

Ville de Laval

Devis technique
Unité de contrôle ATC pour feux de circulation

Préparé par Daniel Magown, ing.
CIMA+

Version 1.0
31 juillet 2023

Ce devis technique a été préparé par CIMA+.

Daniel Magown, ing.

No OIQ : 126018

TABLE DES MATIÈRES

Section	PAGE
1. Généralités	5
1.1 Normes et références.....	5
1.2 Acronymes.....	5
1.3 Définitions.....	6
2. Plate-forme.....	7
2.1 Généralités.....	7
2.2 Exigences fonctionnelles.....	8
2.2.1 Gestion des applications logicielles	8
2.2.2 Accès	9
2.3 Engine Board.....	10
2.4 Interfaces de communication.....	11
2.5 Interface utilisateur, bloc d'alimentation et détail mécaniques	11
2.5.4 L'unité de contrôle peut comporter un interrupteur d'alimentation On/Off.....	12
2.6 Exigences environnementales et procédures d'essai	12
2.7 Matériaux et performances.....	12
2.8 Contrôle de la qualité.....	13
2.9 Outils de développement logiciel	13
3. Logiciel de contrôle.....	13
3.1 Références et généralités.....	13
3.2 Contrôle adaptatif	14
3.2.1 Paramètres des phases	14
3.2.2 Paramètres des détecteurs	19
3.2.3 Paramètres de l'unité de contrôle.....	19
3.2.4 Paramètres de coordination.....	20
3.2.5 Priorité pour autobus (TSP)	23
3.2.6 Paramètres de préemption.....	26
3.2.7 Paramètres des anneaux	27
3.2.8 Paramètres des canaux	27
3.2.9 Paramètres des chevauchements	28
3.2.10 Paramètres des chevauchements piétons	29
3.3 Assignation des entrées et sorties	31

3.4	Contrôle basé sur le temps	32
3.5	Interface utilisateur	33
3.5.1	Généralités	33
3.5.2	Navigation.....	33
3.5.3	Programmation.....	34
3.5.4	Rubriques d'affichage des états	35
3.5.5	Rubriques d'aide	36
3.6	Serveur Web	37
3.7	Scripts logiques.....	38
3.8	Registres des événements.....	40
3.9	Compatibilité NTCIP avec un logiciel de gestion centralisée.....	42
4.	Compatibilité avec les coffrets de contrôle.....	43
5.	Identification.....	43
6.	Documentation	44
7.	Vérifications et essais de conformité.....	44
8.	Livraison	45
9.	Service après-vente.....	46
9.1	Assistance technique.....	46
9.2	Garantie	49
9.3	Pièces	50
9.3.1	Délai de réparation	50
9.4	Mises à jour et mises à niveau.....	51

1. Généralités

1.1 Normes et références

- 1.1.1 ATC 5201 « Advanced Transportation Controller (ATC) » version 6A.37. Toutes les références à la norme ATC 5201 dans le présent devis font référence à la version 6A.37 de cette norme.
- 1.1.2 NEMA TS 2-2021 (Traffic Controller Assemblies with NTCIP Requirements Version 03.08).
- 1.1.3 NTCIP 1201 (National Transportation Communications for ITS Protocol – Global Object (GO) Definitions – version 03)
- 1.1.4 NTCIP 1202 (National Transportation Communication for ITS Protocol – Object Definitions for Actuated Traffic Signal Controller (ASC) Units – version 02.19). Sauf indication contraire, toutes les références à la norme NTCIP 1202 dans le présent devis font référence à la version 2.19 de cette norme.
- 1.1.5 NTCIP 1202 (National Transportation Communication for ITS Protocol – Object Definitions for Actuated Traffic Signal Controller (ASC) Interface – version 03A).
- 1.1.6 NTCIP 1211 (National Transportation Communication for ITS Protocol – Object Definitions for Signal Control and Prioritization (SCP) – version 02.24).

1.2 Acronymes

- 1.2.1 CMU : Moniteur de conflit, soit *Conflict Management Unit* (dans les coffrets NEMA TS 1) ou *Cabinet Management Unit* (dans les coffrets ATC-5301), selon le contexte.
- 1.2.2 CSV : Format de fichier avec données séparées par des virgules (*Comma Separated Values*).
- 1.2.3 FTP : *File Transfer Protocol*
- 1.2.4 FTPS : *File Transfer Protocol Secure*
- 1.2.5 HTML : Format de fichier *HyperText Markup Language*

- 1.2.6 HTTP: *Hypertext Transfer Protocol*
- 1.2.7 HTTPS : *Hypertext Transfer Protocol Secure*
- 1.2.8 MIB : Base des données de gestion des objets (*Management Information Base*)
- 1.2.9 MMU : Moniteur de conflit (*Malfunction Management Unit* dans les coffrets NEMA TS 2)
- 1.2.10 MTQ : Transports Québec, ministère des Transports du Québec ou ministère des Transports et de la Mobilité durable du Québec.
- 1.2.11 PC : Ordinateur personnel de bureau ou portable (*Personal Computer*)
- 1.2.12 PDF : Format de fichier *Adobe Portable Document Format*
- 1.2.13 SCP : *Signal Control and Prioritization* (Voir TSP)
- 1.2.14 TLS: *Transport Layer Security*
- 1.2.15 TRP: Gestion dynamique de la circulation (*Traffic Responsive Programming*)
- 1.2.16 TSP : Ensemble des fonctions de priorité pour autobus (*Transit Signal Priority*), voir aussi SCP
- 1.2.17 TXT : Format de fichier de texte simple ne contenant que des caractères ASCII
- 1.2.18 UC : Unité de contrôle (*Traffic Controller Unit*)
- 1.2.19 USB : Port série universel (*Universal Serial Bus*)
- 1.2.20 XLS ou XLSX : Format de fichier *Microsoft Excel*
- 1.2.21 XML : Format de fichier *Extensible Markup Language*
- 1.2.22 YML : Format de fichier YAML (*YAML Ain't Markup Language*)

1.3 Définitions

- 1.3.1 Base de données : Fichier informatique correspondant à la base de données qui contient tous les paramètres opérationnels de l'unité de contrôle.

- 1.3.2 Contrôleur : Voir Unité de contrôle
- 1.3.3 Élément de menu : Un des éléments listés dans un menu et pointant vers un sous-menu ou vers une rubrique. Le nom de l'élément de menu correspond au titre du sous-menu ou au titre de la rubrique.
- 1.3.4 Intervalle de changement : intervalle du feu jaune
- 1.3.5 Intervalle de dégagement : intervalle du rouge intégral
- 1.3.6 Menu : Liste d'éléments. Peut aussi désigner le titre de la liste d'éléments.
- 1.3.7 Mise à jour : Opération qui consiste à apporter des corrections ou des modifications mineures à un logiciel.
- 1.3.8 Mise à niveau : Opération qui consiste à améliorer les fonctionnalités d'un logiciel.
- 1.3.9 Page : Portion d'une rubrique pouvant être affichée simultanément sur l'écran de l'unité de contrôle.
- 1.3.10 Point de relâchement : *Yield Point*
- 1.3.11 Point limite : *Force Off*
- 1.3.12 Rubrique : Contenu vers lequel pointe un élément de menu et qui affiche de l'information ou des paramètres.
- 1.3.13 Sous-menu : Menu sous un autre menu hiérarchique.
- 1.3.14 Unité de contrôle : Ordinateur spécialisé dont la principale tâche consiste à minuter et déterminer l'affichage de signaux lumineux dans le but de gérer la circulation à un carrefour contrôlé par des feux de circulation.

2. Plate-forme

2.1 Généralités

- 2.1.1 La plate-forme de l'unité de contrôle comprend, sans s'y limiter, les composantes matérielles, le système d'exploitation, les micrologiciels (*firmwares*) et les pilotes (*drivers*).

- 2.1.2 L'Unité de Contrôle (UC) doit respecter les exigences de la norme ATC 5201.
- 2.1.3 Les exigences du présent devis complètent et précisent les exigences de la norme ATC 5201 et NEMA TS 2, sans toutefois en limiter la portée. Aux endroits où les exigences de la norme ATC 5201 et celles de la norme NEMA TS 2 diffèrent, les exigences de la norme ATC 5201 ont préséance. Aux endroits où les exigences de la norme ATC 5201 et celles du présent document diffèrent, les exigences du présent document ont préséance.
- 2.1.4 L'unité de contrôle doit être neuve, être du modèle le plus récent et ne doit pas avoir fait l'objet d'une annonce de fin de vie (*end of life*) ou de fin de vente (*end of sale*) au moment du dépôt de la soumission.

2.2 Exigences fonctionnelles

2.2.1 Gestion des applications logicielles

- 2.2.1.1 L'installation et la mise à jour des applications logicielles peuvent être faites localement seulement.
- 2.2.1.2 Il doit être possible d'effectuer la mise à jour, la mise à niveau et l'installation des applications logicielles et du système d'exploitation à partir d'un ordinateur relié à l'unité de contrôle par le port Ethernet, et ce, sans nécessiter de logiciel externe qui n'est pas inclus dans une installation standard de Windows.
- 2.2.1.3 Il doit être possible d'effectuer la mise à jour et l'installation des applications logicielles et du système d'exploitation à partir d'une clé USB.
- 2.2.1.4 Pour l'installation et la mise à jour des applications logicielles et pour l'installation et la mise à niveau des logiciels du système d'exploitation, les transferts de fichiers avec une clé USB doivent pouvoir être amorcés à l'aide du clavier et de l'écran du panneau frontal.
- 2.2.1.5 L'UC doit gérer automatiquement l'heure avancée d'été. Il doit être possible de désactiver l'ajustement automatique de l'heure avancée d'été.

- 2.2.1.6 L'UC doit supporter au minimum tous les modes suivants (déscrits à l'article 3.1.4 de la norme ATC-5201) pour la configuration et la vérification des paramètres des applications (sans nécessiter de logiciel externe qui n'est pas inclus dans une installation standard de Windows) :
 - 2.2.1.6.1 Écran et clavier du panneau frontal de l'UC;
 - 2.2.1.6.2 Port Ethernet relié à un ordinateur portable local;
 - 2.2.1.6.3 Port Ethernet relié à un ordinateur distant.
- 2.2.1.7 En plus de pouvoir transférer les bases de données des paramètres d'applications via un port de communication Ethernet (localement ou à distance) conformément à l'article 3.1.5 de la norme ATC-5201, il doit être possible de copier les bases de données des paramètres d'applications avec une clé USB, incluant les scripts logiques.
- 2.2.1.8 Lorsqu'une modification de la programmation est en cours sur le clavier ou par téléchargement, l'UC doit continuer de fonctionner normalement.
- 2.2.1.9 L'interface logicielle pour le contrôle par l'usager des applications logicielles (démarrage et arrêt) doit résider sur l'unité de contrôle. Cette interface doit permettre à l'usager de sélectionner les services et applications qui doivent être lancées automatiquement au démarrage.

2.2.2 Accès

- 2.2.2.1 La programmation et la configuration de l'unité de contrôle doivent pouvoir être sécurisées par un code d'accès (mot de passe).
 - 2.2.2.1.1 Les codes pour accéder aux niveaux supérieurs doivent être programmables par l'utilisateur.
 - 2.2.2.1.2 Les codes d'accès doivent avoir une longueur minimale de 4 caractères.
 - 2.2.2.1.3 Les codes d'accès doivent pouvoir être entrés au clavier de l'unité de contrôle.

- 2.2.2.1.4 Lorsque l'unité de contrôle est réinitialisée avec les paramètres par défaut (*factory settings*), les codes d'accès doivent être inhibés afin de permettre l'accès par l'utilisateur ou être initialisés avec un code d'accès par défaut documenté (tel que « admin », « 0000 », « 1111 » ou « 1234 »).
- 2.2.2.2 L'unité de contrôle doit comprendre au moins deux niveaux d'accessibilité.
- 2.2.2.2.1 Le plus bas niveau doit limiter l'usager à la consultation des données en lecture seulement afin de protéger le fonctionnement opérationnel de l'unité de contrôle. Ce niveau doit permettre la lecture et l'affichage de tous les paramètres programmés (incluant, sans s'y limiter, les scripts logiques). L'accès au plus bas niveau ne doit nécessiter aucun code d'accès (mot de passe). Le plus bas niveau peut permettre de transférer une copie de la base de données vers une clé USB, mais ne doit pas permettre la copie d'une base de données vers l'unité de contrôle.
- 2.2.2.2.2 Le plus haut niveau doit donner accès à tous les menus, à toutes les rubriques, à toutes les pages, à tous les paramètres de programmation, à toutes les fonctions et à tout autre élément, incluant, sans s'y limiter, le changement des codes d'accès (mot de passe), la mise à niveau et la mise à jour des composantes logicielles (incluant, sans s'y limiter, le logiciel de contrôle, les micrologiciels et le système d'exploitation) et la gestion des bases de données des paramètres programmés.

2.3 Engine Board

- 2.3.1 Le BSP (*Board Support Package*) doit être fourni avec l'unité de contrôle même s'il n'est pas requis pour le fonctionnement du logiciel de contrôle fourni par le manufacturier.

- 2.3.2 L’interface logicielle pour le contrôle par l’usager des applications logicielles doit permettre de visualiser les adresses MAC des ports Ethernet.

2.4 Interfaces de communication

- 2.4.1 L’unité de contrôle doit comporter un port C12S pour communiquer avec les composantes d’un coffret de contrôle ATC.
- 2.4.2 L’unité de contrôle doit comporter un port (tel que C13S ou NEMA port 2 Terminal ou autre) capable de recevoir les messages NMEA 0183 d’un capteur GPS (avec ou sans signal PPS au choix du manufacturier de l’unité de contrôle).
- 2.4.3 L’unité de contrôle doit être capable de se connecter via un port Ethernet à un serveur NTP pour la synchronisation du temps.
- 2.4.4 L’unité de contrôle doit correspondre à la configuration :
- 2.4.4.1 « A1N », soit Actuated/NTCIP Type 1, lorsque les unités de contrôle sont fournies via un entrepreneur dans le cadre d’un appel d’offres de construction;
- 2.4.4.2 « A2N », soit Actuated/NTCIP Type 2, lorsque l’unité de contrôle est fournie si elle est fournie directement à la Ville dans le cadre d’un appel d’offres d’achat de matériaux pour l’entretien; les unités de contrôle de type A2N doivent être fournies avec un adaptateur pour le connecteur MS-A.
- 2.4.5 Le réceptacle pour clé de données (Datakey KeyceptacleTM) n’est pas exigé.

2.5 Interface utilisateur, bloc d’alimentation et détail mécaniques

- 2.5.1 Toutes les fonctionnalités (incluant, mais sans s’y limiter, les utilitaires de mise à jour et de mise à niveau, les rubriques d’aide, les états, les paramètres de configuration et les scripts logiques) de l’unité de contrôle doivent être accessibles via le clavier et l’écran de l’unité de contrôle.
- 2.5.2 Toutes les fonctionnalités de l’unité de contrôle doivent être accessibles au moyen d’un ordinateur portable.

- 2.5.3 Les fonctionnalités de l'unité de contrôle peuvent être accessibles au moyen d'une application sur une tablette ou un téléphone intelligent.
- 2.5.4 L'unité de contrôle peut comporter un interrupteur d'alimentation On/Off.
- 2.5.5 L'unité de contrôle doit être alimentée par le connecteur MS A.
- 2.5.6 Le panneau frontal (comprenant l'écran et le clavier d'entrée de données) de l'UC doit être intégré (fixe) à l'UC. Un clavier amovible est acceptable s'il est inclus avec l'UC, et si un mécanisme permet de fixer le clavier amovible à l'UC.
- 2.5.7 L'écran doit comporter un minimum de **16 lignes** avec 40 caractères chacune.
- 2.5.8 L'unité de contrôle doit être conforme aux dimensions stipulées dans la norme NEMA TS 2 pour installation sur tablette (shelf mount) et aux autres exigences de l'article 5.7 de la norme ATC 5201.

2.6 Exigences environnementales et procédures d'essai

- 2.6.1 Les essais spécifiés à l'article 7.1 de la norme ATC 5201 doivent pouvoir être réalisés à partir de l'écran et du clavier de l'unité de contrôle.

2.7 Matériaux et performances

- 2.7.1 L'unité de contrôle ne doit comporter aucune protubérance à l'arrière (inclus, sans s'y limiter, toute quincaillerie, boulon, écrou ou boulon moleté) qui pourrait faire contact avec des borniers présents sur une plaque installée à l'intérieur du dos du coffret de contrôle.
- 2.7.2 Tous les fusibles doivent être accessibles sur le panneau frontal de l'UC.
- 2.7.3 En plus des exigences de l'article 8.1.13.1 de la norme ATC 5201, les indicateurs lumineux doivent être clairs, et leur conception ne doit pas dépendre des principes de réflexion ou de diffusion. Ils ne doivent pas apparaître allumés lorsqu'exposés à la lumière du soleil.
- 2.7.4 Le clavier et l'écran du l'UC doivent être en mesure d'opérer à des températures pouvant descendre jusqu'à -40 degrés Celsius.

- 2.7.5 Le temps de réponse de l'écran doit être assez rapide pour permettre la lecture adéquate de chiffres qui changent à chaque 0,1 seconde.
- 2.7.6 L'écran doit maintenir ses performances (contraste et temps de réponse) constantes jusqu'à -37°C. L'écran peut comprendre des éléments chauffants intégrés afin d'atteindre le niveau de performance stipulé.

2.8 Contrôle de la qualité

- 2.8.1 Les procédures de contrôle de la qualité doivent être soumises avant la mise en production. Chaque unité de contrôle doit être livrée avec un rapport d'essais de conformité prévu dans les procédures de contrôle de la qualité. Le rapport doit comprendre le résultat, la procédure de contrôle de la qualité et le nom de l'inspecteur.

2.9 Outils de développement logiciel

- 2.9.1 Le fabricant de l'UC doit fournir tous les outils de développement logiciel conformément aux exigences de l'annexe A de la norme ATC 5201.
- 2.9.2 Le fabricant de l'UC doit fournir tous les utilitaires requis pour supporter l'installation, à l'aide d'une clé USB, du système d'exploitation, des drivers, du BSP, du chargeur de démarrage (*bootloader*) et des utilitaires spécifiques au fabricant.

3. Logiciel de contrôle

3.1 Références et généralités

- 3.1.1 Le logiciel de contrôle doit supporter entièrement tous les objets obligatoires et (sauf indication contraire dans le présent document) tous les objets facultatifs des normes NTCIP 1201 et NTCIP 1202. Tous les objets dont le mode d'accès est « read-write » doivent être accessibles en mode lecture **et** en mode écriture (c'est-à-dire que le mode écriture n'est pas facultatif).
- 3.1.2 Le logiciel de contrôle doit supporter un sous-ensemble des objets de la norme NTCIP 1211 (ou des objets propriétaires équivalents).

- 3.1.3 Les exigences du présent devis complètent et précisent les exigences des normes NEMA TS 2 et NTCIP, sans toutefois en limiter la portée. Aux endroits où les exigences de la norme NEMA TS 2 et celles des normes NTCIP diffèrent, les exigences des normes NTCIP ont préséance. Aux endroits où les exigences des normes NEMA TS 2 ou NTCIP et celles du présent document diffèrent, les exigences du présent document ont préséance.
- 3.1.4 Le logiciel de contrôle ne doit pas avoir fait l'objet d'une annonce de fin de vie (*end of life*) ou de fin de vente (*end of sale*) au moment du dépôt de la soumission.
- 3.1.5 L'unité de contrôle doit être préconfigurée avec, au choix de la Ville, soit la version la plus récente du logiciel de contrôle, soit une version spécifique du logiciel de contrôle compatible avec le système de gestion centralisée. Si l'unité de contrôle est préconfigurée avec une version spécifique du logiciel de contrôle antérieure à la plus récente version, les fichiers (incluant les licences d'utilisation) pour la mise à jour ou la mise à niveau vers la version la plus récente doivent être fournis avec l'unité de contrôle.

3.2 Contrôle adaptatif

3.2.1 Paramètres des phases

- 3.2.1.1 Le nombre maximal de phases (`maxPhases`) doit être égal ou supérieur à 16.
- 3.2.1.2 Chaque phase doit pouvoir comprendre des intervalles d'engagement et de dégagement piétonnier pouvant fonctionner en parallèle avec les intervalles véhiculaires conformément à l'article 3.5.3.2.2 de NEMA TS 2.
- 3.2.1.3 Il doit être possible de retarder l'affichage du feu vert pour une durée programmable par rapport au début de l'engagement piétonnier (le feu rouge doit rester allumé). Ce paramètre doit être directement programmable pour chaque phase sans nécessiter l'utilisation d'un script logique. La durée de cette protection piétonne (avance rouge) doit être indépendante du temps d'engagement piétonnier. Le minutage du vert

minimum doit débuter au moment de l'affichage du feu vert à la fin de la protection piétonne.

3.2.1.4 L'UC doit avoir la capacité de minuter un temps d'engagement piétonnier alternatif (*Walk2*). Ce paramètre peut être programmé à même chaque plan de minutage ou dans une action commandée. L'application du temps d'engagement piétonnier alternatif peut être réalisée à l'aide d'un script logique.

3.2.1.5 Il doit être possible de programmer une durée minimale pour un intervalle vert entre la fin du dégagement piétonnier et le début de l'intervalle de changement (jaune) ou d'un intervalle d'engagement piétonnier (marcheur) servi à nouveau durant la même phase. La durée de cet intervalle vert combiné avec la main fixe ne doit pas être limitée par le chronométrage du vert maximum et doit pouvoir être paramétrable selon les plans de minutage. Cette fonction peut être réalisée à l'aide d'un script logique.

3.2.1.6 Il doit être possible de programmer un intervalle de changement véhiculaire (jaune) de 0 seconde pour chacune des phases.

3.2.1.7 Pour le fonctionnement en mode « Volume/Densité », le paramètre `phaseReduceBy` est facultatif.

3.2.1.8 Phases de démarrage (`phaseStartup`) :

3.2.1.8.1 Lorsque le feu vert est retardé pour protéger les piétons (avance rouge), les phases dont le mode de démarrage est `greenWalk` doivent démarrer au début de l'intervalle d'engagement piétonnier et elles doivent afficher le feu vert après l'intervalle de protection piétonne.

3.2.1.8.2 Lorsqu'un intervalle de priorité pour autobus (`QueueJump`) est inséré au début d'une phase, les phases dont le mode de démarrage est `greenWalk` ou `greenNoWalk` doivent démarrer au début de l'intervalle du feu vert. L'UC

peut omettre le service des priorités pour autobus durant le premier cycle au démarrage. Cette fonction peut être réalisée à l'aide d'un script logique s'il fait partie intégrante de la base de données des paramètres de configuration de l'UC.

3.2.1.8.3 Il peut être possible d'omettre des phases durant le premier cycle au démarrage (facultatif).

3.2.1.8.4 Il peut être possible d'omettre des chevauchements (véhiculaires et piétons) durant le premier cycle au démarrage (facultatif).

3.2.1.9 Options des phases (phaseOptions) :

3.2.1.9.1 Lorsqu'un rappel maximum (phaseOptions, Bit 7 : Max Vehicle Recall) est placé, le chronométrage du vert maximum doit débuter comme s'il y avait un appel constant sur une autre phase du même anneau, mais la phase ne doit se terminer que s'il y a réellement un appel sur une autre phase du même anneau, conformément à l'article 3.5.3.5 de NEMA TS 2.

3.2.1.9.2 L'unité de contrôle doit comporter un paramètre (*Walk Rest Modifier*) pour activer et désactiver l'extension de la période d'engagement piétonne. Ce paramètre doit pouvoir être activé par l'horloge (TOD). Les intervalles d'engagement piétonnier des phases CNA (*Non-Actuated 1* et *Non-Actuated 2*) doivent utiliser le temps de vert lorsque le paramètre *Walk Rest Modifier* est activé.

3.2.1.10 Un paramètre (noServePhases) doit permettre d'empêcher deux phases d'être servies simultanément même si elles sont compatibles entre-elles selon la matrice de compatibilité des phases (phaseConcurrency). Cette fonction peut être réalisée à même la matrice de compatibilité des phases.

- 3.2.1.11 La matrice de compatibilité des phases (`phaseConcurrency`) peut permettre qu'une phase soit compatible avec des phases qui sont mutuellement incompatibles entre elles (facultatif).
- 3.2.1.12 Il doit être possible d'omettre dynamiquement une ou plusieurs phases pendant le service d'une phase sélectionnée. Cette fonction doit faire partie des menus de l'unité de contrôle et ne doit pas nécessiter l'utilisation de scripts logiques. Les omissions dynamiques doivent pouvoir être activées globalement et par plage horaire (TOD). L'activation par plage horaire peut être réalisée à l'aide de séquences alternatives. Les omissions dynamiques doivent être opérationnelles en mode libre et en mode coordonné ainsi que durant le service d'une priorité pour autobus et le service d'une préemption (il peut être nécessaire de forcer un intervalle rouge intégral pour entrer en préemption).
- 3.2.1.13 Il doit être possible de placer un rappel dynamique sur une ou plusieurs phases lors du service d'une phase sélectionnée. Cette fonction peut être réalisée par un script logique seulement s'il fait partie intégrante de la base de données des paramètres de configuration de l'unité de contrôle. Les rappels dynamiques doivent pouvoir être activés globalement et par plage horaire (TOD). L'activation par plage horaire peut être réalisée à l'aide de séquences alternatives. Les rappels dynamiques doivent être opérationnels en mode libre et en mode coordonné ainsi que durant le service d'une priorité pour autobus et le service d'une préemption.
- 3.2.1.14 L'UC doit comprendre un minimum de 4 plans de minutages (`Timing Plans` ou `Banks`) pouvant être sélectionnés par l'horloge (TOD) et par le plan de coordination. Chaque plan de minutage doit inclure, sans s'y limiter, les paramètres suivants pour chaque phase :
- 3.2.1.14.1 Numéro de la phase (`phaseNumber`);
- 3.2.1.14.2 Temps d'engagement piétonnier (`phaseWalk`);

- 3.2.1.14.3 Temps de dégagement piétonnier (phasePedestrianClear);
- 3.2.1.14.4 Durée du vert minimum (phaseMinimumGreen);
- 3.2.1.14.5 Extension véhiculaire (phasePassage);
- 3.2.1.14.6 Durée du vert maximum 1 (phaseMaximum1);
- 3.2.1.14.7 Durée du vert maximum 2 (phaseMaximum2);
- 3.2.1.14.8 Temps de changement (phaseYellowChange); ce paramètre est facultatif sauf pour le premier plan de minutages;
- 3.2.1.14.9 Temps de dégagement (phaseRedClear);
- 3.2.1.14.10 Rappel en douceur (phaseOptions, bit 9 : Soft Vehicle Recall);
- 3.2.1.14.11 Rappel piéton (phaseOptions, bit 8 : Ped Recall);
- 3.2.1.14.12 Rappel maximum (phaseOptions, bit 7 : Max Vehicle Recall);
- 3.2.1.14.13 Rappel minimum (phaseOptions, bit 6 : Min Vehicle Recall);
- 3.2.1.14.14 Détection volatile (phaseOptions, bit 5 : Non Lock Detector Memory);
- 3.2.1.14.15 Délai de l'affichage du feu vert par rapport au début de l'engagement piétonnier; si ce délai n'est pas paramétrable pour chaque plan de minutages, il doit pouvoir être paramétrable par plage horaire (TOD) ou par une action commandée.

3.2.1.15 N'importe quelle phase ou chevauchement doit pouvoir faire éteindre le jaune de n'importe quelle autre phase ou chevauchement sélectionnés (yellowBlanking). Cette fonction peut être réalisée au moyen d'un script logique.

3.2.2 Paramètres des détecteurs

3.2.2.1 Le nombre maximal de détecteurs (maxVehicleDetectors) doit être égal ou supérieur à 64.

3.2.2.2 Chaque détecteur doit pouvoir placer des appels sur plus d'une phase.

3.2.2.3 Le détecteur doit pouvoir transférer les appels à une phase alternative désignée lorsque cette dernière est au vert et que la phase assignée au détecteur n'est pas au vert.

3.2.2.4 En plus des phases appelées par un détecteur, ce dernier doit pouvoir placer un appel sur une autre phase si les phases sélectionnées sont vertes. Cette fonction peut être réalisée à l'aide d'un script logique.

3.2.2.5 Le nombre maximal de détecteurs piétons (maxPedestrianDetectors) doit être égal ou supérieur à 16.

3.2.2.6 Chaque détecteur piéton doit pouvoir placer des appels sur plus d'une phase piétonne. Cette fonction peut être réalisée à l'aide d'un script logique s'il fait partie intégrante de la base de données des paramètres de l'UC.

3.2.3 Paramètres de l'unité de contrôle

3.2.3.1 Un paramètre doit être prévu pour l'identification du carrefour sous forme d'un nombre entier ou d'une chaîne de caractères hexadécimaux. Ce paramètre doit permettre un minimum de 65 535 identifiants. Ce paramètre peut être lié à l'adresse IP (port Système) de l'UC (par encodage d'une partie ou de la totalité de l'adresse IP). Il est souhaitable que l'UC comporte aussi un ou plusieurs paramètres sous forme de chaînes de caractères alphanumériques permettant d'identifier la ville et les axes routiers.

- 3.2.3.2 Durant le démarrage, le contrôleur doit être capable de minuter une période de clignotement et un intervalle tout rouge (d'une durée programmable) avant d'afficher l'intervalle sélectionné des phases de démarrage.
- 3.2.3.3 Le point de décision pour la phase suivante doit être à la fin du feu vert. L'unité de contrôle peut comporter un paramètre facultatif programmable afin de permettre au contrôleur de modifier le choix de la prochaine phase durant les intervalles de changement (jaune) et de dégagement (rouge intégral).
- 3.2.3.4 L'unité de contrôle doit comporter un minimum de 8 fonctions spéciales (`maxSpecialFunctionOutputs`). Chaque fonction spéciale doit pouvoir être utilisée pour activer d'autres fonctions de l'unité de contrôle.

3.2.4 Paramètres de coordination

- 3.2.4.1 Pour le mode d'opération de la coordination (`coordOperationalMode`), une valeur de 253 peut être utilisée pour forcer le clignotement en mode CVM Flash.
- 3.2.4.2 Pour le mode de correction du décalage (`coordCorrectionMode`), l'UC doit avoir la capacité de rattraper son décalage (avec addition ou soustraction de temps selon le chemin le plus court ou avec addition seulement de temps, au choix de l'usager) lui permettant de se synchroniser en 6 cycles au maximum. Il doit être possible de programmer le pourcentage maximum du cycle pouvant être ajusté durant la correction du décalage. Il peut aussi être possible de programmer pour chaque phase le temps maximum pouvant être ajouté et le temps maximum pouvant être retranché.
- 3.2.4.3 Il doit être possible de programmer le mode des points limites (`coordForceMode`) pour chaque plan de coordination.
- 3.2.4.4 L'unité de contrôle doit comporter un minimum de 48 plans de coordination (`maxPatterns`) numérotés séquentiellement.

- 3.2.4.5 Tous les plans de coordination (patterns) doivent être indépendants les uns des autres, c'est-à-dire que l'unité de contrôle doit supporter au minimum le mode « patterns (2) » de l'objet patternTableType.
- 3.2.4.6 Le nombre maximal de répartitions (maxSplits) doit être égal ou supérieur à 32. Il est souhaitable que le nombre de répartitions soit égal au nombre de plans de coordination.
- 3.2.4.7 Lorsque les paramètres splitTime d'un plan de coordination (*pattern*) sont utilisés comme vert maximum (plan de coordination utilisé en mode libre parce que cycleTime est égal à 0), il doit être possible de programmer un vert maximum aussi petit que 5 secondes dans le paramètre splitTime.
- 3.2.4.8 En plus des modes des répartitions (splitMode) spécifiés dans NTCIP 1202, l'UC doit permettre de forcer une phase à demeurer au vert (*no early release*) jusqu'au point limite (*force off*) en mode coordonné et jusqu'au vert maximum applicable. Cette fonction peut être réalisée à l'aide d'un script logique.
- 3.2.4.9 L'UC doit permettre de programmer une phase de coordination (splitCoordPhase) pour chaque anneau.
- 3.2.4.10 Le point de référence de la coordination (patternCoordSyncPoint et unitCoordSyncPoint), soit le début du cycle local (*local zero*), doit pouvoir être référé, au minimum, selon les modes suivants :
- 3.2.4.10.1 Début du feu vert de la première phase coordonnée (firstCoordPhsGrnBegin);
- 3.2.4.10.2 Début du feu vert de la dernière phase coordonnée (lastCoordPhsGrnBegin); cette option est facultative.
- 3.2.4.10.3 Fin du feu vert de la première phase coordonnée (firstCoordPhsGrnEnd); cette option est facultative.

3.2.4.11 Il peut être possible de programmer un décalage entre les anneaux (facultatif).

3.2.4.12 L'UC doit calculer automatiquement les fenêtres permissives en fonction du mode sélectionné. Un mode de permissives doit pouvoir être programmé pour chaque plan de coordination (*pattern*). L'UC doit pouvoir fonctionner avec les modes suivants pour la détermination des fenêtres permissives :

3.2.4.12.1 Permissive unique : une seule fenêtre permissive pour toutes les phases non coordonnées; la durée de la fenêtre permissive doit être programmable (elle peut être programmable en pourcentage du cycle ou en secondes);

3.2.4.12.2 Permissives flexibles : pour chaque anneau, les fenêtres permissives de toutes les phases compatibles non coordonnées sont ouvertes en même temps à la fin de la phase coordonnée, et chacune est refermée lorsqu'il ne reste plus suffisamment de temps pour servir la phase correspondante en respectant le vert minimum, l'engagement piétonnier, le dégagement piétonnier, l'intervalle de changement jaune et l'intervalle de dégagement rouge intégral;

3.2.4.12.3 Permissives par phases : pour chaque anneau, une seule fenêtre permissive n'est ouverte à la fois au point fixe dans le cycle pour la phase correspondante et elle est refermée lorsqu'il ne reste plus suffisamment de temps pour servir la phase correspondante en respectant le vert minimum, l'engagement piétonnier, le dégagement piétonnier, l'intervalle de changement jaune et l'intervalle de dégagement rouge intégral; ce mode doit offrir un minimum de deux fenêtres permissives par anneau, mais il est souhaitable que ce mode offre une fenêtre permissive pour chaque phase.

- 3.2.4.13 Il doit être possible de programmer un point de relâchement (*yield point*) pour les phases coordonnées de chaque plan de coordination (*pattern*) afin de permettre à une phase coordonnée de se terminer (*gap-out*) avant le point limite (*force off*).
- 3.2.4.14 L'UC doit permettre de servir la phase piétonne même lorsque la somme des temps d'engagement piétonnier (*walk*), de dégagement piétonnier (*ped clear*), de changement (jaune) et de dégagement (rouge intégral) est égale ou supérieure à la répartition du temps (*split*) de la phase correspondante. Ce comportement doit être possible pour au moins une phase, et il peut être nécessaire d'activer un paramètre pour le permettre. Au besoin, l'unité de contrôle doit se resynchroniser.

3.2.5 Priorité pour autobus (TSP)

- 3.2.5.1 L'UC doit pouvoir offrir des fonctions TSP (*Transit Signal Priority*) afin d'implanter un programme ajusté de feux priorisant le transport en commun sans perdre la coordination (aucun bris de cycle) de l'UC. Les fonctions doivent être inspirées de la norme NTCIP 1211 et inclure les objets NTCIP 1211 ou des objets propriétaires équivalents pour les stratégies de priorité et les canaux, les phases véhiculaires et piétonnes omises durant chaque stratégie, les tables d'extension TSP pour les répartitions et les temps maximum de réduction et d'extension. Les fonctions doivent aussi comprendre les objets requis pour placer les demandes et informer des états.
- 3.2.5.2 L'UC doit comporter un paramètre pour activer et désactiver globalement les fonctions TSP.
- 3.2.5.3 Il doit être possible de programmer un plan de coordination (*pattern*) qui est effectif par défaut lorsque l'UC fonctionne en mode libre (c'est-à-dire lorsque l'UC opère en mode libre par l'application du plan de coordination 254) pour utiliser les paramètres TSP qui y sont associés. Lorsque l'UC fonctionne en mode libre par un autre plan de coordination (avec un cycle à 0 secondes), l'UC doit utiliser les paramètres TSP de chaque plan de coordination correspondant.

- 3.2.5.4 L'UC doit comporter un minimum de 6 canaux de TSP. Les canaux sont généralement associés à une approche.
- 3.2.5.5 L'UC doit comporter un minimum de 6 stratégies de TSP. Les stratégies permettent de sélectionner les phases à prioriser ou à réduire et de configurer le fonctionnement de chaque canal pour une plage horaire. Les stratégies sont activées en fonction de l'horaire (TOD). Les stratégies de TSP peuvent être liées aux canaux de TSP.
- 3.2.5.6 Les fonctions TSP « extension du vert » et « troncation du rouge » doivent être disponibles au minimum.
- 3.2.5.7 Toutes les phases (coordonnées ou non) doivent pouvoir être associées à une stratégie de TSP.
- 3.2.5.8 Il doit y avoir une corrélation directe entre le numéro (patternSplitNumber) d'une série de répartitions de temps (splitTable) et le numéro de la série de temps pour les priorités pour autobus (priorityStrategyExtensionToSplitTable). Ainsi, le numéro de la série de temps pour les priorités est soit le même que celui des répartitions de temps, soit programmé dans le plan de coordination (pattern) ou dans la série de répartitions (splitTable).
- 3.2.5.9 L'unité de contrôle doit utiliser le temps estimé d'arrivée de l'autobus (*Time of Service Desired* ou TSD) afin de déterminer s'il est pertinent de prolonger le feu vert afin d'éviter de prolonger le feu vert inutilement si l'autobus est trop loin du carrefour (la demande activera plutôt la troncation du rouge). Il doit être possible de programmer un temps estimé d'arrivée de l'autobus (TSD) si cette information n'est pas transmise à même la demande de priorité ou si les demandes sont configurées pour fonctionner en mode « *Check-in/Check-out* ».
- 3.2.5.10 Les messages pour activer les demandes doivent pouvoir être reçus d'une centrale de gestion des feux. Le protocole utilisé peut être conforme à la norme NTCIP 1211 ou être un protocole propriétaire. Les messages

doivent comporter le temps estimé d’arrivé de l’autobus : *Time of Service Desired* (TSD). La capacité de recevoir des demandes de priorité d’une centrale de gestion est feux est facultative.

3.2.5.11 Les fonctions TSP de l’UC doivent être en mesure de fonctionner sous le principe de signaux *Check-in* et *Check-out* transmis par un système externe. Le signal doit fonctionner, au choix de l’usager, soit selon un signal *Check-in* constant (un *Check-out* est appliqué lorsque le signal du *Check-in* est retiré), soit selon des signaux *Check-in* et *Check-out* distincts. Lorsque les signaux sont distincts, une fois le signal *Check-in* reçu, l’UC passe en mode TSP jusqu’à ce que :

3.2.5.11.1 Le signal *Check-out* soit reçu ;

3.2.5.11.2 Le signal de *Check-out* soit prématièrement activé par le système externe en cas d’annulation de demande ;

3.2.5.11.3 Ou encore que le délai maximal (en secondes) choisi par l’usager soit atteint.

3.2.5.12 Dès que le service de la demande de priorité est terminé, l’UC doit revenir en mode de fonctionnement normal sans avoir perdu sa coordination. Le service de demande de priorité doit se terminer au plus tard durant le cycle suivant la fin de la demande de priorité (qui survient au moment du passage de l’autobus au carrefour).

3.2.5.13 Les priorités pour autobus doivent pouvoir appliquer un délai avant de considérer une demande de priorité, doivent avoir un temps de service maximum, doivent pouvoir ignorer les demandes d’un canal pendant un temps programmé après le service d’une demande sur le même canal.

3.2.5.14 L’UC doit comporter un paramètre pour empêcher le service de toute demande de priorité pour autobus pendant un temps programmé après le service d’une demande précédente.

3.2.5.15 Il doit être possible d’associer une avance pour autobus (*Queue Jump*) à chaque canal de TSP et y associer la phase correspondante. La durée de

l'avance pour autobus doit être programmée par stratégie afin de pouvoir être ajustée selon les plages horaires (TOD).

3.2.5.16 Les fonctions TSP doivent pouvoir fonctionner avec tous les modes de permissives spécifiés au présent devis.

3.2.5.17 Les fonctions TSP doivent pouvoir fonctionner avec tous les modes de points limites (*force off*) spécifiés au présent devis.

3.2.5.18 Les fonctions TSP de l'UC ne doivent pas empêcher le bon fonctionnement de la gestion dynamique (*Traffic Responsive Programming*).

3.2.5.19 L'UC doit collecter des mesures d'efficacité sur le fonctionnement du TSP (demandes servies, demandes non complétées, durée des demandes actives, etc.).

3.2.5.20 Le mode `maxInhibit` du `coordMaximumMode` doit s'appliquer aussi au fonctionnement du TSP.

3.2.5.21 Le service d'une demande de priorité doit aussi prolonger le vert maximum (en mode coordonné et en mode libre).

3.2.5.22 Les fonctions TSP doivent pouvoir être ignorées en cas d'appel de feux sonores. Cette fonction peut être réalisée à l'aide d'un script logique.

3.2.5.23 Les entrées TSP doivent pouvoir être désactivées par l'utilisateur. À la réception de l'unité de contrôle, toutes les fonctions TSP de l'UC doivent être programmées à « off » par défaut.

3.2.6 Paramètres de préemption

3.2.6.1 Le nombre maximum de préemptions (`maxPreempts`) doit être égal ou supérieur à 6. Les préemptions 1 et 2 sont généralement utilisées pour les préemptions ferroviaires, et les préemptions 3 à 6 sont généralement utilisées pour les véhicules d'urgence.

- 3.2.6.2 En plus de pouvoir configurer des phases de sortie (`preemptExitPhase`), il doit être possible, pour chaque préemption, de configurer l'unité de contrôle pour sortir « en coordination » (c'est-à-dire que l'unité de contrôle sélectionne les phases de sortie en fonction afin de ne pas avoir à se resynchroniser) tout en respectant les minutages tel que le vert minimum, l'intervalle de changement (jaune) et l'intervalle de dégagement (rouge intégral).
- 3.2.6.3 L'UC doit avoir un paramètre pour maintenir la préemption active pendant un temps programmé après la perte du signal de préemption.
- 3.2.6.4 L'UC doit permettre d'assigner des sorties (*outputs*) à l'état de chaque préemption. Pour chaque préemption, il doit être possible de choisir durant quels intervalles de préemption (`preemptState`) la sortie doit être active.
- 3.2.6.5 L'utilisateur doit pouvoir activer et désactiver chaque préemption. À la réception de l'unité de contrôle, toutes les entrées de préemption de l'UC doivent être programmées à « off » par défaut.

3.2.7 Paramètres des anneaux

- 3.2.7.1 Le nombre maximal d'anneaux (`maxRings`) doit être égal ou supérieur à 4.
- 3.2.7.2 Le nombre maximal de séquences des phases (`maxSequences`) doit être égal ou supérieur à 16.

3.2.8 Paramètres des canaux

En plus des types de contrôle (`channelControlType`) stipulés dans la norme NTCIP 1202 (`phaseVehicle`, `phasePedestrian`, `overlap`, `other`), l'UC doit supporter les types de contrôle suivants :

- 3.2.8.1.1 Chevauchement piéton (`pedestrianOverlap`); le contrôle des chevauchements piétons (`pedestrianOverlap`) peut être fait à même l'objet des

chevauchements (*overlap*) avec un type d'*overlap* spécifique pour les chevauchements piétons;

- 3.2.8.1.2 Feu de priorité pour autobus (*queueJump*); ce type de contrôle n'est pas requis si l'intervalle du feu de priorité pour autobus est contrôlé par un chevauchement;
- 3.2.8.1.3 Feux pour autobus (*tramway*). Les fonctions des feux pour autobus (*tramway*) doivent pouvoir être configurées selon les exigences du Tome V (Signalisation routière) des normes sur les ouvrages routiers du MTQ. Cette exigence est facultative.

3.2.9 Paramètres des chevauchements

- 3.2.9.1 Le contrôleur doit permettre la possibilité de programmer jusqu'à 16 chevauchements (identifiés de A à P ou de 1 à 16).
- 3.2.9.2 En plus des types de chevauchement (*overlapType*) spécifiés dans NTCIP 1202, l'UC doit avoir les types suivants :
 - 3.2.9.2.1 Chevauchement au rouge durant les périodes d'engagement et de dégagement des phases sélectionnées parmi les phases parentes (*minusWalkPedClear*);
 - 3.2.9.2.2 Chevauchement au rouge durant les périodes d'engagement et de dégagement des phases sélectionnées parmi les phases parentes et pendant les 3 secondes après la fin du dégagement piétonnier (*minusWalkPedClearMUTCD*); ce type de chevauchement peut être réalisé à l'aide d'un script logique s'il fait partie intégrante de la base de données des paramètres de configuration de l'unité de contrôle.
 - 3.2.9.2.3 Chevauchement au rouge durant la période d'engagement piétonnier (*minusWalkRed*); ce type de chevauchement est facultatif;

- 3.2.9.2.4 Chevauchement éteint durant la période d’engagement piétonnier (`minusWalkDark`). Ce type de chevauchement peut être réalisé à l’aide d’un script logique s’il fait partie intégrante de la base de données des paramètres de configuration de l’unité de contrôle.
- 3.2.9.3 Au moins huit chevauchements doivent pouvoir être programmés pour le clignotement rapide du feu vert. Le feu vert du chevauchement doit clignoter durant les phases sélectionnées parmi les phases parentes. La fréquence doit être de 100 ou 150 pulsations par minute avec un rapport cyclique de 50 %. La fréquence du clignotement peut être un paramètre unique applicable à tous les chevauchements. La fonction de clignotement peut être intégrée à même la fonction du chevauchement ou elle peut être appliquée au niveau de la sortie (*Output*) du chevauchement.
- 3.2.9.4 La fin de chaque chevauchement doit pouvoir être retardée (*Lag Overlap* ou *Double Clearing Overlap* ou *Trailing Overlap*). Pour les chevauchements dont la fin est retardée (`overlapTrailGreen`), lorsque la somme du temps de changement retardé (`overlapTrailYellow`) et du temps de dégagement retardé (`overlapTrailRed`) est égale à zéro, l’UC peut utiliser les temps de changement et de dégagement programmés dans la phase parente.
- 3.2.9.5 Le début de chaque chevauchement doit pouvoir être retardé (*Delayed Overlap*) d’une durée spécifiée après le début d’une phase parente lorsque la phase précédente n’était pas une phase parente. Cette fonction peut être réalisée à l’aide d’un script logique s’il fait partie intégrante de la base de données des paramètres de configuration de l’unité de contrôle.

3.2.10 Paramètres des chevauchements piétons

- 3.2.10.1 L’UC doit permettre la possibilité de programmer jusqu’à 16 chevauchements piétons. Ceux-ci doivent être capables de chevaucher l’affichage des feux pour piétons sur n’importe quelle

combinaison de phases comprenant un mouvement piéton. Si les chevauchements piétons (*pedestrianOverlap*) sont contrôlés par des chevauchements (*overlap*) avec un type de chevauchement spécifique pour les chevauchements piétons, comme stipulé à la version 3 de la norme NTCIP 1202, l'unité de contrôle doit comprendre un minimum de 32 chevauchements (*overlap*).

3.2.10.2 Pour chaque chevauchement piéton, il doit être possible de programmer des phases parentes qui ne sont pas adjacentes, qui ne sont pas sur le même anneau et qui ne font pas partie du même groupe de compatibilité.

3.2.10.3 Le nombre de phases parentes ne doit être limité que par le nombre maximal de phases de l'UC, et ce, pour tous les modes de chevauchement piéton décrits ci-dessous.

3.2.10.4 Les modes de chevauchement piéton suivants doivent être disponibles :

3.2.10.4.1 L'affichage de l'intervalle de dégagement piétonnier du chevauchement piéton débute avec l'intervalle de dégagement piétonnier de la dernière phase parente. Ce mode doit permettre la possibilité à l'intervalle d'engagement piétonnier d'utiliser le temps résiduel des phases (*walk rest*).

3.2.10.4.2 L'affichage de l'intervalle de dégagement piétonnier du chevauchement piéton débute avec l'intervalle de dégagement piétonnier de chaque phase (*always clear*). Ce mode doit permettre la possibilité à l'intervalle d'engagement piétonnier d'utiliser le temps résiduel des phases (*walk rest*). Cette fonction peut être réalisée à l'aide d'un script logique ou avec une phase factice.

3.2.10.4.3 L'affichage des intervalles d'engagement et de dégagement piétonniers du chevauchement piéton est programmable de manière indépendante des intervalles d'engagement et de dégagement piétonniers des phases parentes (mode *carry*-

over). L’engagement piétonnier ne doit débuter que sur les phases sélectionnées parmi les phases parentes. Ce mode ne permet pas de prolonger l’intervalle d’engagement piétonnier.

3.3 Assignation des entrées et sorties

- 3.3.1 Le logiciel doit être capable de réassigner les entrées et les sorties de l’UC. Cette réassignation doit être disponible par le serveur web ou par le clavier de l’UC. Il doit être possible de lire et d’afficher la réassignation existante des entrées et sorties de l’UC.
- 3.3.2 L’UC doit avoir la capacité de rediriger les entrées et sorties par programmation. Par exemple, les entrées de priorité bus doivent pouvoir être associées aux entrées des détecteurs (BIU 9 à 12 dans les coffrets NEMA TS 2 et SIU 9 à 12 dans les coffrets ATC-5301), et les entrées des fonctions spéciales doivent pouvoir être associées aux fonctions *Check-In* et *Check-Out* du TSP.
- 3.3.3 L’UC doit être capable d’associer les entrées de préemption 3 à 6 aux préemptions 3 à 6 lorsque les signaux d’entrée sont constants et aux priorités pour autobus lorsque les signaux d’entrée sont pulsés.
- 3.3.4 Chacune des trois couleurs de chaque canal doit pouvoir être associée à une phase ou un chevauchement (véhiculaire ou piéton) différent.
- 3.3.5 Chacune des trois couleurs de chaque canal doit pouvoir être associée à n’importe quel numéro et n’importe quelle fonction de sortie de l’UC, incluant, sans s’y limiter: Phase Green, Phase Yellow, Phase Red, Phase Walk, Phase Ped Clear/Flash Don’t Walk, Phase Solid Don’t Walk, Overlap Red, Overlap Yellow, Overlap Green, Ped Overlap Red, Ped Overlap Yellow, Ped Overlap Green, Phase On, Auto Flash Status, Preempt Status, etc. Cette assignation doit pouvoir être configurée directement à partir du logiciel de l’unité de contrôle sans avoir recours à un logiciel externe.

3.4 Contrôle basé sur le temps

- 3.4.1 L'heure, la date, le fuseau horaire et l'heure avancée doivent pouvoir être consultées et éditées à même l'interface de l'unité de contrôle (écran et clavier) et via les objets NTCIP correspondants.
- 3.4.2 La référence de synchronisation des plans de coordination (`timebaseAscPatternSync`) doit être réglée sur minuit (00:00) pour le calcul du début du cycle maître (*Cycle Zero*).
- 3.4.3 Lorsque le contrôleur change d'une opération libre à une opération coordonnée, il doit effectuer ce changement d'une façon douce et ordonnée. Il fait de même lorsqu'il change d'une commande de coordination à une autre commande. Les changements de cycle, de décalage et de répartitions ne prennent effet que lorsque l'unité de contrôle atteint le zéro local (soit le point de référence de la coordination).
- 3.4.4 Le nombre maximal de plages de calendrier (`maxTimeBaseScheduleEntries`) doit être égal ou supérieur à 32.
- 3.4.5 Le nombre maximal de programmes journaliers (`maxDayPlans`) doit être égal ou supérieur à 16.
- 3.4.6 Le nombre maximal d'événements par programme journalier (`maxDayPlanEvents`) doit être égal ou supérieur à 48.
- 3.4.7 Le nombre maximal de plans d'actions (`maxTimebaseAscActions`) doit être égal ou supérieur à 48.
- 3.4.8 En plus des éléments stipulés par la norme NTCIP 1202, chaque plan d'actions doit pouvoir commander un minimum de 10 actions forcées (*overrides* ou *commanded actions*). Si les plans d'actions activent des groupes d'actions forcées (plutôt que d'activer directement une liste d'actions forcées), un minimum de 8 groupes d'actions forcées est requis, et chaque groupe doit pouvoir commander un minimum de 10 actions forcées.

- 3.4.9 Les actions forcées doivent permettre de forcer l'application, sans s'y limiter, du vert maximum 2, du *Walk Rest Modifier*, des rappels minimum et maximum, d'une omission de phase, etc.
- 3.4.10 Même si l'UC fonctionne en mode libre, l'horloge intégrée doit pouvoir être utilisée pour commander des fonctions, incluant un plan de minutage, un vert maximum, des fonctions spéciales, des plans d'actions forcées (*overrides*), etc.
- 3.4.11 L'Unité de contrôle doit avoir la possibilité de désactiver l'ajustement du temps selon le fuseau horaire.
- 3.4.12 L'Unité de contrôle doit avoir la possibilité de désactiver l'ajustement du temps selon l'heure avancée d'été.

3.5 Interface utilisateur

3.5.1 Généralités

- 3.5.1.1 Les valeurs de tous les paramètres définis dans NTCIP et de tous les paramètres définis dans le présent devis doivent pouvoir être visualisées à l'écran (soit dans un menu de programmation, soit dans une rubrique d'affichage des états).

3.5.2 Navigation

- 3.5.2.1 Le logiciel de contrôle doit comporter un système de menus hiérarchiques permettant d'accéder aux différents paramètres de configuration et de programmation, aux rubriques d'affichage des états et à toute autre fonction.
- 3.5.2.2 La navigation à travers les menus hiérarchiques doit être possible en appuyant uniquement sur la touche du clavier correspondant au chiffre ou à la lettre de l'élément du menu, et ce, sans nécessiter d'appuyer ensuite sur la touche *ENTER* pour accéder à la rubrique ou au sous-menu correspondant à l'élément du menu choisi.
- 3.5.2.3 Après que l'usager est entré à l'intérieur d'une rubrique, l'unité de contrôle peut, pour accéder à une page d'une table qui comporte plusieurs pages et dont toutes les pages sont identiques :

- 3.5.2.3.1 Soit attendre que l'usager appuie sur *ENTER* après avoir entré les chiffres qui composent le numéro de la page;
- 3.5.2.3.2 Soit nécessiter que l'usager entre rapidement les chiffres composant le numéro de la page.
- 3.5.2.4 Les touches des flèches haut, bas, droite et gauche (*UP ARROW*, *DOWN ARROW*, *RIGHT ARROW* et *LEFT ARROW*) doivent déplacer le curseur directement d'un champ à l'autre à l'intérieur d'une même page ou d'une même rubrique ou, selon le contexte, déplacer le curseur d'un caractère à l'autre dans un même champ ou défiler l'affichage d'une page à l'autre dans une même rubrique.

3.5.3 **Programmation**

- 3.5.3.1 Tous les paramètres de l'UC doivent être accessibles et doivent pouvoir être programmés par le clavier du panneau frontal de l'UC.
- 3.5.3.2 La nomenclature des paramètres à l'écran doit correspondre au nom des objets définis dans les normes NTCIP 1201 et 1202.
- 3.5.3.3 Le logiciel de contrôle peut protéger contre la modification accidentelle de données lorsque l'usager navigue dans les rubriques des paramètres de programmation. Si cette protection est disponible :
 - 3.5.3.3.1 Les rubriques des paramètres de programmation doivent être initialement affichées en mode de consultation seulement peu importe le niveau d'accès de l'usager.
 - 3.5.3.3.2 L'unité de contrôle doit permettre à un usager ayant un niveau d'accès approprié de passer en mode édition en appuyant successivement sur deux touches spécifiques ou en appuyant simultanément sur une combinaison de deux touches spécifiques du clavier de l'unité de contrôle, et ce, directement à partir de la page affichée sans déplacer le curseur.

- 3.5.3.4 Durant l'entrée de données au clavier de l'UC, les nouvelles données doivent être affichées à l'écran.
- 3.5.3.5 Le logiciel de contrôle doit vérifier la valeur des paramètres entrés en fonction de la plage de valeurs acceptée pour ce champ de données. Un signal sonore et visuel doit être émis en cas d'erreur d'entrée de données.
- 3.5.3.6 Le logiciel de contrôle doit vérifier les paramètres entrés dans leur ensemble en faisant des vérifications croisées et il doit afficher des messages d'erreur descriptifs et intelligents.
- 3.5.3.7 Une fonction de copie doit permettre de copier toutes les données d'une phase à une autre. Elle doit également permettre de copier l'ensemble de toutes les données d'un plan de minutage à un autre, d'un plan de coordination à un autre et d'un détecteur à un autre.
- 3.5.3.8 Si un paramètre est modifié et que cette modification requiert un redémarrage de l'UC pour être appliquée, l'UC doit afficher un message à l'écran indiquant qu'un redémarrage est requis pour que les paramètres modifiés soient appliqués, et ce message doit rester affiché dans le haut ou dans le bas de chaque page à l'écran jusqu'à ce que l'UC soit redémarrée. Cette exigence est facultative.
- 3.5.3.9 Le logiciel doit comprendre un menu pour copier la base de données entre l'unité de contrôle et une clé USB. Ce menu peut faire partie de l'ensemble des menus ou s'afficher uniquement lors de l'insertion d'une clé USB.

3.5.4 Rubriques d'affichage des états

- 3.5.4.1 L'unité de contrôle doit être capable d'afficher le fonctionnement opérationnel de l'UC en temps réel, incluant, sans s'y limiter, les phases en cours de service, les intervalles véhiculaires et piétonniers avec leurs minutages, les appels et rappels véhiculaires et piétonniers (incluant le mode des rappels), la coordination (état, source du contrôle, position dans le cycle local, position dans le cycle maître, décalage, etc.), la

préemption, la détection, les priorités pour autobus, etc. L'affichage doit répondre aux articles 3.4.7, 3.5.7, 3.6.5, 3.7.4 et 3.8.4 de NEMA TS 2.

- 3.5.4.2 Il doit être possible, à partir de la rubrique d'affichage des états opérationnels, de simuler des appels véhiculaires et piétonniers à partir du clavier frontal de l'UC.
- 3.5.4.3 L'unité de contrôle doit comporter une rubrique d'affichage de l'état du moniteur de conflit (CMU - Cabinet Monitoring Unit ou MMU - Malfunction Management Unit, selon que l'UC est installée respectivement dans un coffret ATC-5301 ou dans un coffret NEMA TS 2). Cette rubrique doit aussi permettre de visualiser l'état courant de chaque canal tel qu'observé par le moniteur de conflit. Il est souhaitable que la rubrique d'affichage des états puisse afficher la matrice de compatibilité programmée dans le moniteur de conflit.
- 3.5.4.4 L'unité de contrôle doit comporter une rubrique d'affichage de l'état des entrées (*Inputs*) et sorties (*Outputs*) de l'unité de contrôle.
- 3.5.4.5 L'unité de contrôle doit comporter une rubrique d'affichage des alertes et alarmes.
- 3.5.4.6 L'unité de contrôle doit comporter une rubrique d'affichage de l'état de l'unité de contrôle. Cette rubrique doit permettre de visualiser les informations matérielles et logicielles de l'unité de contrôle, incluant, sans s'y limiter, la version du logiciel et la version du système d'exploitation.

3.5.5 Rubriques d'aide

- 3.5.5.1 Le logiciel de contrôle doit comprendre une aide contextuelle pour chaque paramètre de programmation. La page d'aide doit être accessible directement à partir de la page de programmation, et l'opérateur doit pouvoir alterner facilement entre les deux. L'aide doit correspondre au paramètre du champ d'entrée actif et comprendre, au minimum, la plage des valeurs possibles ainsi que la description du paramètre et de son fonctionnement.

3.5.5.2 Le logiciel de contrôle doit comprendre une aide contextuelle pour chaque paramètre affiché dans une rubrique d'affichage des états. La page d'aide doit être accessible directement à partir de la page de la rubrique d'affichage des états, et l'opérateur doit pouvoir alterner facilement entre les deux. L'aide doit correspondre au paramètre du champ sélectionné et comprendre, au minimum, la légende des codes utilisés dans l'affichage de l'état du paramètre et la description du paramètre.

3.6 Serveur Web

3.6.1 L'unité de contrôle doit comporter un serveur Web intégré.

3.6.1.1 Il doit utiliser le protocole sécurisé de communication client-serveur HTTPS.

3.6.1.2 Il doit utiliser le protocole de sécurisation des échanges TLS version 1.2 ou plus récente.

3.6.1.3 La communication via les protocoles Telnet ou Secure Shell (SSH) n'est pas acceptée comme substitut au serveur Web intégré.

3.6.2 Le serveur Web doit générer des pages Web pouvant être interprétées par des navigateurs Web modernes, incluant, mais sans s'y limiter, Google Chrome, Microsoft Edge, Apple Safari et Mozilla Firefox.

3.6.3 Le serveur Web doit :

3.6.3.1 Donner accès à tous les menus et permettre de modifier tous les paramètres de l'UC;

3.6.3.2 Afficher et permettre d'éditer l'heure, la date, le fuseau horaire et l'heure avancée.

3.6.3.3 Reproduire l'affichage courant de l'écran de l'UC;

3.6.3.4 Le serveur Web peut afficher un clavier virtuel pouvant être activé à l'aide de la souris de l'ordinateur;

- 3.6.3.5 Le serveur Web peut accepter les entrées au clavier physique de l'ordinateur pour les touches équivalentes à celles du clavier de l'unité de contrôle (cette exigence est facultative mais fortement recommandée); la corrélation entre les touches du clavier physique de l'ordinateur et les touches du clavier de l'unité de contrôle doit être documentée.
- 3.6.3.6 Afficher graphiquement l'état des phases et chevauchements (véhiculaires et piétons), des entrées et des sorties, des détecteurs, des appels piétons, des demandes et des états des préemptions et priorités pour autobus, etc.
- 3.6.3.7 Permettre de télécharger tous les registres des événements;
- 3.6.3.8 Permettre de télécharger et de téléverser la base de données (format natif de l'unité de contrôle); le format de fichier doit être le même que lorsque le transfert est fait avec une clé USB; la mise en opération d'une nouvelle base de données peut nécessiter le redémarrage de l'unité de contrôle;
- 3.6.3.9 Permettre d'exporter les données dans un format documenté pouvant être interprété par un autre logiciel (csv, xml, yml, xls, xlsx, etc.). Le format du fichier et la documentation doivent permettre de déchiffrer les variables et leurs valeurs. Un format qui ne permet que l'affichage et l'impression (txt, html, pdf, etc.) de la base de données n'est pas accepté.

3.7 Scripts logiques

- 3.7.1 L'UC doit posséder la capacité d'exécuter des scripts (logique booléenne).
- 3.7.2 Les scripts doivent être exécutés à chaque itération de l'algorithme de contrôle (soit une fois à tous les dixièmes de seconde).
- 3.7.3 Les scripts sont une suite de commandes et de logiques booléennes, incluant, sans s'y limiter, les opérateurs IF, AND, OR, NOT, NAND, NOR, XOR, etc.

- 3.7.3.1 Il doit être possible de programmer au moins 64 opérations IF...THEN...ELSE au total sur l'ensemble des scripts avec chacune :
- 3.7.3.1.1 Au moins 10 conditions (avec opérateurs AND, OR, NAND, NOR, XOR, etc.) comportant de chaque côté une opération d'égalité ou d'inégalité;
- 3.7.3.1.2 Au moins 5 actions si la résultante des conditions est vraie;
- 3.7.3.1.3 Au moins 5 actions si la résultante des conditions est fausse; (ces actions résultantes peuvent être codées à même un opérateur IF...THEN...ELSE ou être codées avec un opérateur IF...THEN et un opérateur IF NOT...THEN en fonction de la résultante desdites conditions);
- 3.7.3.2 L'ensemble des opérations stipulées à l'article 3.7.3.1 doit avoir le temps de s'exécuter à chaque itération de l'algorithme de contrôle.
- 3.7.3.3 Au moins 20 millisecondes doivent être accordée à l'exécution des scripts à chaque itération de l'algorithme de contrôle.
- 3.7.3.4 Le format d'écriture des scripts n'est pas imposé. Les scripts doivent être écrits dans un langage lisible par l'humain. Le langage assembleur est proscrit. Les scripts peuvent être sous forme de programmes écrit en langage informatique ou sous forme de programmes composés d'une liste d'opérations dans le format « Résultat = Opérateur(Paramètre A, Paramètre B) » ou sous forme de programmes composés d'une opération IF...THEN...ELSE.
- 3.7.4 Il doit être possible d'utiliser des délais (*Delay*), des minuteries (*Timers*) et des variables (*Flags*) dans les scripts.
- 3.7.5 Les scripts doivent pouvoir utiliser l'état courante des entrées (*Inputs*) et des sorties (*Outputs*).
- 3.7.6 Les scripts doivent pouvoir modifier les données d'entrée (*Inputs*) et les données de sortie (*Outputs*) de l'unité de contrôle.

- 3.7.7 L’écriture des scripts ne doit pas nécessiter de logiciel externe autre qu’un éditeur de texte.
- 3.7.8 Si les scripts ne font pas partie intégrante de la base de données, ils doivent pouvoir être copiés (via une clé USB) de et vers l’unité de contrôle sans nécessiter de logiciel externe. Le fichier contenant les scripts exportés doit comprendre le code source des scripts (en langage lisible par l’humain).
- 3.7.9 Il doit être possible de programmer les scripts à même le clavier de l’unité de contrôle.
- 3.7.10 Il doit être possible de lire et d’afficher, à même l’écran de l’unité de contrôle, le code source des scripts programmés.
- 3.7.11 Un script logique ne doit jamais être utilisé si la désactivation de ce script peut causer un problème de sécurité, c’est-à-dire que la programmation (séquence des phases, minutages, etc.) doit être sécuritaire et conforme aux normes même si les scripts ne fonctionnent pas.
- 3.7.12 L’intégration de fonctions directement dans les paramètres, menus et rubriques du logiciel de contrôle doit être favorisée afin de minimiser l’utilisation des scripts.

3.8 **Registres des événements**

- 3.8.1 L’unité de contrôle doit maintenir un registre des événements comme les alertes, les fautes, les pertes de communication avec un équipement du coffret, les mises à jour du logiciel, les redémarrages de l’unité de contrôle, les changements d’heure et de date, etc.
- 3.8.2 L’unité de contrôle doit maintenir un registre des événements haute résolution (*hi-res logs*) afin de faciliter la résolution de problèmes.
 - 3.8.2.1 Le registre doit avoir une résolution de 100 millisecondes.
 - 3.8.2.2 Les événements enregistrés dans le registre doivent être fidèles aux événements réellement survenus durant l’opération de l’unité de contrôle.

- 3.8.2.3 Il doit être possible de choisir les catégories d'événements qui sont enregistrées au registre ou de filtrer les événements à afficher ou exporter.
- 3.8.2.4 Le registre doit être capable de couvrir la dernière période de 3 mois. Si la mémoire de l'unité de contrôle n'est pas suffisante pour emmagasiner les données sur la période spécifiée, une carte SD de capacité suffisante doit être installée.
- 3.8.2.5 L'unité de contrôle doit permettre la récupération par le système de gestion centralisée des événements du registre haute résolution à intervalle régulier soit via le protocole https ou via le protocole ftps.
- 3.8.2.6 Il doit être possible d'exporter les données du registre haute résolution selon le dictionnaire *Indiana Traffic Signal Hi Resolution Data Logger Enumerations*¹.
- 3.8.2.6.1 Le Fournisseur doit fournir le dictionnaire pour tous les codes additionnels utilisés par le registre haute résolution qui ne figurent pas dans ce dictionnaire ou si le registre haute résolution utilise un dictionnaire différent.
- 3.8.2.6.2 Si le registre haute résolution utilise un dictionnaire différent, le Fournisseur doit fournir un utilitaire pour convertir les données exportées du registre haute résolution vers le dictionnaire stipulé à l'article 3.8.2.5.
- 3.8.2.6.3 Les données exportées du registre haute résolution doivent être dans un format (tel que CSV) permettant leur analyse à l'aide d'un tableur (comme Microsoft Excel ou Google Sheets). Si le registre haute résolution est exporté dans un format différent, le Fournisseur doit fournir un utilitaire

¹ Li, H., A. M. Hainen, J. R. Sturdevant, T. Atkison, S. Talukder, J. K. Mathew, D. M. Bullock, D. Nelson, D. M. Maas, Jr., J. Fink, and T. Stiles. *Indiana Traffic Signal Hi Resolution Data Logger Enumerations*. Indiana Department of Transportation and Purdue University, West Lafayette, Indiana, 2019. <https://doi.org/10.5703/1288284316998>

pour convertir les données exportées du registre haute résolution vers le format stipulé au présent article.

3.9 Compatibilité NTCIP avec un logiciel de gestion centralisée

3.9.1 La première unité de contrôle doit être fournie avec une clé USB contenant les fichiers ASCII des modules MIB suivants en format ASN.1 :

3.9.1.1 Les modules MIB standards selon NTCIP pour l'unité de contrôle;

3.9.1.2 Les modules MIB spécifiques au fabricant (en spécifiant le format et la plage de valeurs acceptées dans le champ SYNTAX de la macro OBJECT-TYPE) pour tous les objets qui ne sont pas définis par NTCIP ou qui sont différents des objets définis par NTCIP.

3.9.2 La transmission des fichiers des modules MIB par le Fournisseur peut être accompagnée d'une entente de non-divulgation (NDA). Toute entente de non-divulgation doit comporter :

3.9.2.1 Une disposition permettant à la Ville de transmettre les fichiers des modules MIB à un fournisseur (actuel ou futur) d'une solution de gestion centralisée de la circulation.

3.9.2.2 Une disposition permettant à la Ville de transmettre les fichiers des modules MIB à un consultant engagé par la Ville pour l'intégration d'une solution de gestion centralisée de la circulation ou le développement de systèmes reliés à ce dernier.

3.9.3 Le Fournisseur doit transmettre les fichiers des modules MIB à jour avec toute nouvelle version du logiciel de contrôle.

3.9.4 L'unité de contrôle doit être supportée par le système de gestion centralisée des feux de circulation Kinetic Signals de QFree (version en fonction au moment du présent appel d'offres). Le Fournisseur doit déposer une lettre de conformité de QFree à cet effet.

3.9.4.1 Toutes les fonctions de l'unité de contrôle doivent être supportées par le logiciel de gestion centralisée.

- 3.9.4.2 Au moins une version stable du logiciel de contrôle doit être déjà supportée par le logiciel de gestion centralisée.
- 3.9.4.3 Si l'intégration des modules MIB de la plus récente version du logiciel de contrôle n'est pas complétée dans le système de gestion centralisée au moment de la fermeture de l'appel d'offres, la lettre de conformité produite par le fournisseur du logiciel de gestion centralisée doit :
 - 3.9.4.3.1 Confirmer que ce dernier a obtenu les modules MIB du Fournisseur de l'unité de contrôle ;
 - 3.9.4.3.2 Comprendre l'engagement du fournisseur du logiciel de gestion centralisée à intégrer les modules MIB de la version correspondante du logiciel de contrôle ;
 - 3.9.4.3.3 Indiquer la version antérieure du logiciel de contrôle déjà supportée par le logiciel de gestion centralisée;
 - 3.9.4.3.4 Préciser le délai pour la disponibilité de la mise à jour du logiciel de gestion centralisée intégrant les modules MIB de la version la plus récente du logiciel de contrôle.

4. Compatibilité avec les coffrets de contrôle

- 4.1 L'unité de contrôle doit être compatible avec une interface simulant un coffret de contrôle ATC 5301 et un coffret de contrôle NEMA TS 2, tel que « The Virtual Cabinet » modèle TVC-3800 de ATSI.
- 4.2 L'unité de contrôle doit être compatible avec les coffrets de contrôle NEMA TS 2 de plusieurs manufacturiers.
- 4.3 L'unité de contrôle doit être compatible avec les coffrets de contrôle ATC 5301 de plusieurs manufacturiers. À la demande de la Ville, le Fournisseur doit en faire la démonstration.

5. Identification

- 5.1 Le boîtier de l'unité de contrôle doit comporter une étiquette permanente indiquant les informations suivantes :

- 5.1.1 Marque de commerce du fabricant;
- 5.1.2 Numéro de modèle et sa version;
- 5.1.3 Numéro de série;
- 5.1.4 Date de fabrication.

6. Documentation

- 6.1 Une copie imprimée des manuels n'est pas requise.
- 6.2 Une version numérique (en format PDF) des manuels d'installation, de programmation, d'opération et d'entretien doit être transmise au plus tard avant la première des échéances suivantes :
 - 6.2.1 Deux (2) semaines avant une formation donnée aux employés de la Ville;
 - 6.2.2 Quatre (4) semaines avant la réalisation d'essais de conformité;
 - 6.2.3 Au moment de la livraison d'unités de contrôle.
- 6.3 Une version numérique (en format PDF) du manuel de programmation doit être transmise (sans frais supplémentaires) avec chaque nouvelle version du logiciel.
- 6.4 La documentation du format de fichier utilisé lors de l'exportation de la base de données.
- 6.5 La documentation des dictionnaires, notamment pour l'interprétation des événements à haute résolution (*hi-res logs*).
- 6.6 La documentation associée aux modules MIB.
- 6.7 Les rapports et attestations démontrant la conformité de l'unité de contrôle (vérifications et essais de conformité et contrôle de la qualité).

7. Vérifications et essais de conformité

- 7.1 Tous les essais applicables stipulés aux normes ATC-5201 et NEMA TS 2 doivent être réalisées par le manufacturier. Le Fournisseur doit transmettre les attestations et rapports du manufacturier suivants :

- 7.1.1 Une attestation de conformité du modèle de l'unité de contrôle aux essais d'acceptation de la conception (*Design Acceptance Testing*) réalisés par le manufacturier sur la première unité de production (*First Article Unit*);
- 7.1.2 Le rapport démontrant la conformité de chaque unité de contrôle aux essais de production et aux processus de contrôle de la qualité du manufacturier, conformément aux exigences de la norme ATC-5201.
- 7.2 Le Fournisseur doit réaliser des essais afin de démontrer la conformité de l'unité de contrôle au protocole de communication NTCIP. Les essais doivent comprendre la vérification des objets selon leur définition dans les modules MIB et dans les normes NTCIP. Le Fournisseur doit soumettre un rapport attestant de la conformité de l'unité de contrôle aux normes NTCIP 1201 et 1202 ainsi que la concordance avec les modules MIB du manufacturier.
- 7.3 Le Fournisseur doit réaliser des essais afin de démontrer la capacité du logiciel de l'unité de contrôle à exécuter les fonctions stipulées au présent devis et aux normes NEMA TS 2 et NTCIP. Le Fournisseur doit soumettre un rapport détaillé attestant de la conformité de l'unité de contrôle pour chaque fonction stipulée. Pour chaque fonction qui n'est pas directement supportée par le logiciel de contrôle, le rapport doit indiquer la méthode proposée pour exécuter une fonction équivalente (notamment, mais sans s'y limiter, à l'aide d'un script logique).
- 7.4 Après que la documentation a été revue et commentée par la Ville, le Fournisseur doit livrer une première unité de contrôle. La Ville se réserve le droit de vérifier la conformité de la première unité de contrôle avant d'accepter livraison des autres unités de contrôle. Advenant que l'unité de contrôle ne soit pas conforme aux exigences du présent devis, la Ville se réserve le droit d'annuler les commandes non livrées ou de reporter la livraison des unités non encore livrées.

8. Livraison

- 8.1 Toutes les unités de contrôle, incluant le Prototype, doivent être livrées aux ateliers du Service des immeubles, parcs et espaces publics (feux de circulation et éclairage) de la Ville à l'adresse suivante : 1489, rue Cunard (porte 1), Laval (Qc), H7S 2H8.

- 8.2 Tous les frais de transport, de chargement et de déchargement sont à la charge du Fournisseur.
- 8.3 Toutes les composantes enfichables de l'unité de contrôle (incluant, sans s'y limiter, le clavier s'il est amovible et toutes les cartes d'extension et de communication enfichables) doivent être installées et solidement fixées à l'unité de contrôle afin qu'elles ne se détachent pas durant le transport et la manutention.
- 8.4 Toutes les composantes amovibles externes à l'unité de contrôle (notamment l'adaptateur pour le connecteur MS-A) doivent être emballées adéquatement et livrées séparément des unités de contrôle afin d'en faciliter l'inventaire au moment de la livraison.
- 8.5 Les unités de contrôle doivent être emballées dans des boîtes en carton, et l'écran doit être recouvert par une pellicule permettant de le protéger contre les égratignures durant le transport et la manutention.

9. Service après-vente

9.1 Assistance technique

- 9.1.1 Le Fournisseur doit accorder son entière collaboration et offrir une assistance technique en français par du personnel technique qualifié durant la période de garantie, et ce, sans frais supplémentaires. Au minimum, l'assistance technique doit être offerte par téléphone et par courriel.
- 9.1.2 En cas de problème durant l'installation ou durant l'opération de l'unité de contrôle, la Ville se charge de réaliser un premier diagnostic afin de déterminer la nature générale du problème. Le diagnostic réalisé par la Ville ne vise pas à déterminer la source exacte du problème, mais uniquement à déterminer si le problème est couvert par la garantie. Si le problème est lié à un défaut couvert par la garantie :
 - 9.1.2.1 La Ville peut transmettre au Fournisseur la base de données de l'unité de contrôle ainsi que les informations et les journaux d'événements recueillis.
 - 9.1.2.2 La Ville peut retourner au Fournisseur les composantes défectueuses pour réparation ou remplacement.

9.1.2.3 Dans tous les cas, le Fournisseur doit :

- 9.1.2.3.1 Procéder au dépannage (*troubleshooting*) et au débogage afin d'établir le diagnostic détaillé, déterminer les causes du problème et identifier les solutions;
- 9.1.2.3.2 Réparer ou remplacer les composantes défectueuses à l'intérieur d'un délai maximal de trois (3) mois;
- 9.1.2.3.3 Proposer une solution temporaire pour contourner les bogues identifiés, et ce, à l'intérieur d'un délai maximal de trois (3) mois;
- 9.1.2.3.4 S'assurer de la correction des bogues par le manufacturier dans un délai maximal d'un (1) an par une mise à jour stable des composantes logicielles. La mise à jour doit maintenir toutes les fonctionnalités et ne doit pas présenter de bogue majeur qui pourrait empêcher son déploiement.

9.1.3 Le Fournisseur doit faciliter le lien entre la Ville et le manufacturier.

9.1.4 Le manufacturier doit :

- 9.1.4.1 Offrir un accès direct pour la Ville au support technique du manufacturier sans nécessiter la permission du Fournisseur.
- 9.1.4.2 Avoir un système de billets pour le suivi de la correction des bogues.
 - 9.1.4.2.1 Le système de billets doit permettre à la Ville d'ouvrir de nouveaux billets.
 - 9.1.4.2.2 Le système de billets doit permettre à la Ville de consulter les billets qu'elle a fait ouvrir et de suivre leur état d'avancement.
 - 9.1.4.2.3 L'ouverture et la consultation d'un billet peut, au choix du manufacturier, être manuelle (via un courriel ou un appel

téléphonique) ou être automatisée (via le site Web du manufacturier).

9.1.4.2.4 Le délai pour ouvrir un nouveau billet ou pour consulter l'état d'un billet déjà ouvert doit être inférieur à 72 heures.

9.1.4.2.5 L'état d'avancement de chaque billet doit permettre un suivi de l'avancement de la recherche de solution et de connaître, au minimum, les informations suivantes lorsque chaque étape est franchie :

9.1.4.2.5.1 Étape 1 : La date à laquelle le problème a été reproduit en atelier par le manufacturier et la procédure pour reproduire le problème de manière prévisible (incluant, sans s'y limiter, les paramètres, les conditions et la séquence des événements et des actions).

9.1.4.2.5.2 Étape 2 : La date à laquelle la source du bogue a été trouvée, la description de la source du problème et la liste des conditions et des événements dans lesquelles le problème peut se présenter.

9.1.4.2.5.3 Étape 3 : La version du logiciel qui intégrera les correctifs nécessaires.

9.1.4.2.5.4 Étape 4 : L'échéancier associé à la version du logiciel qui corrigera le problème identifié.

9.1.5 La Ville se réserve le droit de bloquer tout nouvel achat (lié au même appel d'offres ou à tout autre appel d'offres) d'unités de contrôle du même manufacturier tant que les défauts ou bogues majeurs ne sont pas corrigés. Un défaut ou un bogue est considéré comme majeur s'il empêche le bon et complet fonctionnement de l'unité de contrôle et qu'aucune solution temporaire ne permet de le contourner ou si le billet associé à ce bogue est ouvert depuis plus d'un an ou si le défaut ou le bogue pose un problème pour la sécurité des usagers.

9.2 Garantie

- 9.2.1 L'unité de contrôle doit être garantie par le manufacturier contre tout défaut de fabrication ou d'assemblage pour une période de 5 ans à partir de la date de livraison de chaque unité de contrôle.
- 9.2.1.1 Lorsque les unités de contrôle sont fournies directement à la Ville dans le cadre d'un appel d'offres d'achat de matériaux, la date de livraison est déterminée par la date de réception de chaque unité de contrôle aux ateliers de la Ville.
- 9.2.1.2 Lorsque les unités de contrôle sont fournies via un entrepreneur dans le cadre d'un appel d'offres de construction, la date de livraison est déterminée par la date de mise en opération de chaque système de feux de circulation.
- 9.2.2 La garantie couvre aussi tout bogue logiciel dont le problème est survenu pendant la période de garantie. La résolution sous garantie de ces bogues doit résulter en une version stable des composantes logicielles. La mise à jour des composantes logicielles doit être couverte par une garantie d'au moins un (1) an ou jusqu'à la fin de la garantie initiale (selon la dernière de ces éventualités à survenir). Tout nouveau bogue découvert durant cette période de garantie doit être corrigé par une nouvelle mise à jour selon les mêmes conditions.
- 9.2.3 Les garanties couvrent toutes les réparations nécessaires pour corriger les défauts de fabrication et d'assemblage. Toute unité de contrôle présentant une défectuosité durant la période de garantie et qui ne peut être réparée doit être remplacée sans frais supplémentaires. Toute unité de contrôle présentant un problème ayant fait l'objet de deux (2) réparations préalables doit être remplacée sans frais supplémentaires.
- 9.2.4 Toute composante réparée ou remplacée sous garantie doit être couverte par une garantie d'au moins un (1) an à partir de la date à laquelle la composante (nouvelle ou réparée) est réinstallée ou jusqu'à la fin de la garantie initiale de l'unité de contrôle, selon la dernière de ces éventualités à survenir.
- 9.2.5 Chaque unité de contrôle doit être identifiée d'un numéro de série unique.

- 9.2.6 Le Fournisseur doit produire un certificat de garantie pour chaque unité de contrôle.
- 9.2.7 Le certificat de garantie doit inclure, sans s'y limiter, les informations suivantes :
- 9.2.7.1 Les numéros d'identification de la Ville (numéros de projet, d'appel d'offres, de contrat, etc.);
- 9.2.7.2 Le nom et l'adresse de l'entreprise qui a émis le certificat de garantie;
- 9.2.7.3 La description de la garantie;
- 9.2.7.4 Le numéro de série de l'unité de contrôle;
- 9.2.7.5 La date de livraison de l'unité de contrôle;
- 9.2.7.6 La date d'expiration de la garantie.
- 9.2.8 La Ville prend à sa charge les frais de transport pour le retour des pièces défectueuses au Fournisseur. Tous les autres frais de transport pour le service sous garantie des composantes défectueuses sont à la charge du Fournisseur, incluant, sans s'y limiter, pour les échanges auprès du manufacturier.

9.3 Pièces

9.3.1 Délai de réparation

- 9.3.1.1 Le Fournisseur doit garantir que les réparations pourront être complétées et que les pièces de rechange (incluant des unités de contrôle entières) pourront être fournies dans un délai maximum de trois (3) mois, et ce, peu importe les contraintes de délai d'approvisionnement auprès du manufacturier.
- 9.3.1.2 Si le délai de réparation n'est pas respecté, le Fournisseur doit prêter une unité de contrôle de rechange équivalente temporaire avant l'expiration du délai de réparation et jusqu'à la réparation ou au remplacement de la pièce défectueuse.

9.3.1.3 Si le Fournisseur est incapable de réparer une unité de contrôle défectueuse à l'intérieur d'un délai d'un (1) an, il doit la remplacer avant l'échéance dudit délai par une unité de contrôle neuve.

9.4 Mises à jour et mises à niveau

- 9.4.1 Le manufacturier doit avoir un processus de développement actif et continu des composantes logicielles, incluant (sans s'y limiter) le logiciel de contrôle, le système d'exploitation, les clés d'encryptage et les certificats de sécurité, les pilotes, les microgiciels, les modules MIB et les utilitaires externes, afin de produire des mises à jour pour la correction de bogues et produire des mises à niveau afin de suivre l'évolution des normes, particulièrement (mais sans s'y limiter) les normes NTCIP.
- 9.4.2 Toutes les mises à jour des composantes logicielles doivent être transmises à la Ville sans frais supplémentaires. Le manufacturier doit supporter les composantes logicielles au minimum pour la durée de la garantie. Si le manufacturier cesse de publier des mises à jour sur la version principale des composantes logicielles avant la fin de la période de garantie (et que la dernière mise à jour disponible n'est pas stable ou ne corrige pas les problèmes identifiés), toutes les mises à niveau vers une version principale plus récente ou vers un nouveau logiciel doivent être transmises à la Ville sans frais supplémentaires, et ce, jusqu'à la fin de la période de garantie.
- 9.4.2.1 La licence d'utilisation des mises à jour doit couvrir toutes les unités de contrôle sous la juridiction de la Ville qui sont compatibles avec la mise à jour.
- 9.4.2.2 La licence d'utilisation des mises à niveau (fournies après la fin du support de la version principale en vertu de l'article 9.4.2 du présent devis) doit couvrir toutes les unités de contrôle sous la juridiction de la Ville qui sont compatibles avec la mise à niveau et qui sont encore sous garantie.

- 9.4.3 Le Fournisseur doit aviser la Ville de toutes les nouvelles mises à jour et mises à niveau des composantes logicielles au plus tard 30 jours après qu'elles sont rendues disponibles par le manufacturier soit par courriel, soit en inscrivant la Ville dans la liste de distribution des mises à jour et des mises à niveau du manufacturier, soit en diffusant ces informations directement sur le site web du manufacturier.
- 9.4.4 Les mises à jour et les mises à niveau doivent être accompagnées :
- 9.4.4.1 D'un journal des modifications, des notes techniques ou des notes de mise à jour (*release notes*) précisant les bogues corrigés et les changements apportés par cette version du logiciel;
 - 9.4.4.2 De toute la documentation, incluant la documentation des nouvelles fonctions, sous forme d'addenda aux manuels pour les mises à jour et de manuels révisés pour les mises à niveau;
 - 9.4.4.3 Des utilitaires, notamment les dictionnaires et outils de traduction des registres des événements à haute résolution (*hi-res logs*);
 - 9.4.4.4 Des modules MIB correspondants;
- 9.4.5 Il doit toujours être possible d'installer ultérieurement une version antérieure du logiciel de contrôle.