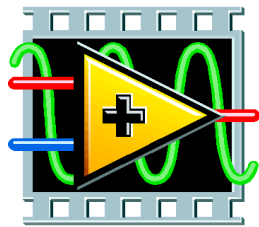


Préparation à la certification CLD



NATIONAL INSTRUMENTS

LabVIEW[™]



CERTIFIED DEVELOPER

Maxime RENAUD - MESULOG
Chef de projet – Développeur logiciel

Certified LabVIEW Architect

- ❑ Développeur LabVIEW depuis septembre 2013
- ❑ Chef de projet et développeur
- ❑ Certifications :
 - Certifié CLAD en 2008 et en 2013
 - Certifié CLD en janvier 2014
 - Certifié CLA en septembre 2016
 - Certifié CTD en octobre 2016



- ❑ **Présentation de MESULOG**
- ❑ **Préparation à la certification CLD**
 - Les certifications NI
 - Organisation de l'examen
 - Objectifs et thèmes de l'examen
 - Recommandations, trucs et astuces
- ❑ **Les étapes suivantes**

❑ Création

- Statut : S.A.R.L. au capital de 15 000 €
- Démarrage activité : mars 2000

❑ Localisation

- Grenoble (Parc Activités Centr'Alp)

❑ Effectif

- 6 ingénieurs
- 1 technicien

❑ Données 2014

- Chiffre d'affaires HT : 462 k€
- Résultat net comptable : +10 k€

- ❑ **Développement logiciel**
 - Orienté test et mesure
 - Sur plate-forme logicielle National Instruments™

- ❑ **Étude, conseil, expertise**
 - Étude de faisabilité
 - Optimisation, fiabilisation de code existant
 - Expertise, assistance technique

❑ Plate-formes logicielles

- LabVIEW
- LabVIEW Real-Time
- LabVIEW FPGA
- LabVIEW DSC
- TestStand
- VeriStand



❑ Technologies

- Acquisition de données
- PXI et instrumentation
- Commande d'axes
- Génération de rapports
- Liaison base de données
- OPC

- ❑ Membre NI Alliance



- ❑ Certification National Instruments

- LabVIEW



- TestStand

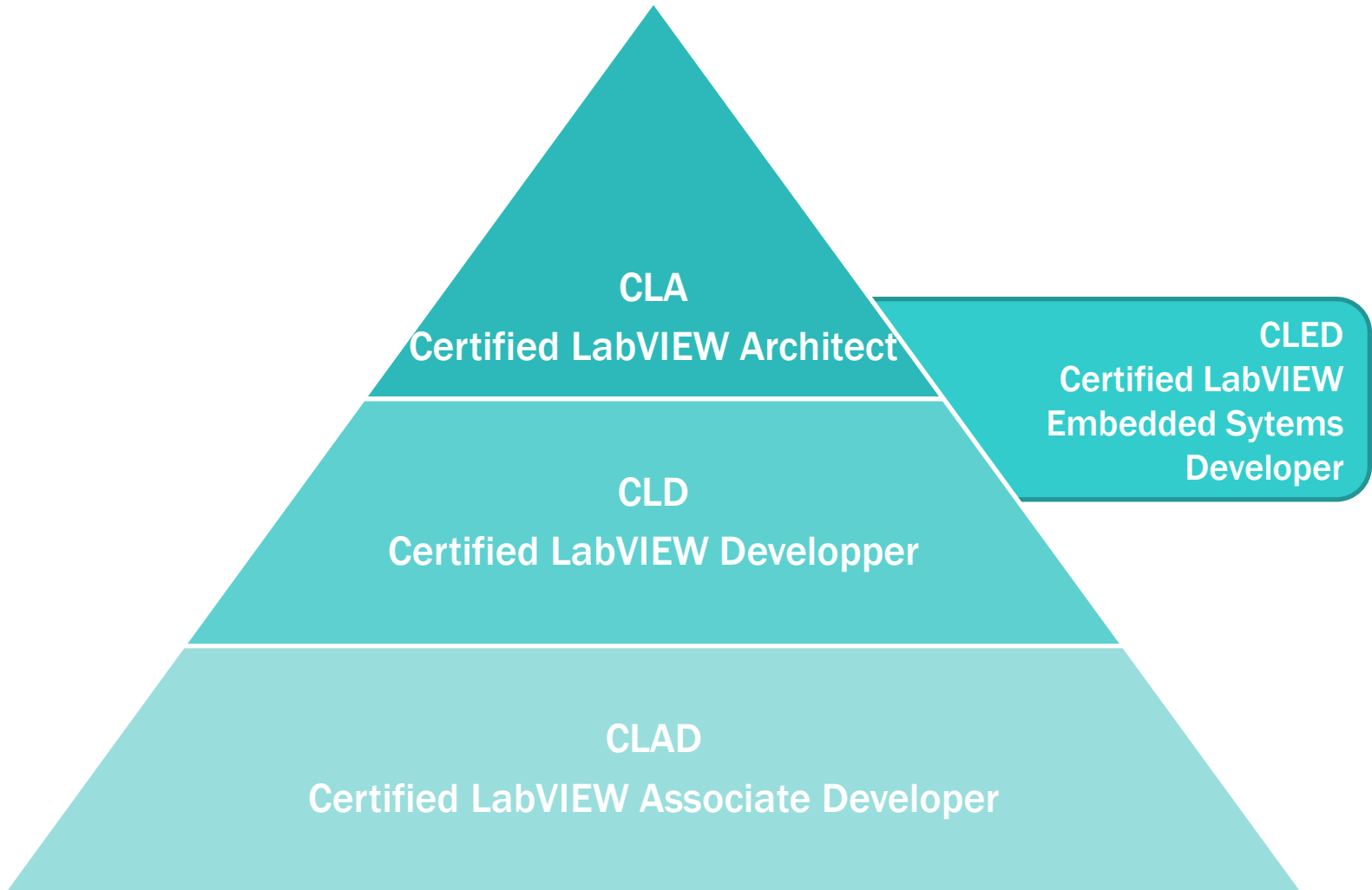


- ❑ ALSTOM T&D
- ❑ AREVA NP
- ❑ CETIAT
- ❑ HONEYWELL Security
- ❑ LEGRAND
- ❑ L.N.E.
- ❑ ONERA
- ❑ RADIALL
- ❑ RENAULT
- ❑ STMICROELECTRONICS
- ❑ SOCOMEC
- ❑ THALES Alenia Space
- ❑ THALES Avionics LCD

❑ Autres compétences

- Métrologie
- Interférométrie laser
- Mesure de position
- Codeurs incrémentaux / absolus
- Liaison PC ↔ automate

Les certifications NI



- ❑ **Monter en compétences avec la préparation à l'examen**
 - Découverte de nouvelles techniques
 - Amélioration de la vitesse d'écriture de code
 - Mise en œuvre de bonnes pratiques
- ❑ **Justifier un niveau de compétence**
- ❑ **Gagner en reconnaissance**
- ❑ **Gagner en confiance en soi sur l'outil de développement**

- ❑ **Requiert et démontre :**
 - Bonne connaissance de l'environnement LabVIEW
 - Capacité à résoudre des problèmes
 - Utilisation et création de code modulaire, évolutif et maintenable
 - Documentation du code
 - Bonne vitesse de codage / câblage
 - Mise au point, test et débogage de fonctions/applications

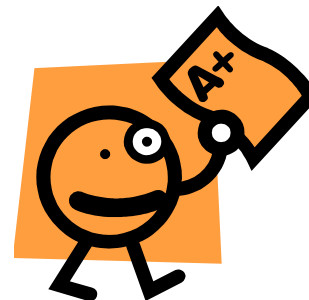
- ❑ **Durée de validité : 3 ans**

Organisation de l'examen

- ❑ Mise à disposition d'un PC avec une version récente de LabVIEW Professional en français
- ❑ Matériel :
 - Sujet papier de l'examen en français
 - Une clé USB contenant :
 - Prototype d'IHM contenant les commandes et indicateurs
 - Fichiers E/S nécessaires si besoin
- ❑ Conditions :
 - Interdiction d'utiliser des ressources externes
 - Utilisation autorisée de toutes les ressources disponibles dans LabVIEW (Modèles, exemples, VI Express...)
 - Durée de l'examen : 4 heures
- ❑ Seul le code sauvegardé sur la clé sera évalué

Critère d'évaluation	Points	Pourcentage
Fonctionnalités	15	37.5 %
Style	15	37.5 %
Documentation	10	25 %
Total des points	40	100 %

Pour valider l'examen,
il faut un score de **70% ou plus**

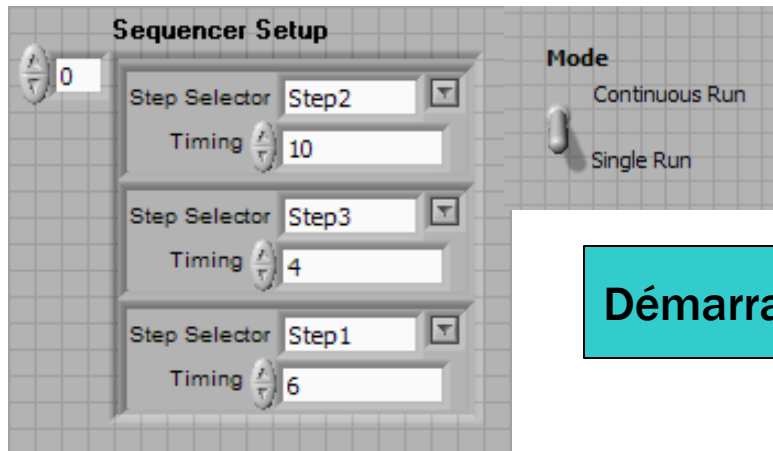


- ❑ **Évaluation réalisée par 2 ingénieurs NI**
 - Délai d'évaluation de 4 semaines
 - ❑ **Style et documentation :**
 - Utilisation du VI Analyzer
 - Vérification des méthodes mais pas du contenu
 - ❑ **Fonctionnalités :**
 - Test des fonctions : valeurs des indicateurs en fonction des commandes
 - Points par catégories, et sous-catégories
 - Chaque point correspond à une fonctionnalité demandée
 - Documentation ou code complexe ≠ fonctionnalité
 - ❑ **Évaluation objective, pas de “ça marche à peu près”**
- Implémentez des fonctions simples et modulaires

Objectifs et thème de l'examen

- ❑ Les examens sont tous basés sur une application séquentielle
- ❑ L'ordre d'exécution est configurable depuis l'IHM ou un fichier
- ❑ Le cadencement est une composante essentielle de l'application
- ❑ Des accès en lecture et/ou écriture sur des fichiers sont possibles
- ❑ L'application doit répondre aux événements utilisateurs en moins de 100 ms
- ❑ La gestion d'erreur est simple

Scénarios possibles



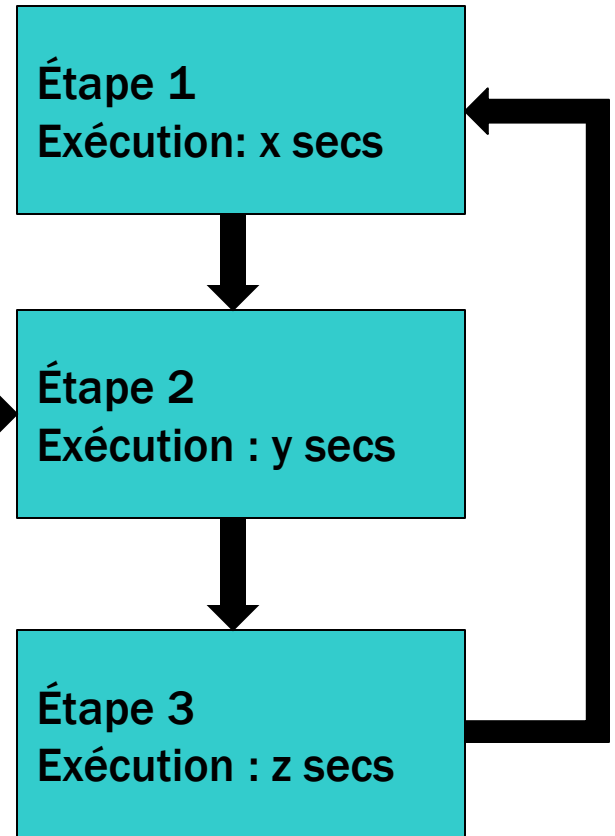
Démarrage

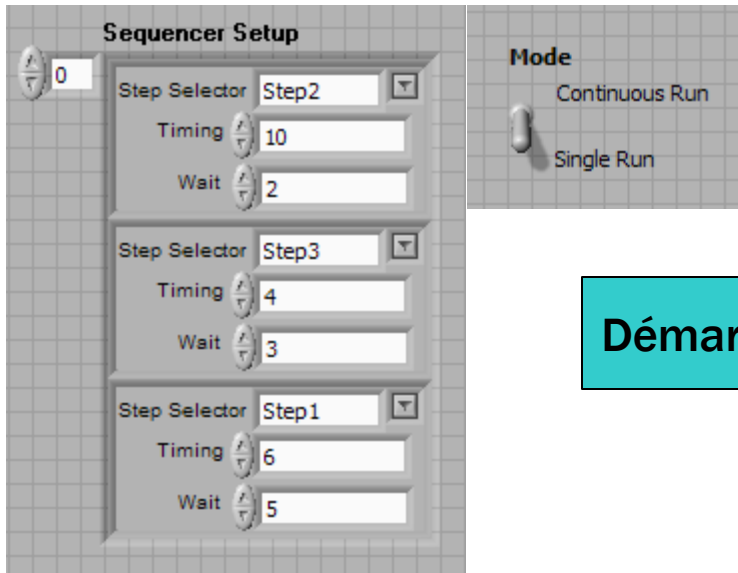
Étape 1
Exécution: x secs

Étape 2
Exécution : y secs

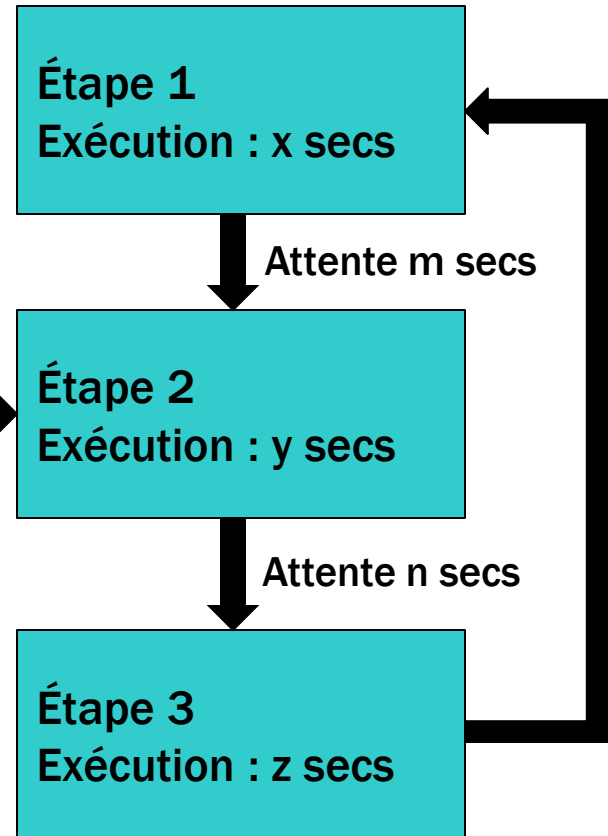
Étape 3
Exécution : z secs

L'application doit pouvoir s'adapter facilement à l'ajout de nouvelles étapes





Démarrage

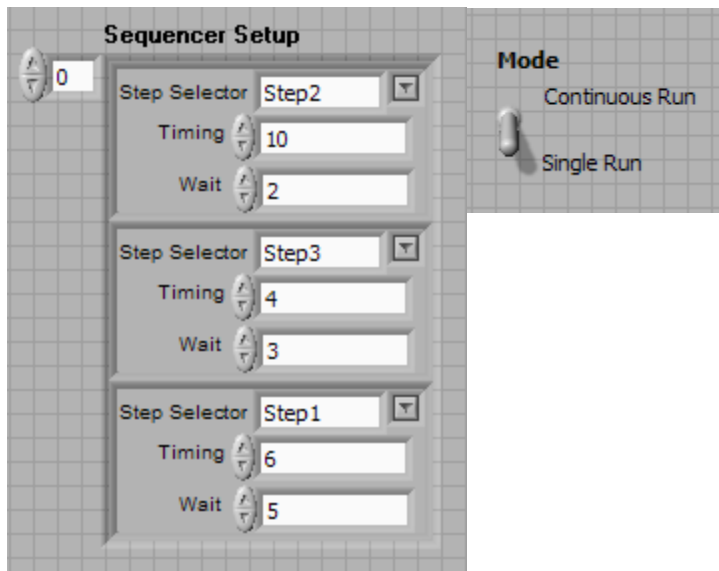


L'application doit pouvoir s'adapter facilement à l'ajout de nouvelles étapes

Fichier csv ou ini

Étape 2,10,2
Étape 3,4,3
Étape 1,6,5

Démarrage



Étape 1
Exécution : x secs

Attente m secs

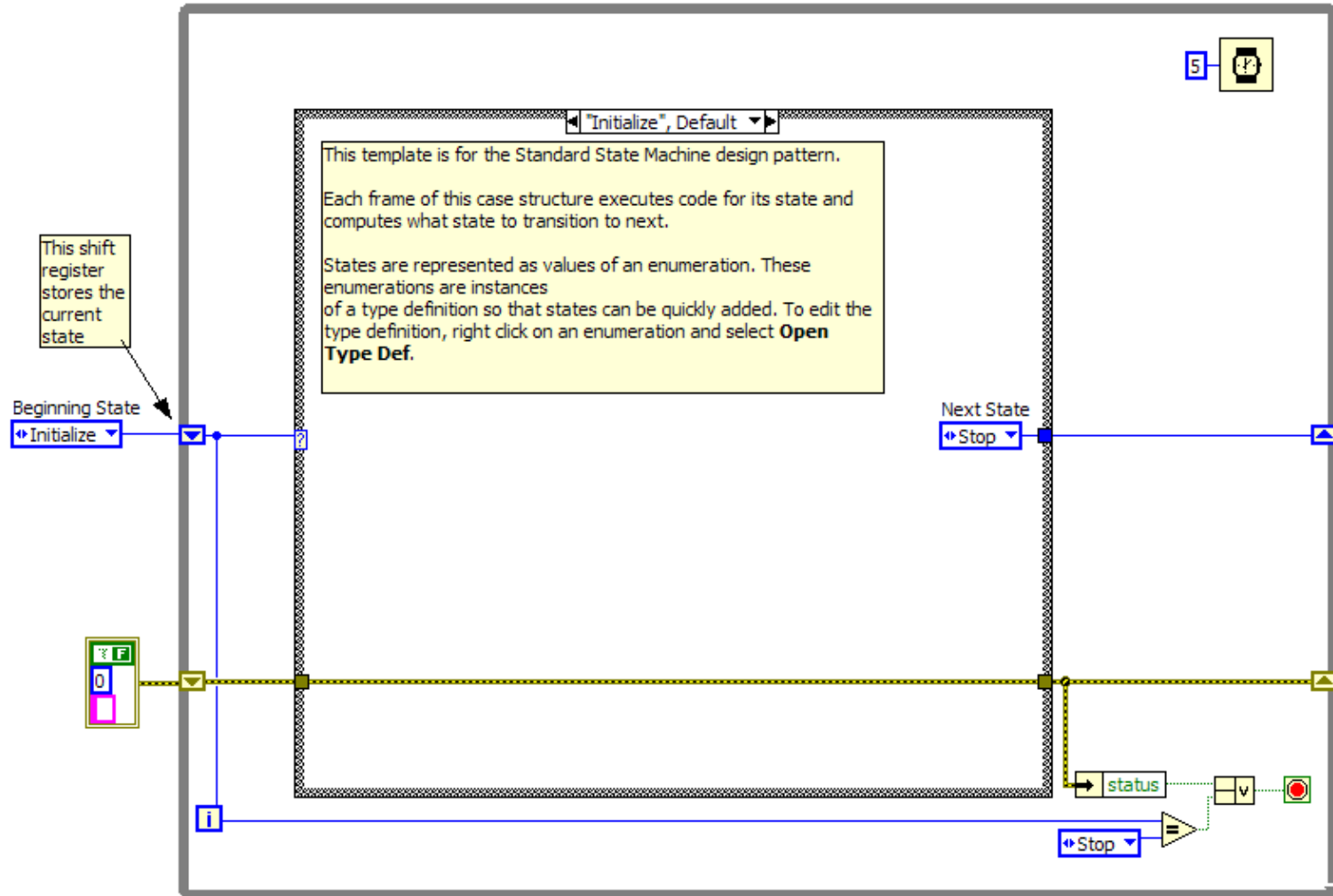
Étape 2
Exécution : y secs

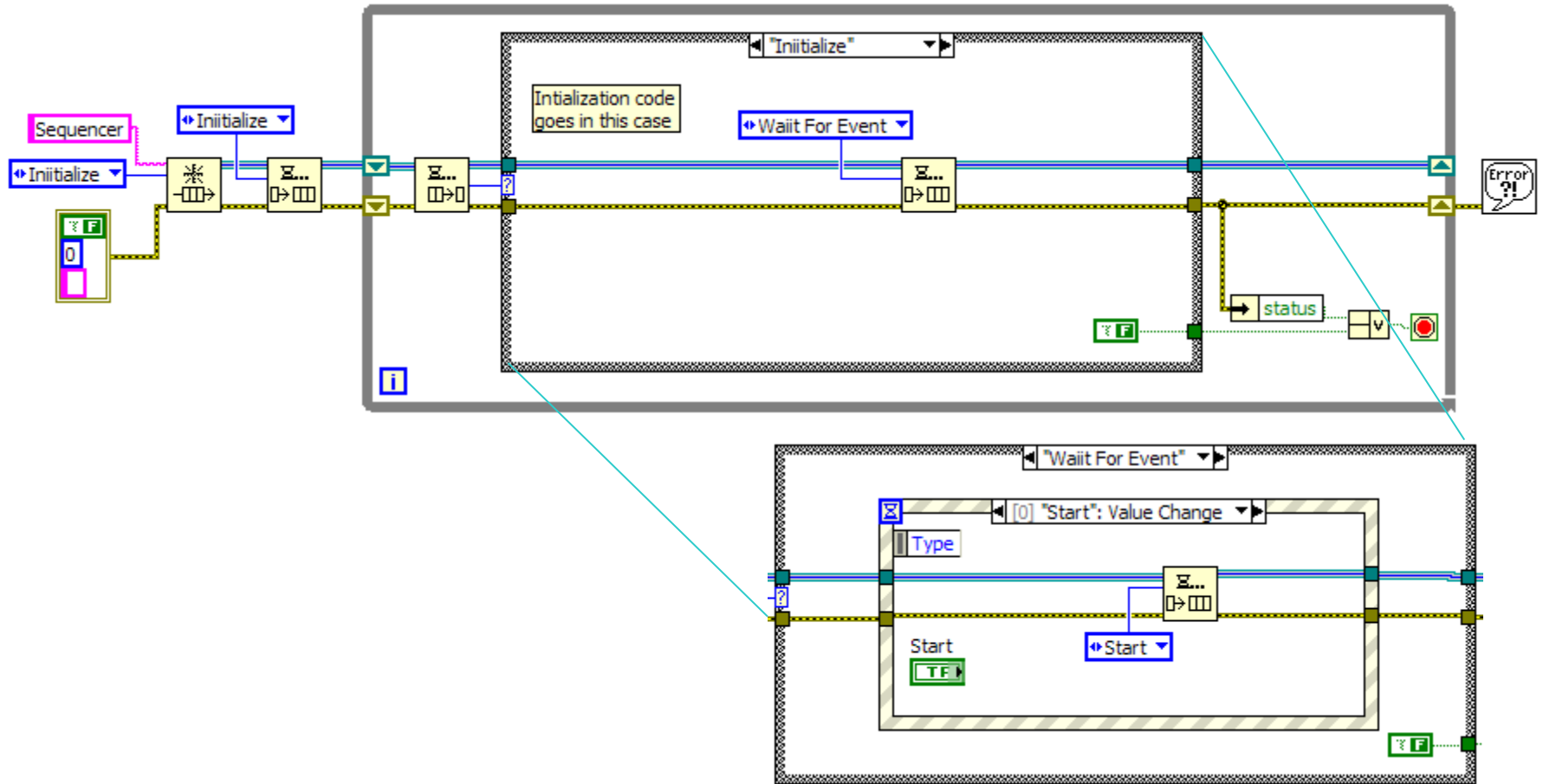
Attente n secs

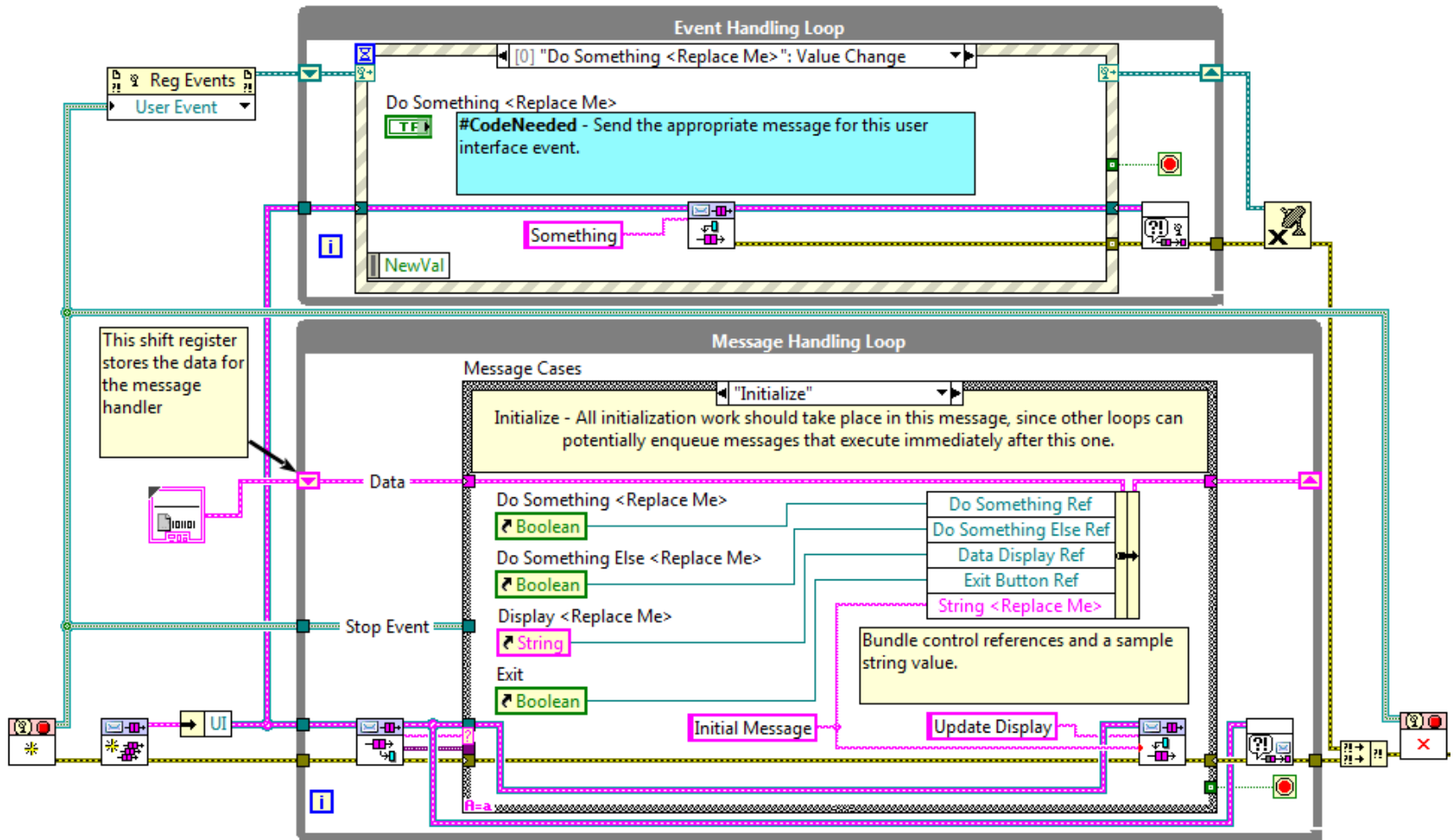
Étape 3
Exécution : z secs

Attente k secs

Modèle de conception







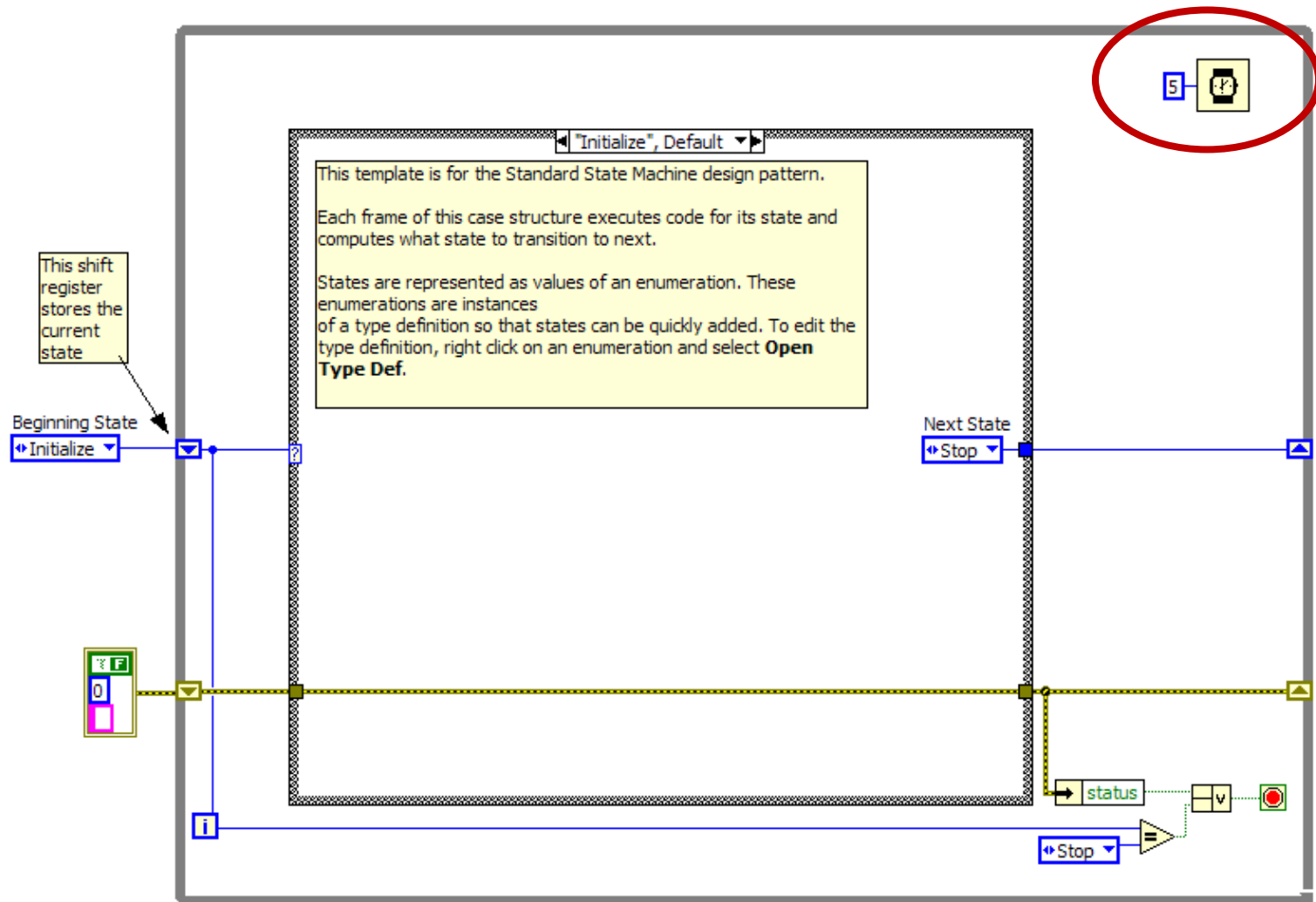
- ❑ **Généré à partir des modèles de projet de LabVIEW**
- ❑ **Fusion du VI prototype avec le VI principal du modèle**
 - Placer le VI fourni dans le projet
 - Copier le contenu du main du modèle dans le VI fourni
 - Supprimer le main.vi du modèle
- ❑ **Avantages :**
 - Création d'un projet avec de la documentation
 - Gestion de l'arrêt de l'application
 - Modularité
- ❑ **Considérations**
 - Nécessite une phase de prise en main pendant la préparation de l'examen si ce modèle est méconnu
 - Attention à l'arrêt de l'application et à la gestion de l'erreur

Modèle de conception	Avantages	Considérations
Machine à état	<ul style="list-style-type: none"> • Permet l'exécution en séquence 	<ul style="list-style-type: none"> • Pas de bufférisation • Réactivité de l'IHM
Machine à état à file d'attente avec événement	<ul style="list-style-type: none"> • Bufférisation → Empilement d'actions • Gestion des événements utilisateurs possible 	<ul style="list-style-type: none"> • Gestion de multiples événements • Gestion complexe des événements de l'IHM
Modèle QDMH	<ul style="list-style-type: none"> • Réactif aux événements utilisateurs • Gestion simple d'un grand nombre d'événement • Documentation • Style 	<ul style="list-style-type: none"> • Implémentation des actions non événementielles • Prise en main nécessaire

Gestion du temps pour le CLD

Type de cadencement	Méthode de cadencement	Mise en œuvre
Contrôle du temps d'exécution	Attente (ms) Attendre un multiple de (ms)	Cadence l'exécution d'une machine d'état libérant de la ressource pour les autres processus.

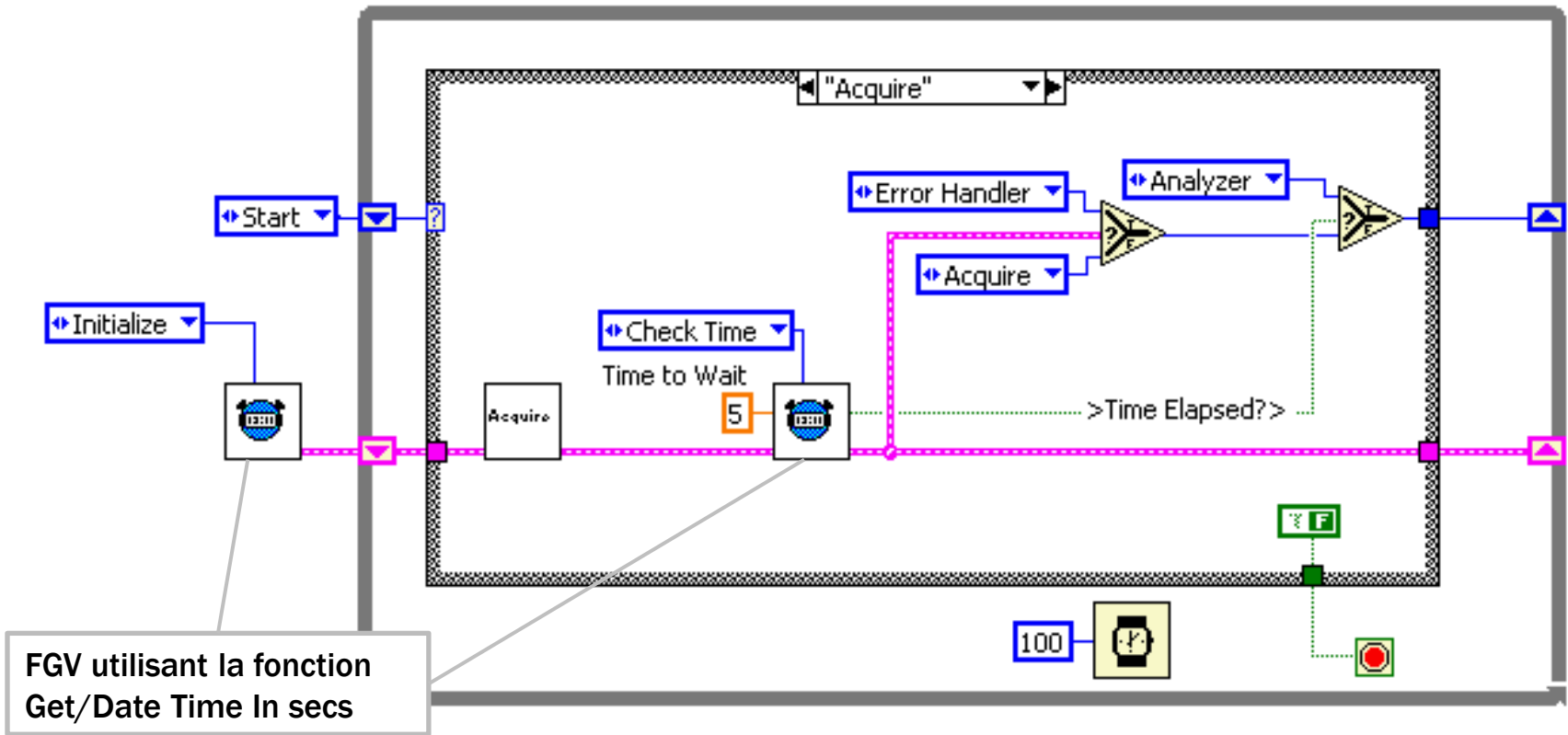
- ❑ Les fonctions de contrôle du temps d'exécution permettent de définir à quelle cadence une boucle s'exécute sur le processeur



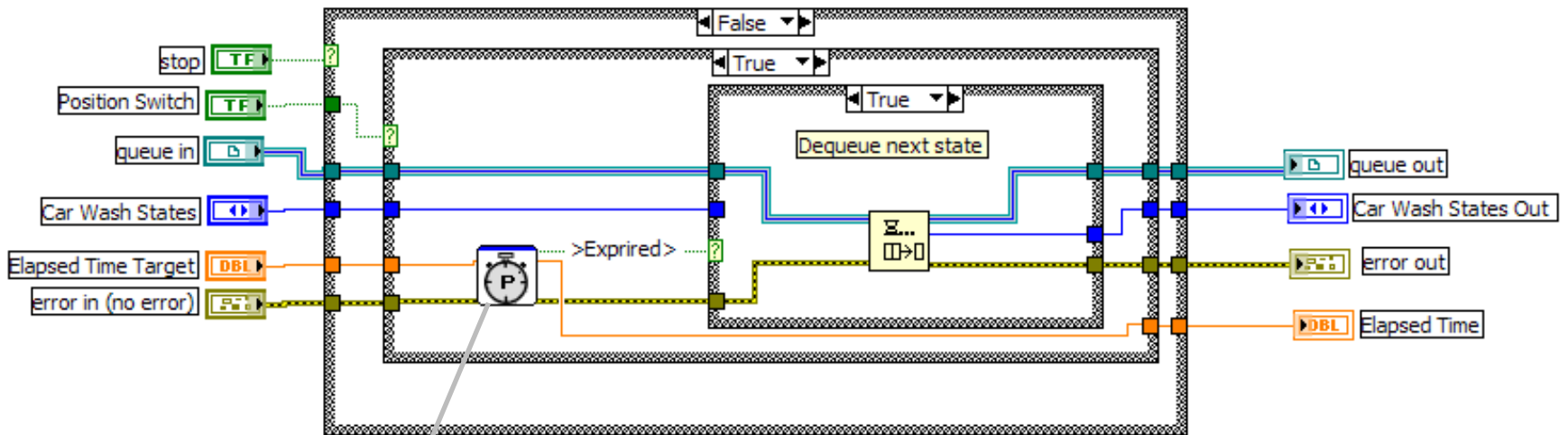
Type de cadencement	Méthode de cadencement	Mise en œuvre
Cadencement logiciel	Get Date/Time in sec. Tick Count	Bonne méthode générale de cadencement pour des opérations séquentielles. Encapsulation dans une FGV ou sous-VI
	VI Express “Temps écoulé”	Mesure le temps écoulé avec possibilité de faire un stop/reset. Solution clé en main si on n’a pas besoin de faire une pause !!

❑ Autre technique utilisable :

- Timeout de la structure événement – utile pour arrêter une opération si aucune activité de l'utilisateur est détectée

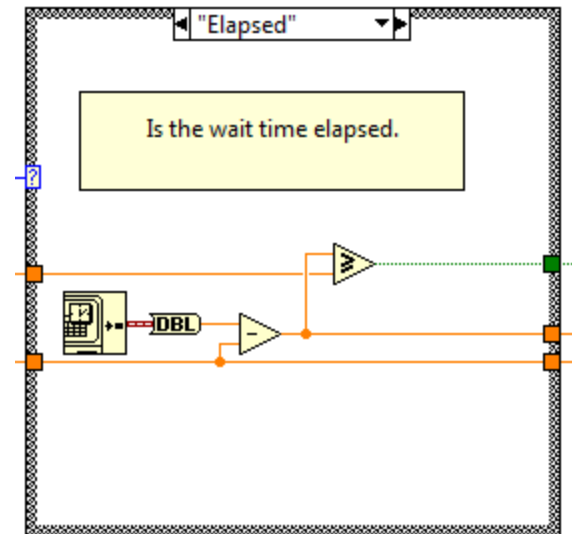
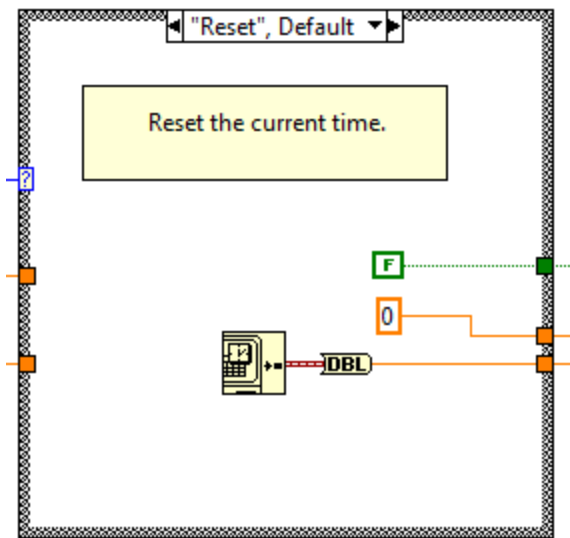
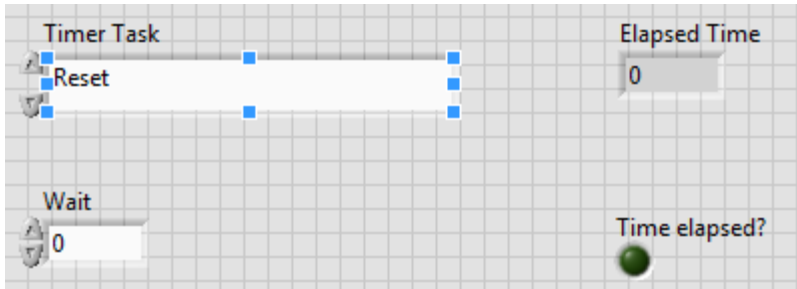


FGV utilisant la fonction
Get/Date Time In secs



VI Express Temps écoulé

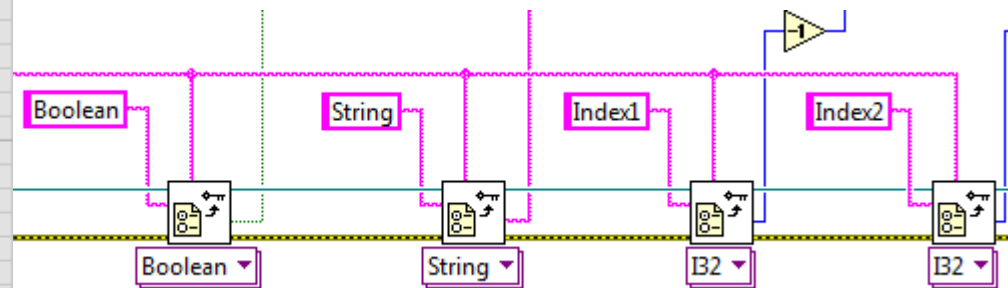
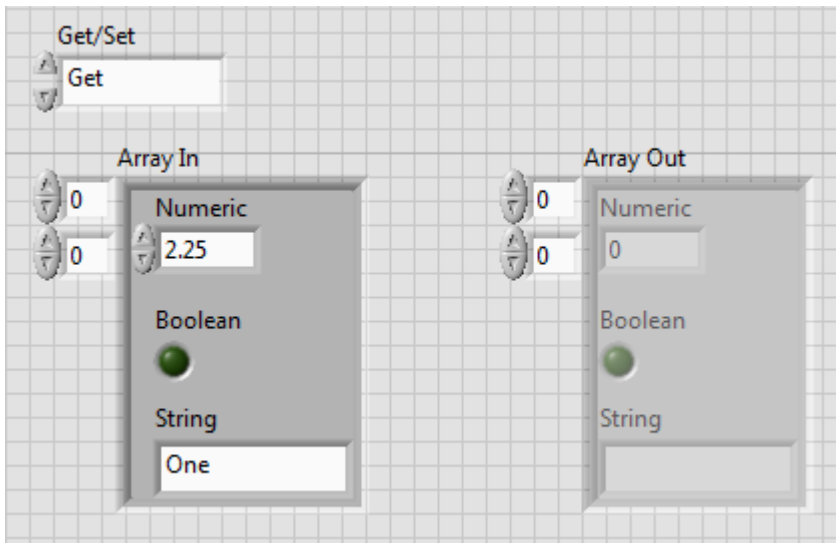
Sous VI issu de l'exemple d'examen "Car Wash sample exam"



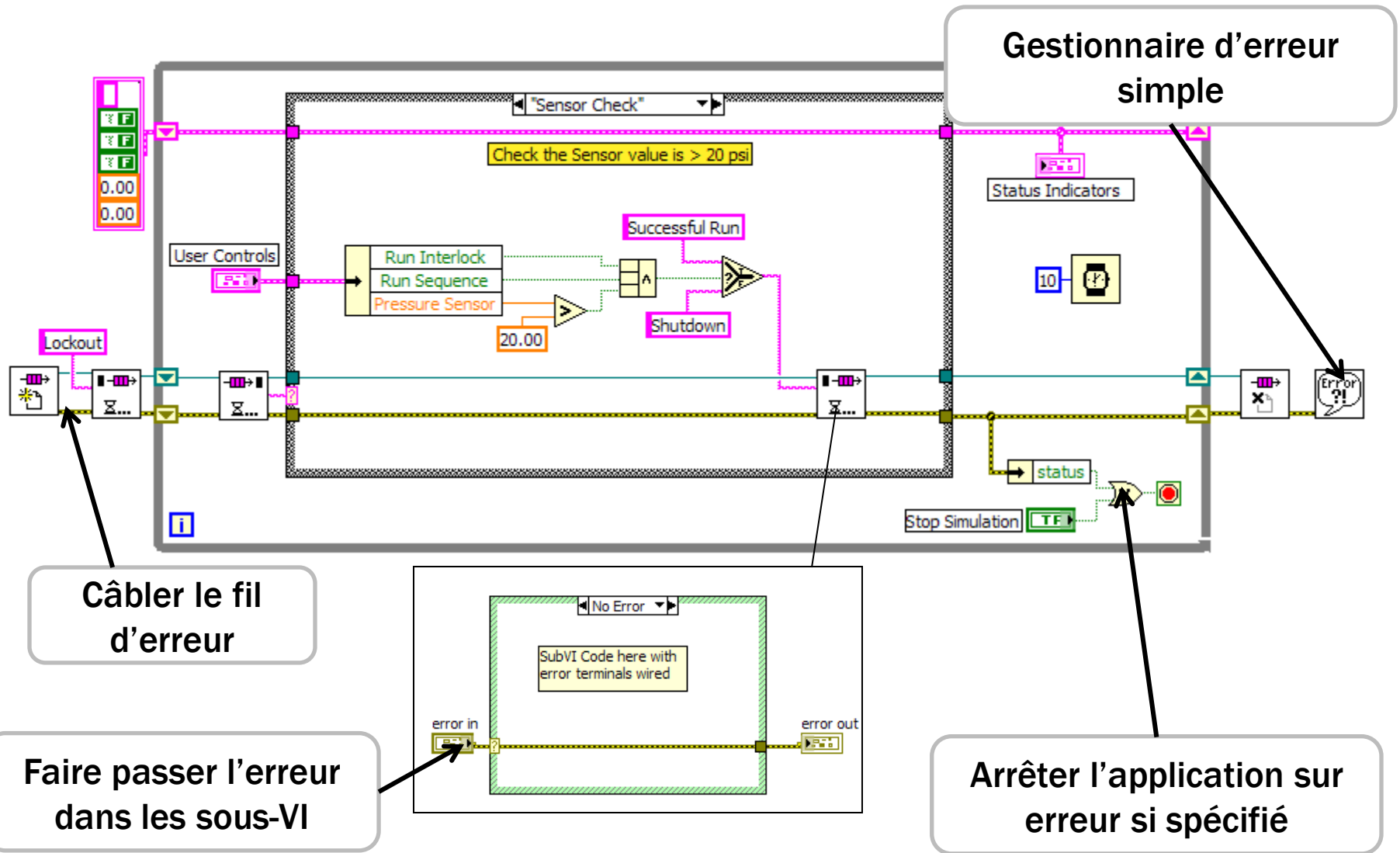
- ❑ Il faut savoir coder un VI permet de connaître le temps écoulé avec la possibilité de mettre en pause ce compteur
- ❑ L'utilisation de la FGV est une méthode efficace
- ❑ Ce VI ne doit pas être bloquant. Il ne fait que retourner le temps restant et si le temps est écoulé

Entrée/Sortie sur fichier

- ❑ Éventuellement besoin de lire un fichier CSV ou INI
- ❑ Faire cette fonction sous forme d'une Variable Globale Fonctionnelle (FGV)
- ❑ Utiliser les fonctions natives pour les INI ou de lecture/écriture de fichier tableur



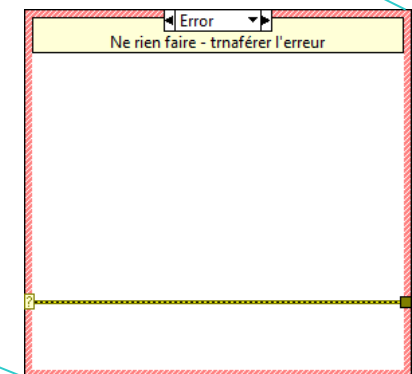
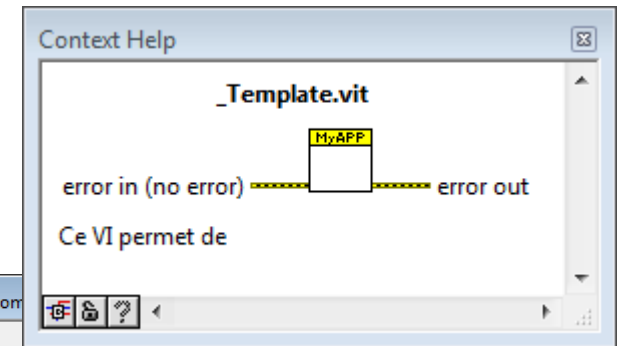
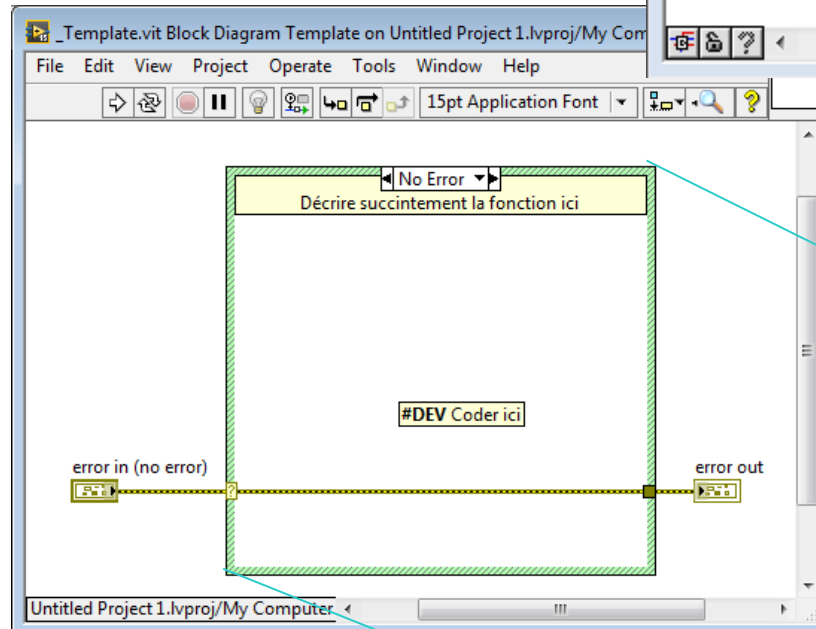
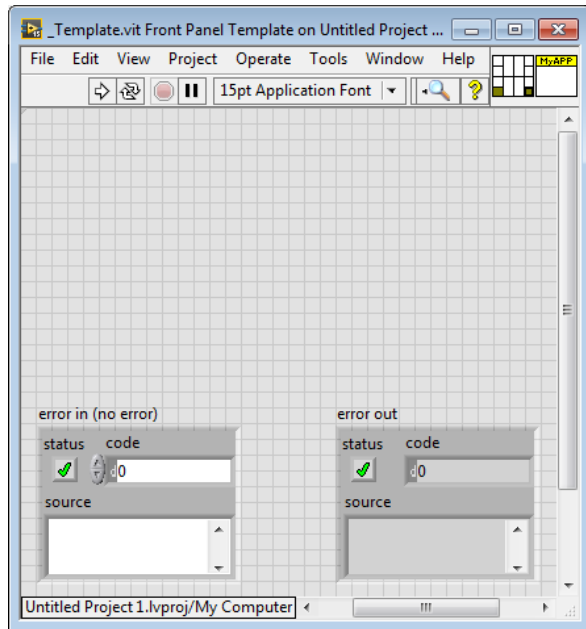
Gestion d'erreur



Recommandations, trucs et astuces

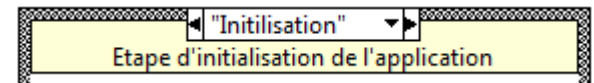
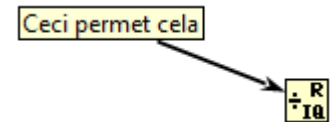
- ❑ Lire le contenu de préparation existant (il y en a beaucoup)
 - Site NI
- ❑ Connaître les règles de programmation LabVIEW
 - Aide LabVIEW >> LabVIEW Style Checklist
- ❑ Préparation pratique
 - Faire un examen blanc en conditions réelles
 - Faire une revue de son code par rapport aux exigences du sujet
 - Regarder les solutions proposées
 - Faire le deuxième sujet blanc en conditions réelles
- ❑ Étaler cette préparation sur plusieurs semaines
- ❑ Le préparer avec des collègues si possible
- ❑ Connaître les règles à l'avance.
Seul le sujet change, ca fait gagner du temps

- ❑ Utiliser le modèle de projet QDMH
- ❑ Créer un template de sous-VI



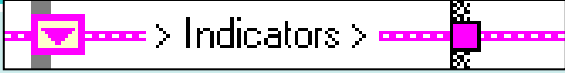

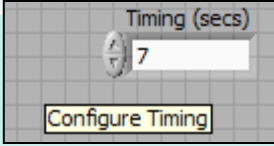
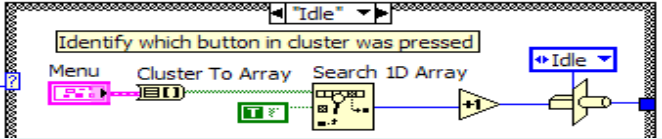
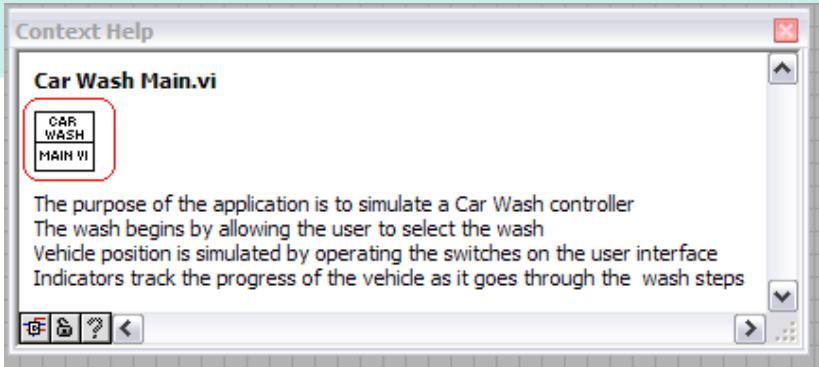
❑ Créez des diagrammes lisibles et documentés

- Limitez les fils croisés
- Utilisez des mots-dièses pour les commentaires. Exemple : #DEV
- Liez les commentaires au code correspondant avec la flèche de commentaire
- Remplissez les commentaires des structures
- Documentez dans la langue qui vous fait gagner le plus de temps



→ Vous êtes évalué sur l'utilisation des outils,
par sur la qualité du commentaire

→ La documentation représente 25% de la notation et est
indépendante des fonctionnalités

Documentation	Exemple
Étiquettes sur les fils	
Étiquettes sur les constantes	
Infobulles sur les commandes et indicateurs	
Commentaires dans le diagramme	
Icône et documentation des VIs	

Soyez bref !!

- ❑ Limitez l'utilisation de variables locales, privilégiez les nœuds de propriétés
- ❑ Nœuds de propriétés
 - Utilisez des nœuds de propriétés pour mettre à jour une commande
 - Toujours câbler les fils d'erreur → Faute pour le VI Analyzer
- ❑ Mettez à jour un indicateur en le câblant
- ❑ Utilisation obligatoire de définitions de type pour les clusters et les enums
- ❑ Fermez les références ouvertes explicitement
- ❑ Évitez les nœuds de coercition et les valeurs par défaut pour les tunnels
- ❑ Évitez la duplication de code → Créer un sous-VI

Options recommandées de LabVIEW

❑ Front Panel

- Connecteur Requis par défaut → Force un bon style

❑ Diagramme

- Désactiver les terminaux sous forme d'icône → Place sur le diagramme
- Subdiagram label by default → Force à documenter le code

❑ Environnement

- Show Created Constant labels → « Documentation automatique »

❑ Avant le démarrage

- Arriver un peu à l'avance et bien reposé
- Évitez la nuit blanche de révision la veille !
- Emmenez un petit truc à manger et à boire si besoin
- Configurez rapidement votre environnement LabVIEW

❑ Au début de l'examen

- Pas la peine de relire les consignes, vous les connaissez déjà
- Parcourez l'ensemble du sujet au début
- Validez votre architecture avant de démarrer
- Identifiez les sous-VI à créer
- Choisissez la méthode de cadencement

- ❑ **Faites les choses dans l'ordre**
 - Concentrez-vous sur la réalisation des éléments principaux un par un
 - Assurez vous que éléments principaux sont fonctionnels
 - Ne passez pas trop de temps sur une fonction → Mettez un commentaire plutôt que de finir à tout prix cette fonction
- ❑ **Dès la création d'un VI, faites son icône et sa documentation, vous n'aurez plus besoin de revenir dessus**
- ❑ **Faites des sauvegardes régulières**
- ❑ **Essayez de rester concentré** durant tout l'examen, 4h c'est court !!!
- ❑ **20 dernières minutes : polissage final**
 - Repasser rapidement sur la documentation de vos VIs
 - Remettez en forme du code
 - Vérifiez que tout votre code est bien sauvegardé
 - Documentez des fonctionnalités manquantes

- ❑ **Innover** !!! Vous n'aurez pas le temps d'essayer de nouvelles techniques
- ❑ **Paniquer** !!! Vous vous êtes entraîné, il n'y a pas de raisons d'échouer. Votre application n'a pas besoin d'être exécutable à tout prix.
- ❑ **Supprimer le dossier de travail** du PC avant d'avoir vérifié que tout était bien sur la clé USB
- ❑ **Parler** du contenu de l'examen une fois terminé !!!

- ❑ Télécharger les supports de préparation de NI
 - <http://sine.ni.com/nips/cds/view/p/lang/fr/nid/10647>
- ❑ Utiliser LabVIEW régulièrement
- ❑ S'entraîner
- ❑ S'inscrire pour une session en contactant NI France
 - 01.57.66.24.24
 - france.fomation@ni.com



**KEEP
CALM
AND
GOOD
LUCK**



Maxime RENAUD
Email : mxr@mesulog.fr
Tél : 04.76.35.20.17