NORME INTERNATIONALE

ISO 19014-4

Première édition
2020-07

Engins de terrassement — Sécurité fonctionnelle —

Partie 4: Conception et évaluation du logiciel et de la transmission des données pour les parties relatives à la sécurité du système de commande

Earth-moving machinery — Functional safety —

Part 4: Design and evaluation of software and data transmission for safety-related parts of the control system

Numéro de référence
ISO 19014-4:2020(F)

ISO copyright office
Case postale 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Genève
Tél.: +41 22 749 01 11
E-mail: copyright@iso.org
Web: www.iso.org

Publié en Suisse

Sommaire

Page

Avant-propos ..... **iv**

Introduction ..... **v**

1 Domaine d'application **1**

2 Références normatives **1**

3 Termes et définitions **1**

4 Développement de logiciel **4**
4.1 Généralités 4
4.2 Planification 5
4.3 Artefacts 6
4.4 Spécification des exigences relatives à la sécurité du logiciel 7
4.5 Conception de l'architecture du logiciel 8
4.6 Conception et codage des modules logiciels 8
4.7 Choix du langage et des outils 9
4.8 Essais des modules logiciels 10
4.9 Intégration et essais des modules logiciels 11
4.10 Validation du logiciel 12

5 Paramétrage fondé sur le logiciel **13**
5.1 Généralités 13
5.2 Intégrité des données 13
5.3 Vérification du paramétrage fondé sur le logiciel 13

6 Protection de la transmission de messages relatifs à la sécurité sur les systèmes bus **14**

7 Indépendance par partitionnement du logiciel **15**
7.1 Généralités 15
7.2 Plusieurs partitions dans un microcontrôleur unique 16
7.3 Plusieurs partitions dans le domaine d'application d'un réseau d'UCE 17

8 Informations pour l'utilisation **18**
8.1 Généralités 18
8.2 Notice d'instructions 18

Annexe A (informative) **Description des méthodes/mesures du logiciel** **19**

Annexe B (normative) **Environnements d'essais de validation d'un logiciel** **33**

Annexe C (informative) **Calcul de l'assurance d'intégrité des données** **36**

Annexe D (informative) **Méthodes et mesures de protection de la transmission** **38**

Annexe E (informative) **Méthodes et mesures de protection des données internes au microcontrôleur** **40**

Bibliographie **42**

Avant-propos

L'ISO (Organisation internationale de normalisation) est une fédération mondiale d'organismes nationaux de normalisation (comités membres de l'ISO). L'élaboration des Normes internationales est en général confiée aux comités techniques de l'ISO. Chaque comité membre intéressé par une étude a le droit de faire partie du comité technique créé à cet effet. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec l'ISO participent également aux travaux. L'ISO collabore étroitement avec la Commission électrotechnique internationale (IEC) en ce qui concerne la normalisation électrotechnique.

Les procédures utilisées pour élaborer le présent document et celles destinées à sa mise à jour sont décrites dans les Directives ISO/IEC, Partie 1. Il convient, en particulier, de prendre note des différents critères d'approbation requis pour les différents types de documents ISO. Le présent document a été rédigé conformément aux règles de rédaction données dans les Directives ISO/IEC, Partie 2 (voir www.iso.org/directives).

L'attention est attirée sur le fait que certains des éléments du présent document peuvent faire l'objet de droits de propriété intellectuelle ou de droits analogues. L'ISO ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de propriété et averti de leur existence. Les détails concernant les références aux droits de propriété intellectuelle ou autres droits analogues identifiés lors de l'élaboration du document sont indiqués dans l'Introduction et/ou dans la liste des déclarations de brevets reçues par l'ISO (voir www.iso.org/brevets).

Les appellations commerciales éventuellement mentionnées dans le présent document sont données pour information, par souci de commodité, à l'intention des utilisateurs et ne sauraient constituer un engagement.

Pour une explication de la nature volontaire des normes, la signification des termes et expressions spécifiques de l'ISO liés à l'évaluation de la conformité, ou pour toute information au sujet de l'adhésion de l'ISO aux principes de l'Organisation mondiale du commerce (OMC) concernant les obstacles techniques au commerce (OTC), voir www.iso.org/avant-propos.

Le présent document a été élaboré par le comité technique ISO/TC 127 *Engins de terrassement*, sous-comité SC 2 *Sécurité, ergonomie et exigences générales* en collaboration avec le Comité européen de Normalisation (CEN) Comité Technique CEN/TC 151, *Machines de génie civil et de production de matériaux de construction – Sécurité*, selon avec l'Accord de coopération entre l'ISO et le CEN (Accord de Vienne).

Cette première édition de l'ISO 19014-4, conjointement avec les autres parties de la série ISO 19014, annule et remplace l'ISO 15998:2008 et l'ISO/TS 15998-2:2012 qui ont fait l'objet d'une révision technique.

Les principales modifications par rapport à l'édition précédente sont les suivantes:

— les exigences supplémentaires pour le développement de logiciel,

— les exigences pour le développement du paramétrage fondé sur le logiciel,

— les exigences pour la transmission de messages relatifs à la sécurité sur un bus de communication et

— les exigences pour la validation du logiciel et la vérification des niveaux de performance de la machine.

Une liste de toutes les parties de la série ISO 19014 peut être trouvée sur le site internet de l'ISO.

Il convient que l'utilisateur adresse tout retour d'information ou toute question concernant le présent document à l'organisme national de normalisation de son pays. Une liste exhaustive desdits organismes se trouve à l'adresse www.iso.org/members.html.

Introduction

Le présent document établit des recommandations pour les systèmes combinés de composants électriques, électroniques et électroniques programmables [systèmes électriques/électroniques/électroniques programmables (E/E/PES)] qui sont utilisés pour la sécurité fonctionnelle dans les engins de terrassement.

La structure des normes de sécurité dans le domaine des machines est la suivante.

Les normes de type A (normes fondamentales de sécurité), contiennent des notions fondamentales, des principes de conception et des aspects généraux relatifs aux machines.

Les normes de type B (normes génériques de sécurité) traitent d'un ou de plusieurs aspects de la sécurité ou d'un ou de plusieurs types de moyens de protection valables pour une large gamme de machines:

— normes de type B1, traitant d'aspects particuliers de la sécurité (par exemple distances de sécurité, température superficielle, bruit);

— normes de type B2, traitant de moyens de protection (par exemple commandes bimanuelles, dispositifs de verrouillage, dispositifs sensibles à la pression, protecteurs).

Les normes de type C (normes de sécurité par catégorie de machines) traitent des exigences de sécurité détaillées s'appliquant à une machine particulière ou à un groupe de machines particulier.

Le présent document est une norme de type C telle que définie dans l'ISO 12100.

Le présent document est notamment pertinent pour les groupes de parties prenantes suivants représentant les acteurs du marché pour ce qui concerne la sécurité des machines:

— fabricants de machines (petites, moyennes et grandes entreprises);

— organismes de santé et de sécurité (autorités réglementaires, organismes de prévention des risques professionnels, surveillance du marché, etc.)

D'autres peuvent être affectés par le niveau de sécurité des machines obtenu au moyen du document par les groupes de parties prenantes mentionnés ci-dessus:

— utilisateurs de machines/employeurs (petites, moyennes et grandes entreprises);

— utilisateurs de machines/salariés (par exemple syndicats de salariés, organisations représentant des personnes ayant des besoins particuliers);

— prestataires de services, par exemple sociétés de maintenance (petites, moyennes et grandes entreprises);

Les groupes de parties prenantes mentionnés ci-dessus ont eu la possibilité de participer à l'élaboration du présent document.

Les machines concernées et l'étendue des phénomènes dangereux, situations dangereuses ou événements dangereux couverts sont indiquées dans le Domaine d'application du présent document.

Lorsque des exigences de la présente norme de type C sont différentes de celles énoncées dans les normes de type A ou les normes de type B, les exigences de la présente norme de type C ont priorité sur celles des autres normes pour les machines ayant été conçues et fabriquées conformément aux exigences de la présente norme de type C.

Engins de terrassement — Sécurité fonctionnelle —

Partie 4: Conception et évaluation du logiciel et de la transmission des données pour les parties relatives à la sécurité du système de commande

1 Domaine d'application

Le présent document spécifie les principes généraux applicables aux exigences en matière de développement de logiciel et de transmission des signaux des parties relatives à la sécurité des systèmes de commande de la machine (MCS) dans les engins de terrassement et leur équipement tels que définis dans l'ISO 6165. De plus, le présent document traite des phénomènes dangereux significatifs tels que définis dans l'ISO 12100 en rapport avec les logiciels intégrés dans le système de commande de la machine. Les phénomènes dangereux significatifs traités sont les réponses incorrectes du système de commande de la machine aux entrées du système de commande de la machine.

La cybersécurité n'est pas couverte par le présent document.

NOTE Voir une norme appropriée relative à la sécurité pour des recommandations à propos de la cybersécurité.

Le présent document n'est pas applicable aux engins de terrassement fabriqués avant la date de sa publication.

2 Références normatives

Les documents suivants sont cités dans le texte de sorte qu'ils constituent, pour tout ou partie de leur contenu, des exigences du présent document. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

ISO 6750-1, *Engins de terrassement — Manuel de l'opérateur — Partie 1: Présentation et contenu*

ISO 12100:2010, *Sécurité des machines — Principes généraux de conception — Appréciation du risque et réduction du risque*

ISO 13849-1, *Sécurité des machines — Parties des systèmes de commande relatives à la sécurité — Partie 1: Principes généraux de conception*

ISO 19014-1, *Engins de terrassement — Sécurité fonctionnelle — Partie 1: Méthodologie pour la détermination des parties relatives à la sécurité des systèmes de commande et les exigences de performance*

ISO 19014-2:—,[1)] *Engins de terrassement — Sécurité fonctionnelle — Partie 2: Conception et évaluation des exigences de matériel et d'architecture pour les parties relatives à la sécurité du système de commande*

3 Termes et définitions

Pour les besoins du présent document, les termes et les définitions de l'ISO 12100, ISO 19014-1, l'ISO 13849-1 ainsi que les suivants s'appliquent.

1) En préparation. Stade au moment de la publication: ISO/DIS 19014-2:2020.

L'ISO et l'IEC tiennent à jour des bases de données terminologiques destinées à être utilisées en normalisation, consultables aux adresses suivantes:

— ISO Online browsing platform: disponible à l'adresse https://www.iso.org/obp

— IEC Electropedia: disponible à l'adresse http://www.electropedia.org/

3.1
système bus
sous-système utilisé dans un système de commande électronique pour la transmission de *messages* (3.6)

Note 1 à l'article: Le système bus se compose de l'unité système (sources et récepteurs d'information), d'un trajet de transmission/support de transmission (par exemple des lignes électriques, des lignes en fibres optiques, transmission par radio fréquence) et de l'interface entre la source/le récepteur de message et l'électronique de bus (par exemple, circuit intégré à application spécifique de protocole, émetteurs-récepteurs).

3.2
système bus encapsulé
système bus (3.1) comprenant un nombre fixe ou un nombre maximal prédéterminé des nœuds du bus qui sont connectés l'un à l'autre par un support de transmission ayant des performances/caractéristiques bien définies et fixes

3.3
défaillance de l'homologue de communication
situation dans laquelle l'homologue de communication n'est pas disponible

3.4
répétition de message non prévue
situation dans laquelle le même *message* (3.6) est renvoyé de manière accidentelle

3.5
répétition de message
situation dans laquelle le même *message* (3.6) est renvoyé intentionnellement

Note 1 à l'article: Cette technique de renvoi du même message permet de remédier à des défaillances telles que la *perte du message* (3.10).

3.6
message
transmission électronique de données

Note 1 à l'article: Les données transmises peuvent comprendre des données d'utilisateur, une adresse ou des données d'identificateur et des données pour assurer l'intégrité de la transmission

3.7
ECU
unité de commande électronique
dispositif électronique (dispositif de commande électronique programmable) utilisé dans un système de commande sur des engins de terrassement

[SOURCE: ISO 22448:2010, 3.3, modifiée — Les termes admis «ECM» et "module de commande électronique" ont été supprimés.]

3.8
temps de réaction
délai entre la détection d'un événement relatif à la sécurité et l'initiation d'une réaction de sécurité

3.9
artefact
produits du travail qui sont produits et utilisés au cours d'un projet pour capturer et transmettre des informations

3.10
perte de message
suppression imprévue d'un *message* (3.6) en raison d'un défaut d'un nœud du bus

3.11
séquence incorrecte
modification imprévue d'une séquence de *message* (3.6) en raison d'une défaillance d'un nœud du bus

Note 1 à l'article: Les *systèmes bus* (3.1) peuvent contenir des éléments avec des messages stockés (premier entré, premier sorti (FIFO), etc.) qui peuvent modifier la séquence correcte.

3.12
déformation de message
modification imprévue d'un *message* (3.6) en raison d'une erreur d'un nœud du bus ou en raison d'erreurs sur le canal de transmission

3.13
retard de message
délai imprévu ou prévention de la fonction de sécurité, causés par une surcharge du trajet de transmission par l'échange normal de données ou l'envoi de *messages* (3.6) incorrects

3.14
compteur de maintien en activité
composant de comptage initialisé à «0» lorsque l'objet à surveiller est créé ou restauré

Note 1 à l'article: Le compteur incrémente du temps t-1 au temps t tant que l'objet est en activité. Finalement, le compteur de maintien en activité indique la durée pendant laquelle l'objet a été en activité au sein d'un réseau.

3.15
essai à la boîte noire
essai d'un objet qui n'exige pas de connaître sa structure interne ou sa mise en œuvre concrète

3.16
partition
entité de ressource attribuant une portion de la mémoire, des dispositifs d'entrée/sortie et de l'utilisation d'une unité centrale à une ou plusieurs *tâches du système* (3.21)

Note 1 à l'article: Les partitions peuvent être assignées à un ou plusieurs sous-systèmes du réseau de microcontrôleurs.

3.17
partitionnement de logiciel
méthode de confinement d'un *défaut logiciel* (3.26) qui consiste à assigner des ressources à des composants de logiciel spécifiques dans le but d'éviter la propagation du défaut logiciel à plusieurs *partitions* (3.16)

3.18
composant de logiciel
un ou plusieurs *modules logiciels* (3.19)

[SOURCE: ISO 26262-1:2018, 3.157, modifié — Le terme «unités» a été remplacé par «modules».]

3.19
module logiciel
partie indépendante d'un logiciel qui peut être soumise à l'essai de manière indépendante et suivie en fonction d'une spécification

Note 1 à l'article: Le module logiciel est un composant de logiciel indivisible.

3.20
partitions de logiciel
environnement d'exécution auquel sont assignées des ressources système distinctes

3.21
tâche système
entités d'exécution qui sont exécutées dans le cadre du budget de ressources des *partitions* (3.16) et avec des priorités différentes

3.22
indépendance de logiciel
exclusion des interactions non prévues entre les composants de logiciel, ainsi qu'absence d'impact sur le fonctionnement correct d'un composant de logiciel résultant d'erreurs d'un autre composant de logiciel

3.23
historique de fonctionnement
données relatives au fonctionnement d'un composant ou d'un *module logiciel* (3.19) pendant sa durée de service

3.24
temps de cycle maximal
temps statique pour accéder à un bus de communication entre nœuds au niveau d'un bus ou d'un nœud

Note 1 à l'article: L'application d'un protocole «TTP» (time-triggered protocol ou à déclenchement temporel) permet de s'assurer que ce temps de cycle n'est pas dépassé.

3.25
temps de réponse maximal
temps fixe assigné à une activité système pour échanger des *messages* (3.6) synchronisés globalement sur un bus dans une architecture de type «à déclenchement temporel»

3.26
défaut logiciel
étape, processus ou définition de données incorrect(e) dans un logiciel qui amène le système à produire des résultats inattendus

3.27
analyse d'impact
documentation qui contient des renseignements sur la signification et les répercussions d'une modification proposée

3.28
processus de gestion de la configuration
tâche qui consiste à suivre et à contrôler les changements apportés aux *artefacts* (3.9) dans le processus de développement

3.29
transmission constante de messages
situation dans laquelle le nœud défectueux transmet continuellement des *messages* (3.6) qui compromettent le fonctionnement du bus

3.30
blocage d'accès au bus de données
situation dans laquelle le nœud défectueux ne respecte pas les schémas d'utilisation prévus et engendre un trop grand nombre de demandes de service, réduisant ainsi sa disponibilité pour d'autres nœuds

4 Développement de logiciel

4.1 Généralités

Le principal objectif des exigences détaillées ci-dessous est d'obtenir un logiciel fiable qui soit lisible, compréhensible, vérifiable et maintenable. Le présent article donne des recommandations relatives à la conception d'un logiciel et les essais qui en découlent. La prévention des défauts logiciels doit être prise en compte tout au long du processus de développement du logiciel.

Lorsqu'un composant de logiciel existant a été développé conformément à une norme antérieure et qu'il a été démontré par l'utilisation et la validation qu'il réduit le risque à un niveau aussi bas que raisonnablement possible, il ne doit y avoir aucune exigence de mettre à jour la documentation du cycle de vie du logiciel au niveau du module logiciel.

Les logiciels de commande des machines doivent être conformes aux exigences de sécurité du présent article. De plus, les logiciels de commande de la machine doivent être conçus et développés conformément aux principes de l'ISO 12100:2010 pour les phénomènes dangereux pertinents mais non significatifs qui ne sont pas traités par le présent document.

4.2 Planification

Un planning doit être établi pour définir la relation entre les phases individuelles de développement du logiciel et les artefacts connexes.

Les méthodes et mesures appropriées du Tableau 3 au Tableau 9 doivent être choisies pour le développement du logiciel conformément au niveau de performance requis de la machine (MPLr).

Le MPLr du système peut être atteint en ajoutant, en parallèle, deux systèmes d'un niveau de performance inférieur. Lors de la mise en parallèle (selon l'ISO 19014-2), le logiciel peut être développé dans chaque système pour répondre aux exigences du MPLr inférieur. Cela n'est permis que lorsqu'il n'existe pas de défaillances de cause commune entre les deux systèmes.

L'adéquation des méthodes ou des mesures choisies à l'application doit être justifiée et doit être effectuée au début de chaque phase de développement prévue. Pour une application particulière, la combinaison appropriée des méthodes ou des mesures doit être spécifiée pendant la planification du développement. Il est permis d'utiliser des méthodes ou des mesures qui ne sont pas énumérées du Tableau 3 au Tableau 9.

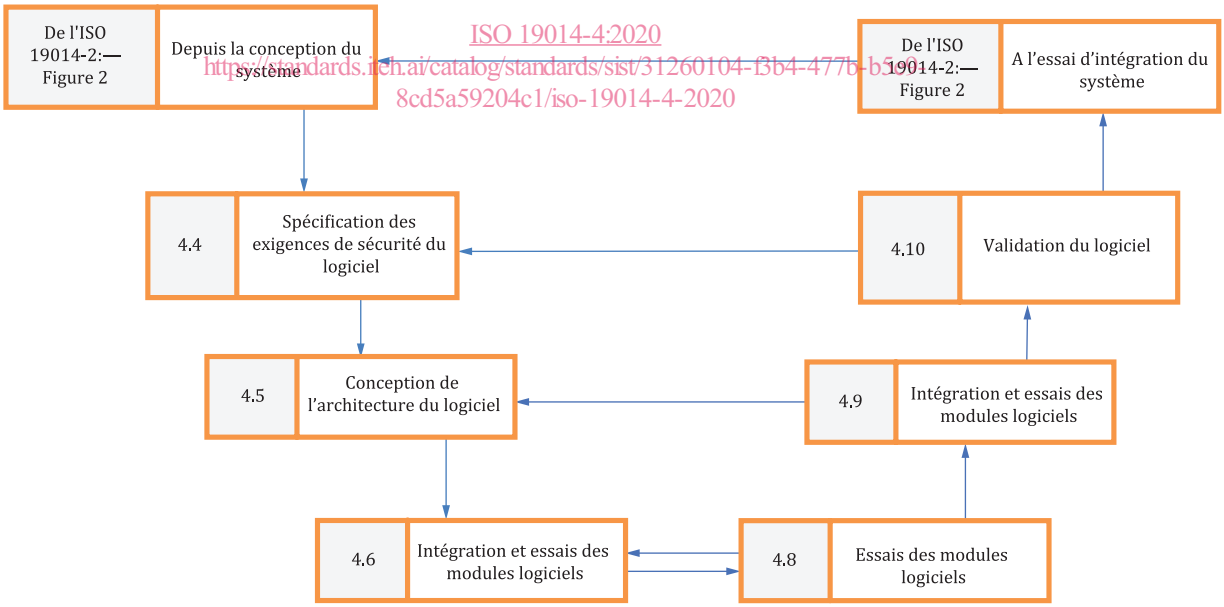

Figure 1 — Développement du logiciel — Modèle en V

La Figure 1 est une représentation d'une méthode de conception possible (modèle en V). Tout processus de développement organisé et reconnu qui répond aux exigences du présent document peut être utilisé pour le développement d'un logiciel.

Lors du choix des méthodes et des mesures, outre le codage manuel, un développement fondé sur un modèle peut être appliqué lorsque le code source est automatiquement généré à partir de modèles.

Pour chaque méthode ou mesure figurant dans les tableaux, il existe un niveau de disposition différent pour chaque niveau de performance. Le Tableau 1 indique les exigences.

Tableau 1 — Spécification des exigences relatives à la sécurité du logiciel

Symbole	Spécification des exigences relatives à la sécurité du logiciel
+	La méthode ou la mesure doit être utilisée pour ce MPLr. Si cette méthode ou mesure n'est pas utilisée, la justification correspondante doit être documentée pendant la phase de planification de la sécurité.
o	La méthode ou la mesure peut être utilisée pour ce MPLr.
–	La méthode ou la mesure n'est pas adéquate pour satisfaire à ce MPLr.

Les méthodes et mesures correspondant au MPLr respectif doivent être choisies. Des méthodes et mesures subsidiaires ou équivalentes sont définies par des lettres qui suivent le nombre. Au moins l'une des méthodes et mesures subsidiaires ou équivalentes marquées d'un «+» doit être choisie, auquel cas il n'est pas exigé de fournir une justification. Le Tableau 2 en contient un exemple.

Tableau 2 — Exemple de spécification des exigences relatives à la sécurité du logiciel

Méthode/mesure		MPLr = a	MPLr = b, c	MPLr = d	MPLr = e
1.a	Mesure 1	+	+	-	-
1.b	Mesure 2	+	+	+	+
1.c	Mesure 3	+	+	+	+

Dans ce cas,

— une mesure de la Mesure 1, Mesure 2 ou Mesure 3 doit être respectée pour MPLr = a, b, c;

— une mesure de la Mesure 2 ou Mesure 3 doit être respectée pour MPLr = d, e;

— dans le cas contraire, une justification doit être fournie au sujet de la méthode ou de la mesure subsidiaire non précisée pour satisfaire à l'exigence de la norme relative au MPLr en question.

Une justification doit être fournie si d'autres méthodes ou mesures équivalentes sont utilisées au lieu des méthodes ou mesures énumérées.

Si un composant de logiciel a un impact sur différentes fonctions de sécurité avec un MPLr différent, les exigences relatives au MPLr le plus élevé doivent s'appliquer.

Si le logiciel contient des composants liés à la sécurité et d'autres non liés à la sécurité, le niveau de performance atteint de la machine (MPLa) global du logiciel intégré doit alors être limité au composant de logiciel ayant le plus petit MPLa; cette exigence ne s'applique pas lorsque l'indépendance adéquate entre les composants de logiciel peut être démontrée conformément à l'Article 7.

Lors de la réutilisation d'un composant de logiciel destiné à être modifié, une analyse d'impact doit être effectuée. Un plan d'action doit être élaboré et mis en œuvre pour l'ensemble du cycle de vie du logiciel, sur la base des résultats de l'analyse d'impact, afin de s'assurer que les objectifs de sécurité sont atteints.

4.3 Artefacts

Une fois les phases individuelles du plan de développement de logiciel déterminées, les artefacts doivent être définis pour chaque phase à réaliser. D'autres phases et artefacts connexes peuvent être ajoutés en

répartissant les activités et les tâches. Compte tenu de l'étendue et de la complexité du projet, tous les artefacts des phases individuelles présentées à la Figure 1 peuvent être modifiés.

NOTE Il est courant de combiner les phases individuelles si la méthode/mesure utilisée rend difficile la distinction claire entre les phases. Par exemple, la conception de l'architecture du logiciel et la mise en œuvre du logiciel peuvent être générées successivement avec le même outil de développement assisté par ordinateur, comme c'est le cas dans le processus de développement fondé sur le modèle.

Dans le cadre du processus de développement du logiciel, les artefacts doivent être:

a) étayés par des documents en fonction des résultats attendus des phases prévues;

b) modifiés à la suite d'une analyse d'impact, et seul le logiciel concerné doit être soumis à un essai de régression;

c) soumis à un processus de gestion de la configuration.

Le premier artefact applicable au processus est le plan de développement du logiciel. Les artefacts ultérieurs, définis par le plan, doivent inclure:

— les spécifications de conception et le rapport de vérification connexe, pour chaque phase de conception du logiciel (branche descendante du modèle en V de la Figure 1);

— les spécifications d'essai et le rapport d'essai correspondant, pour chaque phase d'essai du logiciel (SW) (branche montante du modèle en V de la Figure 1);

— le logiciel exécutable.

4.4 Spécification des exigences relatives à la sécurité du logiciel

La spécification des exigences relatives à la sécurité du logiciel doit décrire les exigences portant sur les éléments suivants, le cas échéant:

— fonctions permettant au système de réaliser ou maintenir un état de sécurité;

— fonctions relatives à la détection, à l'indication et au traitement des défauts par les parties d'un système de commande relatives à la sécurité (SRP/CS);

— fonctions relatives à la détection, à l'indication et au traitement des défauts dans le logiciel;

— fonctions relatives aux essais en ligne et hors connexion des fonctions de sécurité;

NOTE 1 Un essai en ligne est réalisé pendant que le système soumis à l'essai est en cours d'utilisation. Un essai hors connexion est réalisé pendant que le système soumis à l'essai n'est pas en cours d'utilisation.

NOTE 2 Un exemple d'essai en ligne serait la vérification des défauts dans le système de direction pendant la conduite de la machine. Un exemple d'essai hors connexion serait la vérification des défauts dans le système de direction avant d'autoriser le déplacement de la machine.

— fonctions qui permettent de modifier des paramètres du logiciel relatifs à la sécurité;

— interfaces avec des fonctions qui ne sont pas liées à la sécurité;

— performance et temps de réponse;

— interfaces entre le logiciel et le matériel de l'unité de commande électronique.

Les méthodes ou les mesures appropriées doivent être choisies dans le Tableau 3 pour satisfaire au MPLr spécifié.

Tableau 3 — Spécification des exigences relatives à la sécurité du logiciel

Méthode/mesure		Référence	MPLr = a	MPLr = b, c	MPLr = d	MPLr = e
1.	Spécification des exigences en langage naturel	A.1	+	+	+	+
2.	Outils de spécification assistée par ordinateur	A.2	o	o	o	+
3.a	Méthodes informelles	A.3	+	+	+	-
3.b	Méthodes semi-formelles	A.4	+	+	+	+
3.c	Méthodes formelles	A.5	+	+	+	+
4.	Traçabilité directe entre les exigences de sécurité du système et les exigences de sécurité du logiciel	A.6	o	o	o	+
5.	Traçabilité rétrospective entre les exigences de sécurité du système et les exigences de sécurité du logiciel		o	o	o	+
6.a	Revues informelles des exigences de sécurité du logiciel	A.7	+	+	+	–
6.b	Inspection des exigences de sécurité du logiciel	A.8	+	+	+	+
NOTE Les descriptions détaillées de ces méthodes/mesures sont présentées dans l'Annexe A.						

4.5 Conception de l'architecture du logiciel

L'architecture d'un logiciel décrit la structure hiérarchique de tous les composants du logiciel relatifs à la sécurité de chaque système de commande de sécurité (SCS). Elle doit être développée sur la base des exigences relatives à la sécurité du logiciel. Les méthodes ou les mesures appropriées doivent être choisies dans le Tableau 4 pour satisfaire le MPLr spécifié.

Tableau 4 — Conception de l'architecture du logiciel

Méthode/mesure		Référence	MPLr = a	MPLr = b, c	MPLr = d	MPLr = e
1.a	Méthodes informelles	A.3	+	+	+	–
1.b	Méthodes semi-formelles	A.4	+	+	+	+
1.c	Méthodes formelles	A.5	+	+	+	+
2.	Outils de conception assistée par ordinateur	A.9	o	o	o	+
3.a	Comportement cyclique, avec temps de cycle maximal garanti	A.10	o	o	+	+
3.b	Architecture de type «à déclenchement temporel»		o	o	+	+
3.c	Déclenché par événement, avec un temps de réponse maximal garanti		o	o	+	+
4.	Traçabilité directe entre la spécification des exigences de sécurité du logiciel et l'architecture du logiciel	A.6	o	o	o	+
5.	Traçabilité rétrospective entre l'architecture du logiciel et la spécification des exigences de sécurité du logiciel		o	o	o	+
6.a	Revues informelles de l'architecture du logiciel	A.7	+	+	+	–
6.b	Inspection de l'architecture du logiciel	A.8	+	+	+	+
NOTE Les descriptions détaillées de ces méthodes/mesures sont présentées dans l'Annexe A.						

4.6 Conception et codage des modules logiciels

Les objectifs de cette phase de développement du logiciel consistent à:

— spécifier en détail le comportement des modules logiciels relatifs à la sécurité qui sont spécifiés par l'architecture du logiciel;

— générer des modules logiciels lisibles, vérifiables et maintenables (par exemple code manuel, modèle, etc.);

— vérifier que l'architecture du logiciel a été totalement et correctement mise en œuvre.

Les méthodes ou les mesures appropriées doivent être choisies dans le Tableau 5 pour satisfaire le MPLr spécifié. Il n'est pas exigé d'examiner le code généré automatiquement.

Tableau 5 — Conception et codage du module logiciel

Méthode/mesure		Référence	MPLr = a	MPLr = b, c	MPLr = d	MPLr = e
1.a	Méthodes informelles	A.3	+	+	–	–
1.b	Méthodes semi-formelles	A.4	+	+	+	+
1.c	Méthodes formelles	A.5	+	+	+	+
2.	Outil de conception assistée par ordinateur	A.9	o	o	o	+
3.	Utilisation des normes de conception et de codage	A.11	o	+	+	+
4.	Pas de flux de contrôle non structuré dans les programmes en langages de plus haut niveau[b]		o	o	+	+
5.	Conversion de type automatique limitée[b]		o	o	+	+
6.	Utilisation limitée des interruptions[b]		o	o	o	+
7.	Utilisation limitée des pointeurs[b]		o	o	o	+
8.	Utilisation limitée de la récursivité		o	o	o	+
9.a	Variables ou objets dynamiques sans vérification en ligne[b]	A.12	o	o	–	–
9.b	Variables ou objets dynamiques avec vérification en ligne[b]	A.13	o	o	+	+
10.	Limite de la taille du module logiciel	A.14	+	+	+	+
11.	Un point d'entrée/un point de sortie dans les sous-programmes et les fonctions[b]		o	+	+	+
12.	Interface entièrement définie		o	+	+	+
13.	Dissimulation/encapsulation de l'information		o	o	+	+
14.	Contrôle de la complexité du logiciel		o	o	o	+
15.	Conception ou codage structuré(e)	A.15	o	+	+	+
16.	Conception ou codage défensif(ve)	A.16	o	o	o	+
17.	Utilisation d'éléments logiciels de confiance/vérifiés[a]	A.17	o	o	o	o
18.	Traçabilité directe entre la spécification des exigences relatives à la sécurité du logiciel et la conception du logiciel	A.6	o	o	o	+
19.a	Revues informelles de la conception du logiciel, du code source ou des deux	A.7	+	+	+	–
19.b	Inspection de la conception du logiciel, du code source ou des deux	A.8	+	+	+	+

NOTE Les descriptions détaillées de ces méthodes/mesures sont présentées dans l'Annexe A.

[a] L'utilisation d'éléments logiciels de confiance et vérifiés est fortement recommandée.

[b] Ces méthodes ou mesures ne s'appliquent pas toujours aux notations de modélisation graphique utilisées dans le développement fondé sur un modèle.

4.7 Choix du langage et des outils

L'intégrité de la sécurité du logiciel en cours de développement peut être directement affectée par le langage de programmation choisi, les outils utilisés pendant le développement et les essais, et