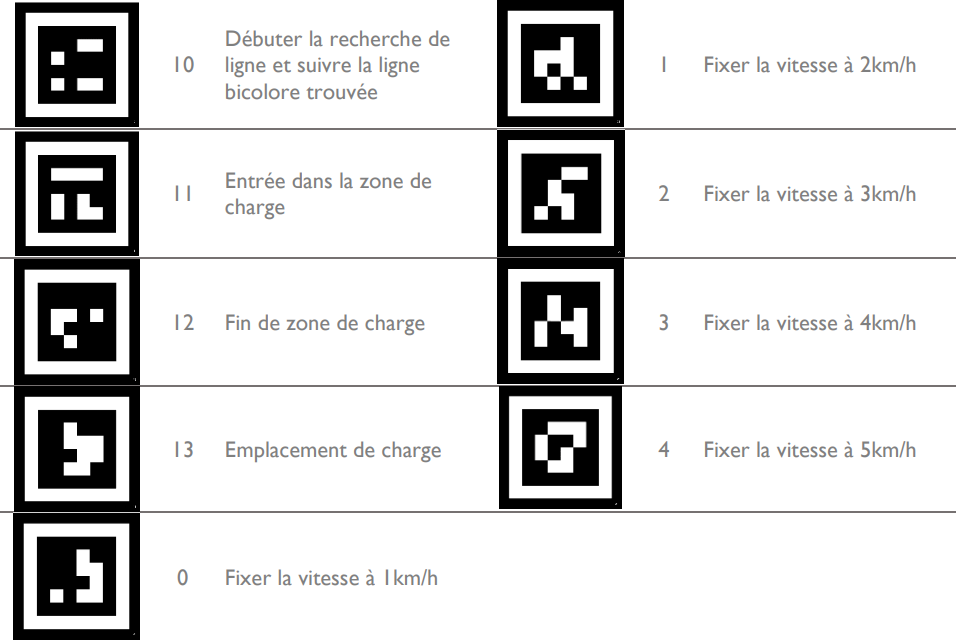
La distance de Hamming permet ici de quantifier la différence entre deux images codées, pour éviter de les confondre en cas d’erreur de codage ou de reconnaissance (une case noire devenant blanche ou vice-versa, en raison de la luminosité, de salissures…). Certaines images parmi toutes les possibilités ne sont ainsi pas disponibles dans une famille donnée.

Tableau des familles avec le nombre d’images réellement utilisables :

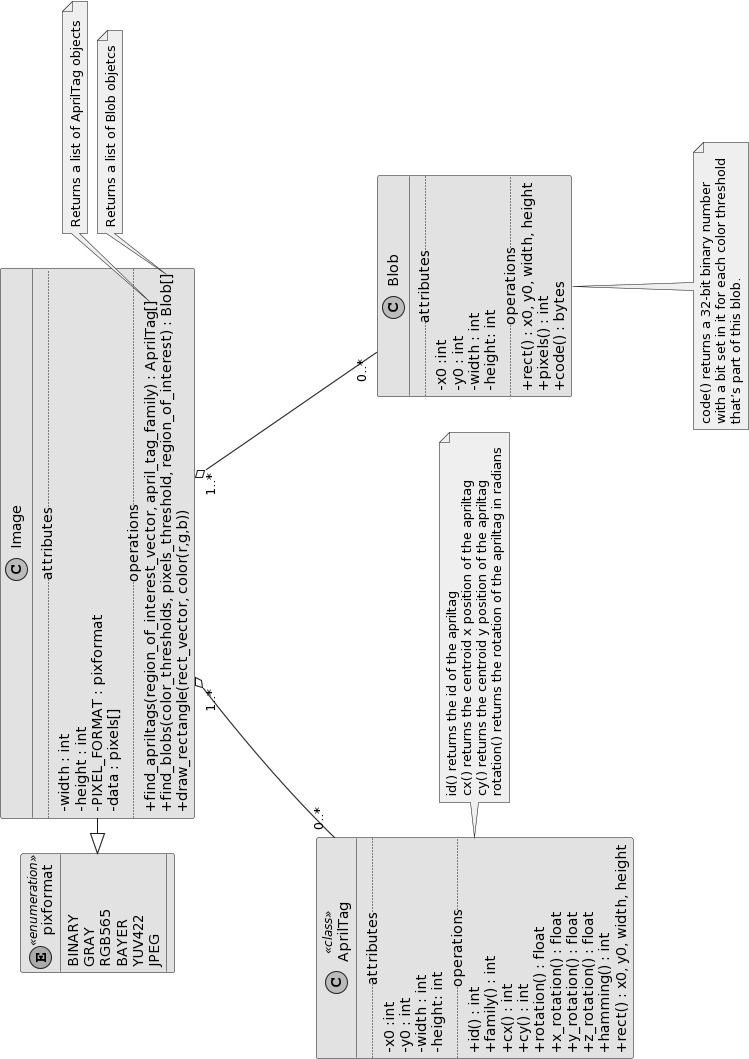
|  |  |
| --- | --- |
| Famille | Identifiants des cibles disponibles |
| TAG16H5 | De 0 de à 29 |
| TAG25H7 | De 0 de à 241 |
| TAG25H9 | De 0 à 34 |
| TAG36H10 | De 0 à 2319 |
| TAG36H11 | De 0 à 586 |

**DT 4 : Tags April reconnus par le miniporteur**

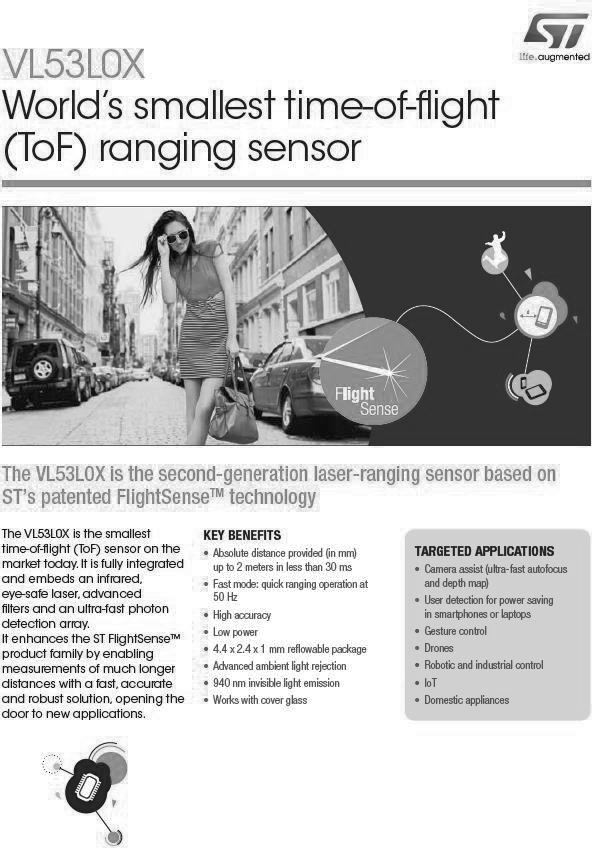
Note : deux cadres, 1 noir et 1 blanc, ont été ajoutés aux cibles de base imprimées pour le miniporteur.

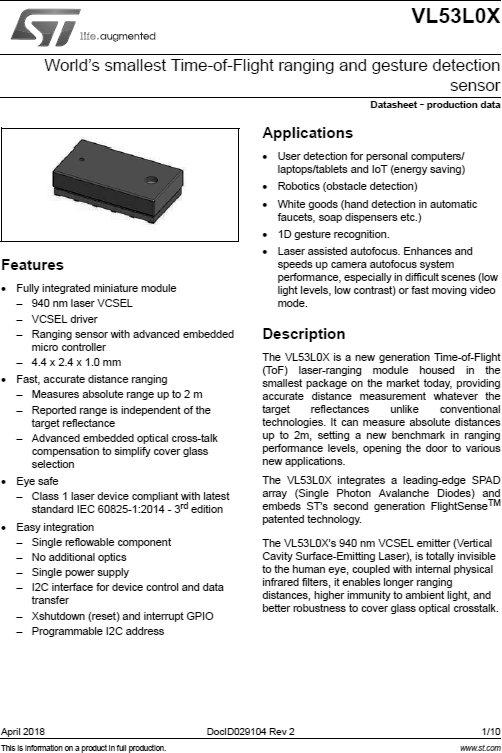


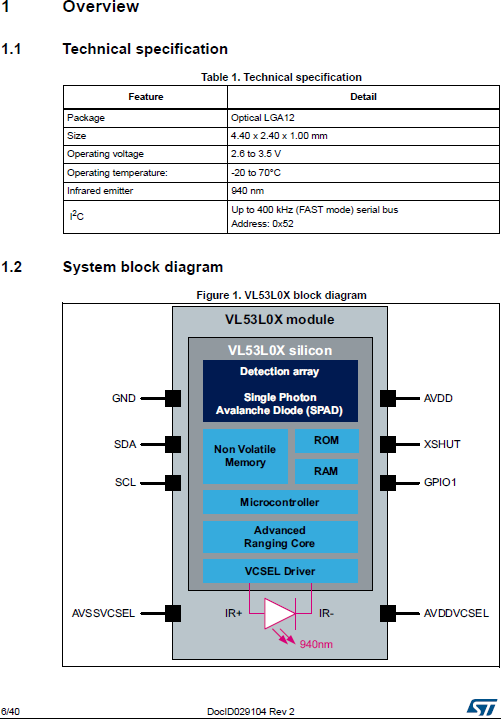
## DT 5 : Organisation logicielle partielle simplifiée de l’analyse d’image

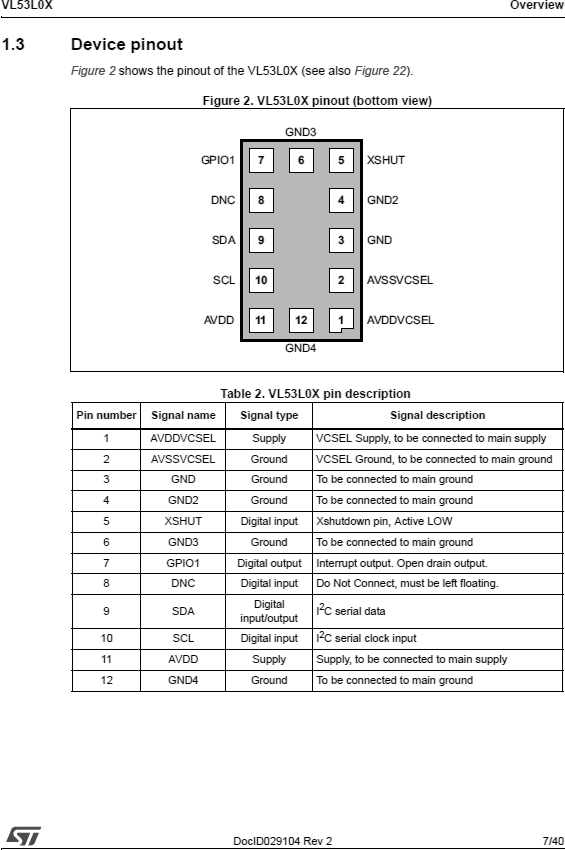


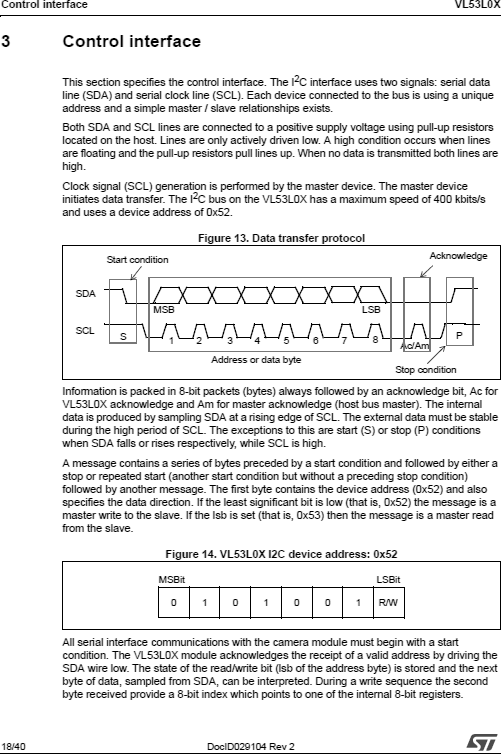
**DT 6 : Documentation partielle du capteur VL53L0X**

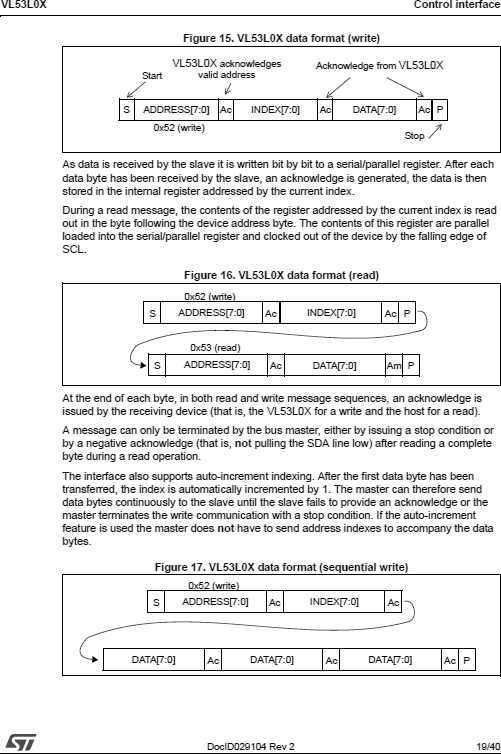






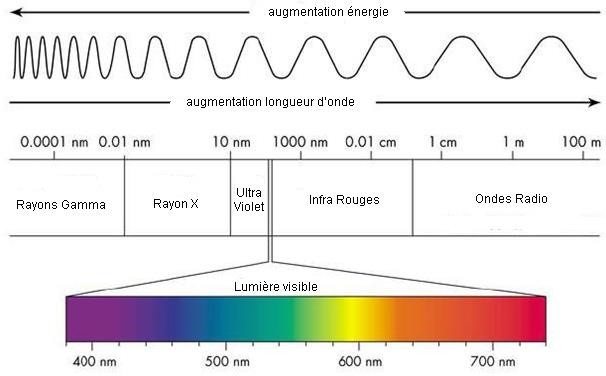




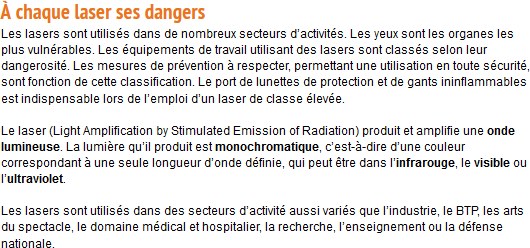


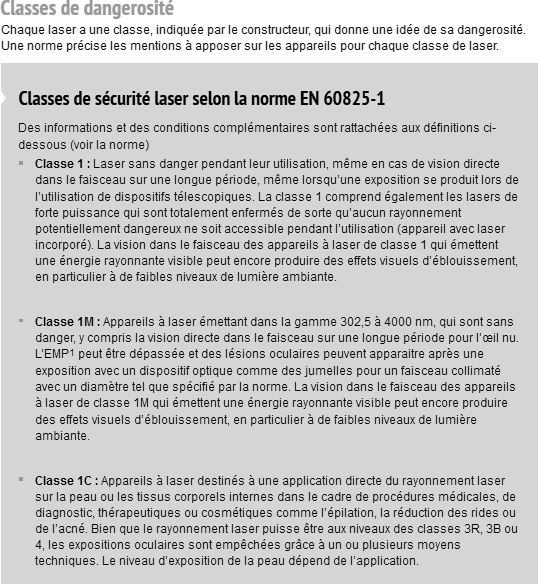
**DT 7 : Longueurs d’ondes et rayonnement Laser**

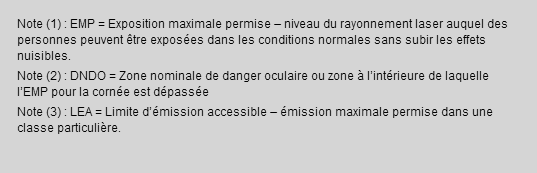
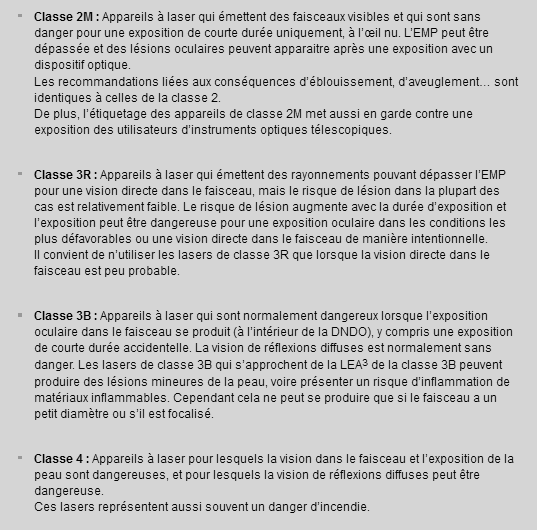
**Longueurs d’ondes**



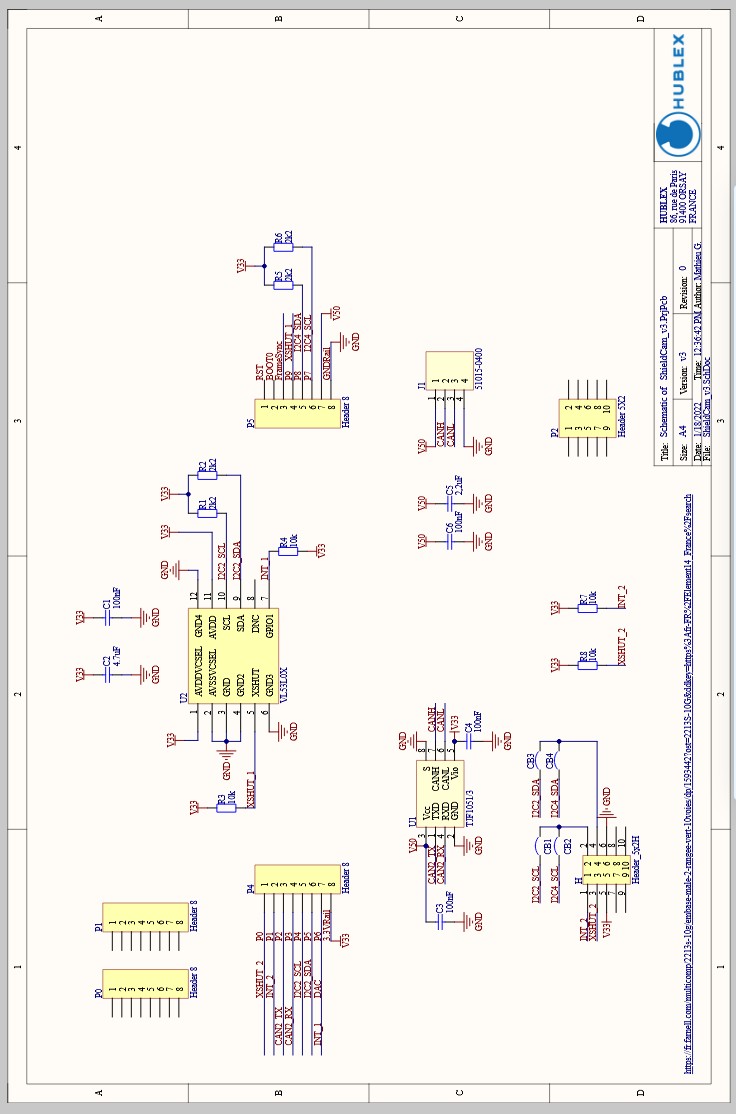
**Rayonnement laser (INRS.fr)**



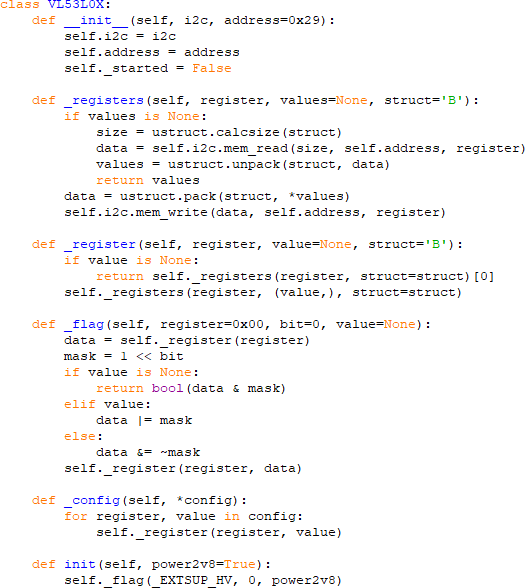




**DT 8 : Shield VL53L0X**



**DT 9 : Fichier partiel de la classe VL53L0X**

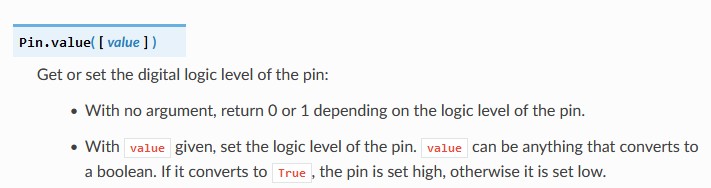




**DT 10 : Extrait de la documentation de la classe PIN**

**Partie 7 - class Pin – control I/O pins**

A pin is the basic object to control I/O pins. It has methods to set the mode of the pin (input, output, etc) and methods to get and set the digital logic level.



## DT 11 : Format d’une trame CAN

Une trame CAN est composée des champs suivants :

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Start | Arbitrage | Commande | Données | CRC | ACK | EOF |

SOF (Start Of Frame) : Début de trame. Il est réduit à sa plus simple expression : 1 bit dominant qui peut apparaître dès la fin de l'IFS. Il sert de synchronisation pour toutes les stations.

Champ d’arbitrage : Il contient 2 données qui sont l’identificateur (11 bits en mode standard et 29 en mode étendu) et le bit de RTR (Remote Transmission Request) qui est dominant (= ‘0’) pour une trame de données et récessif (= ‘1’) pour une trame de requête.

Champ de commande : Il comporte 6 bits. Les 2 premiers sont toujours dominants en mode standard. Les 4 suivants représentent le DLC (Data Length Code), c’est-à-dire le nombre d’octets présents dans le champ Data.

Champ de données (Data) : Il peut varier de 0 à 8 octets. Dans le cas d’une trame de

requête, ce champ est vide.

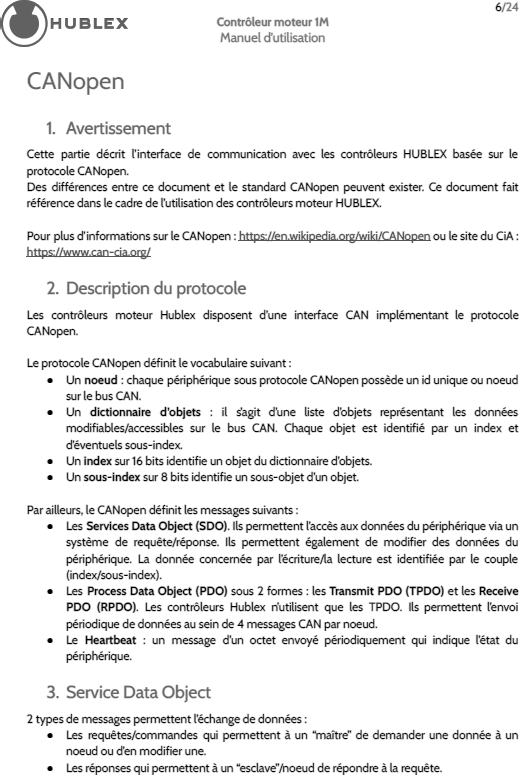
Champ de CRC (Contrôle de Redondance Cyclique) : Il est composé de 16 bits (15 + 1). La séquence calculée CRC est contenue dans les 15 premiers bits tandis que le dernier bit est un délimiteur de fin de champ de CRC (bit toujours récessif).

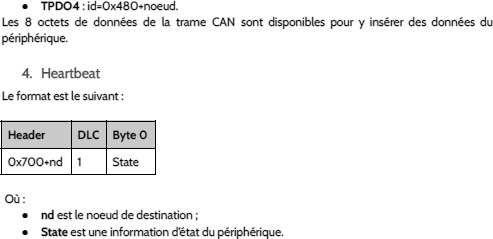
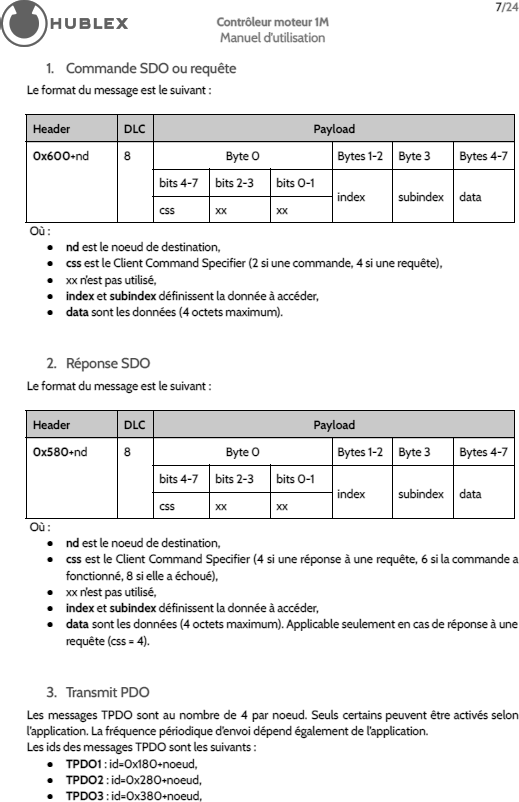
Champ d’acknowledge (acquittement) : Il est composé de 2 bits laissés libres par la station émettrice. Le premier correspond à l’acquittement par l’ensemble des nœuds ayant reçus le message ( ‘0’ signifie acquitté et donc, pas d’erreur détectée ; ‘1’ signifie non acquitté, émission d’une trame d’erreur). Le second est toujours récessif et délimite l’« acknowledge ».

EOF (End of Frame) : Ce champ de fin de trame est constitué de 7 bits récessifs.

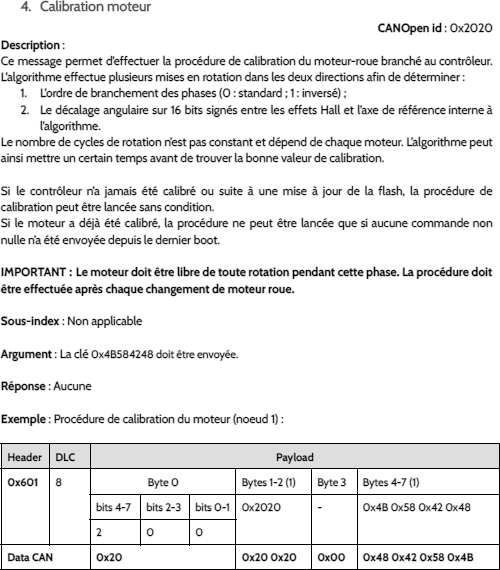
34

## DT 12 : Documentation CANopen



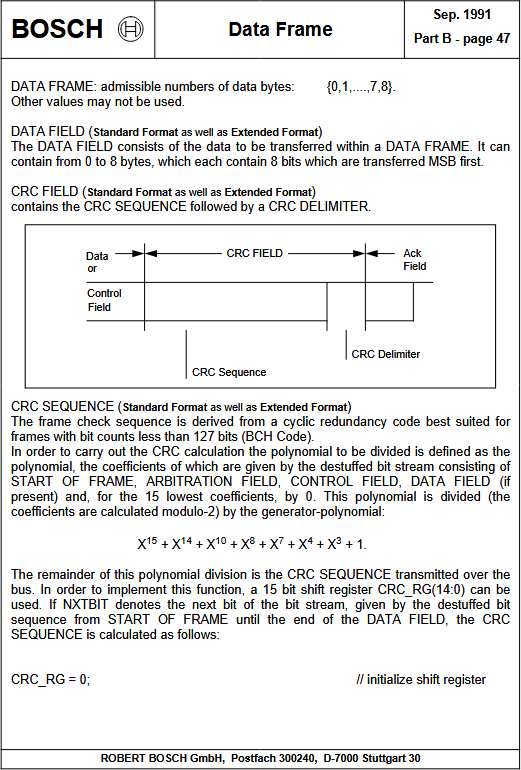


Exemple :



(1) : Octet de poids faible en première position

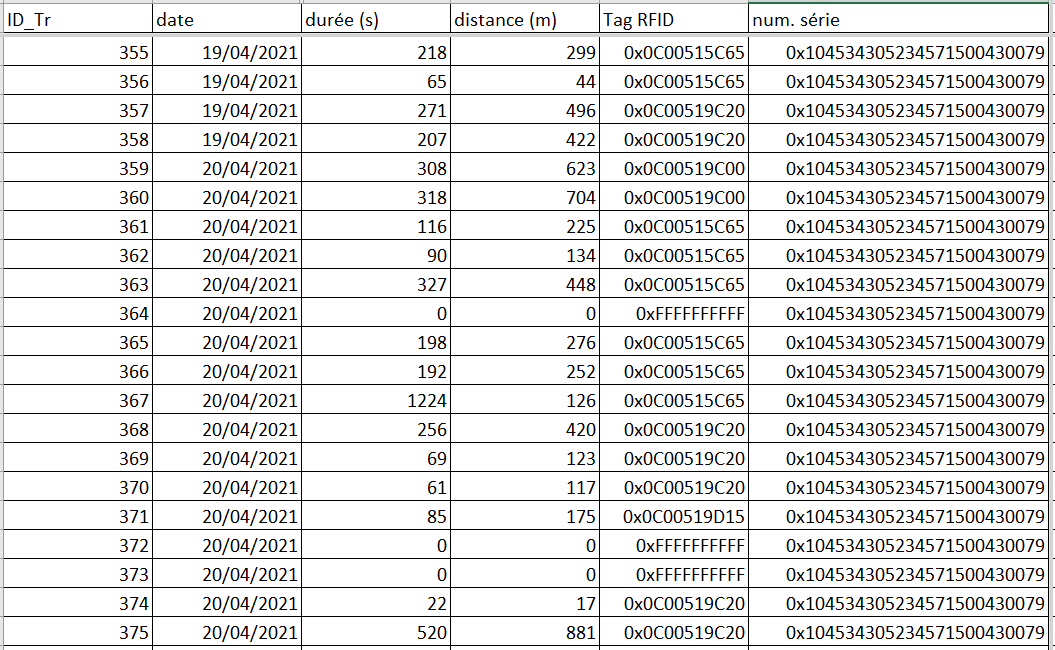
## DT 13 : Calcul du CRC



**DT 14 : Extraits de tables intermédiaires**

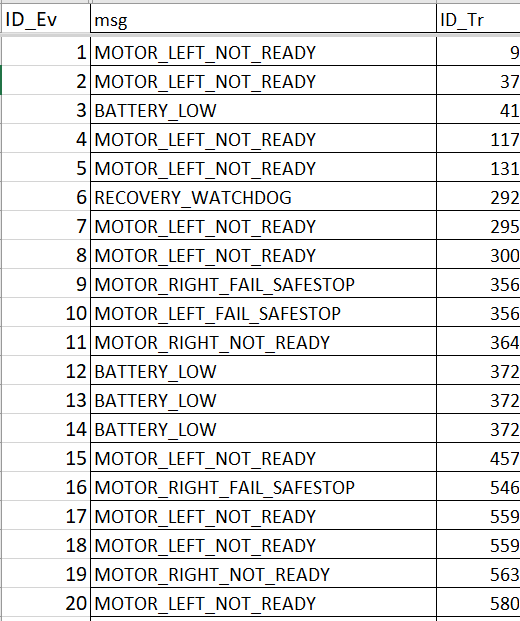
Résultat de la requête :

SELECT \* FROM trajet as T\_extrait WHERE ID\_Tr<376 AND ID\_Tr>354 T\_extrait :

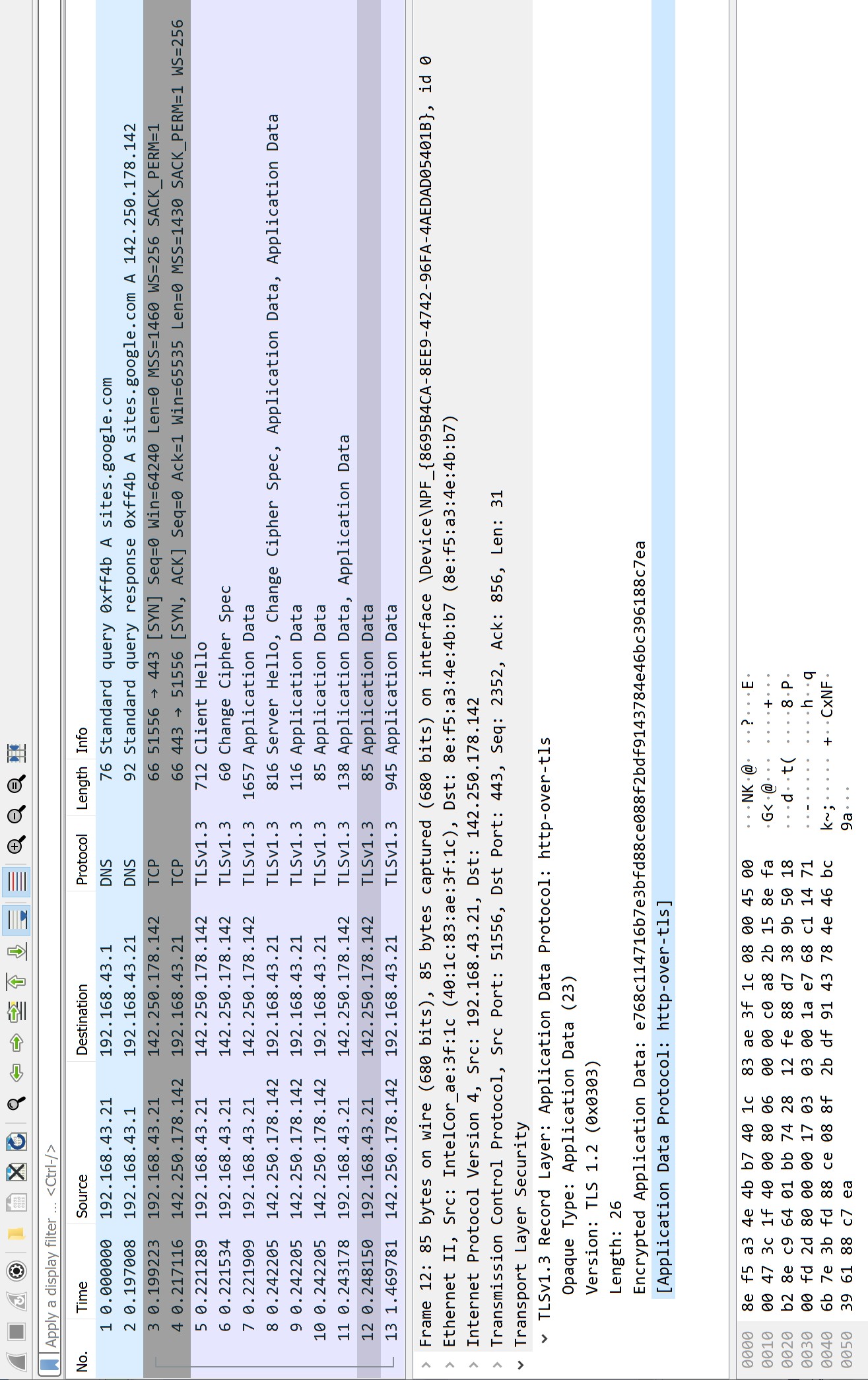


Résultat de la requête :

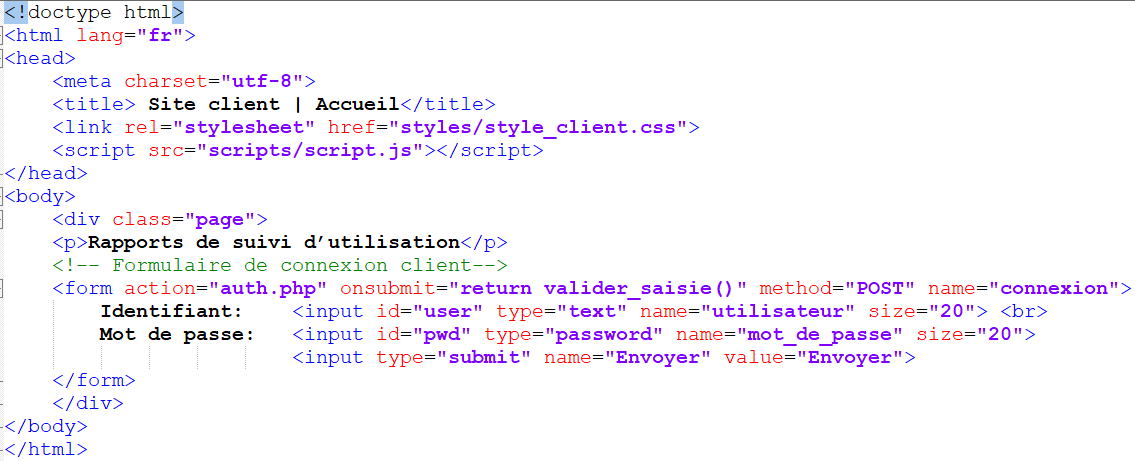
SELECT \* FROM evenement as E\_extrait WHERE ID\_Ev<21 E\_extrait :



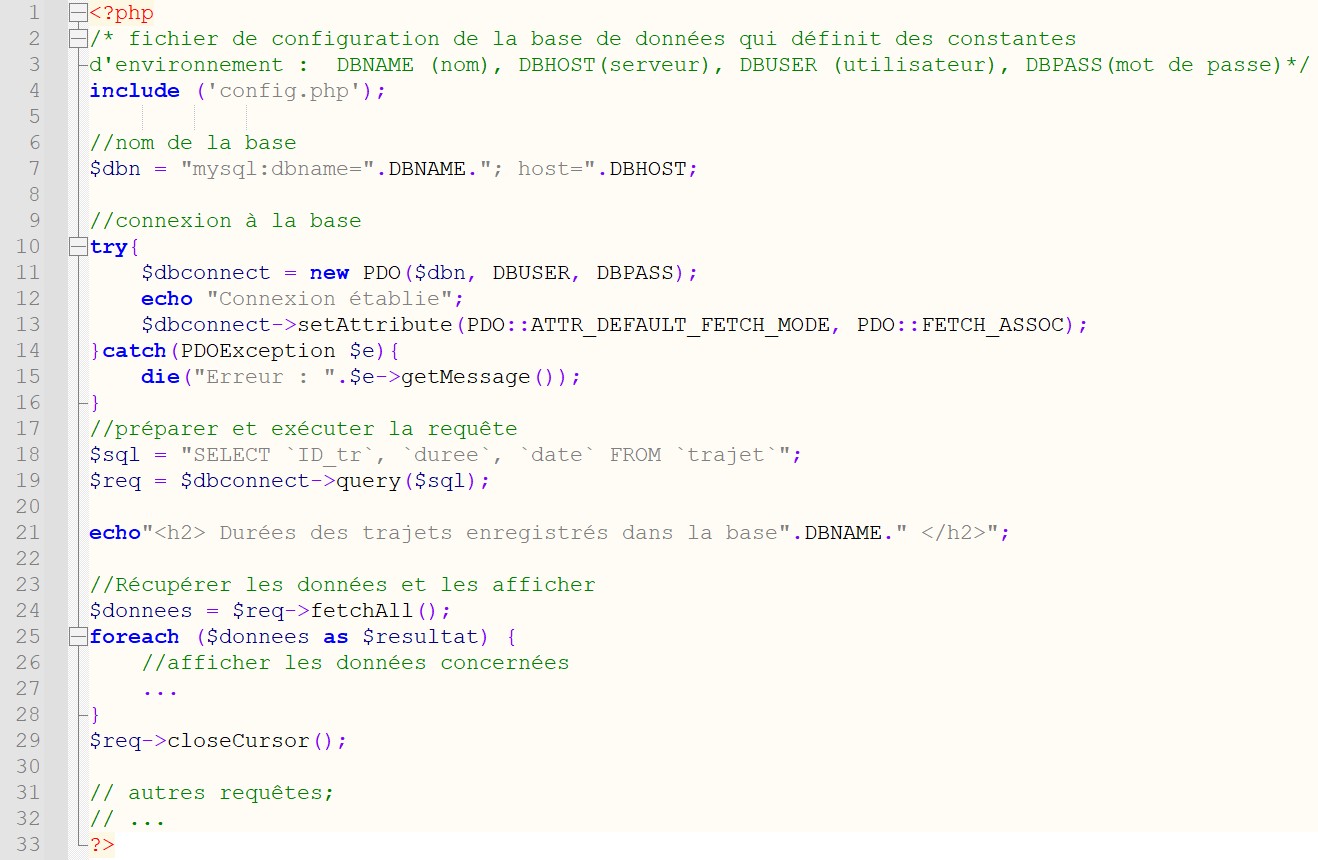
## DT 15 : Capture Wireshark



**DT 16 : Fichier simplifié index.html**



**DT 17 : Fichier partiel rapports.php**



|  |
| --- |
| **Modèle CMEN-DOC v2 ©NEOPTEC**  *(Suivi, s’il y a lieu, du nom d’usage)* $$$$$$$$$$$$$$$$$$$$$$$  **Nom de famille :**  **Prénom(s) :** $$$$$$$$$$$$$$$$$$$$$$$  **Inscription :** $$$$$$$$$$ **Né(e) le :** $$ $$ $$$$  **Numéro**  *(Le numéro est celui qui figure sur la convocation ou la feuille d’émargement)* |
| *(Remplir cette partie à l’aide de la notice)*  **Concours / Examen :** ……………………………….. **Section/Spécialité/Série :** ………………………………………………………  **Epreuve :** ……………………………….. **Matière :** ………………………….. **Session :** ……………………………….. |
| *Remplir soigneusement, sur CHAQUE feuille officielle, la zone d’identification en MAJUSCULES.*  *Ne pas signer la composition et ne pas y apporter de signe distinctif pouvant indiquer sa provenance.*  ***CONSIGNES*** *Numéroter chaque PAGE (cadre en bas à droite de la page) et placer les feuilles dans le bon sens et dans l’ordre.*  *Rédiger avec un stylo à encre foncée (bleue ou noire) et ne pas utiliser de stylo plume à encre claire. N’effectuer aucun collage ou découpage de sujets ou de feuille officielle. Ne joindre aucun brouillon.* |

EDE NUM 1

**DR1 à DR8**

**Tous les documents réponses sont à rendre, même non complétés.**

## Tournez la page S.V.P.

D



**NE RIEN ECRIRE DANS CE CADRE**

**DR 1**

while(len(cam\_analysing.tags\_on\_CAN) > 0) : tag = cam\_analysing.tags\_on\_CAN[0]

can.send\_tag(tag[0], tag[1], tag[2], tag[3], tag[4])

...................................................

**DR 2**



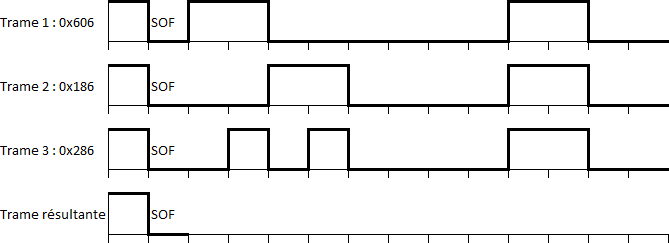
**DR 3**



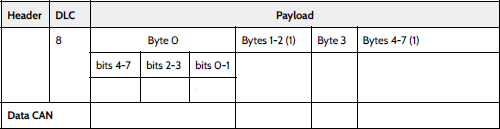
…..

…..

**DR 4**

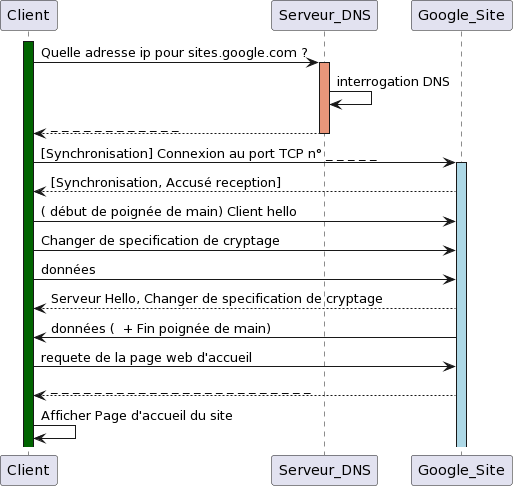


**DR 5**



**DR 6**

Diagramme de séquence lors de la connexion au site de suivi du client.

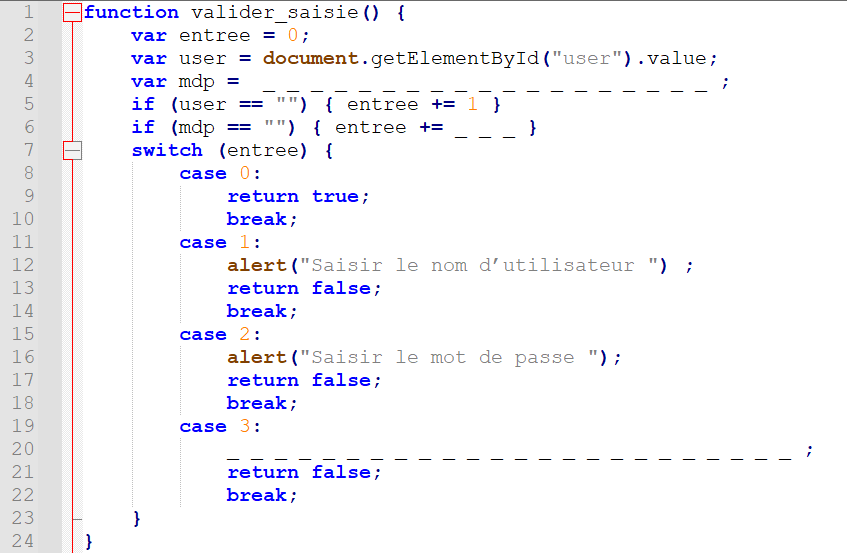


43

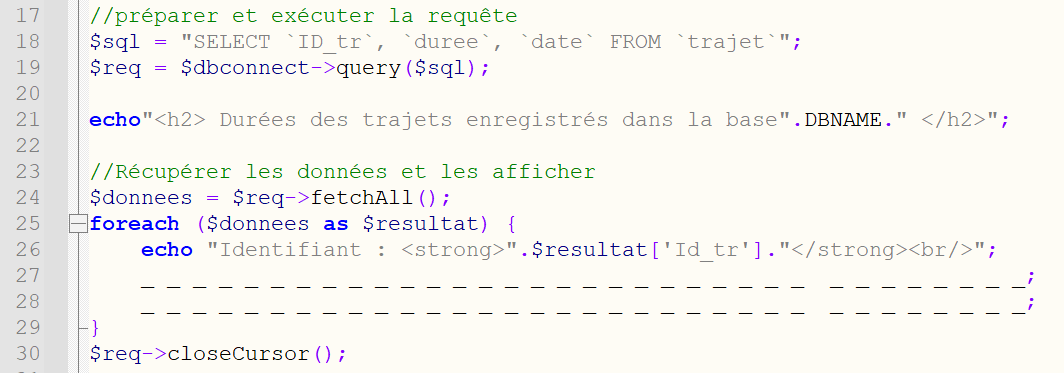
## Tournez la page S.V.P.

**DR 7**

Fichier script.js :



# DR 8



44