



# Innover pour Prévenir les Infections : Le Rôle des Systèmes Embarqués chez STERIS

STERIS Canada ULC

# Plan



*PRÉSENTATION DE  
STERIS*



*SYSTÈMES  
EMBARQUÉS 101*



*PROTOCOLES DE  
COMMUNICATION*



*GESTION DE  
MÉMOIRE*



*DIVERS*

# Présentation de STERIS





STERIS

17 000 employés

Opère dans plus de 100 pays

Laveurs

Stérilisation

Endoscopie

Pharmaceutique

Produits de désinfection

Dentaire

Tables chirurgicales

# Que fait STERIS

## Notre mission

Aider nos clients à  
créer un monde  
plus sain et plus  
sûr

Décontamination  
d'instruments  
médicaux souillés  
afin de les  
réutiliser

- Stérilisateurs
- Laveurs/Désinfecteurs





Qu'est-ce qu'un  
Laveur/Désinfecteur ?



## Qu'est-ce qu'un Laveur/Désinfecteur ?

---

60 entrées/sorties  
(IOs)

Plus de 350 Classes  
avec plus de 100  
000 lignes de code

3 ans de  
développement

Connectivité et  
maintenance  
préventive





## Réglementation



Régis par la FDA /  
Santé Canada



Normes ISO



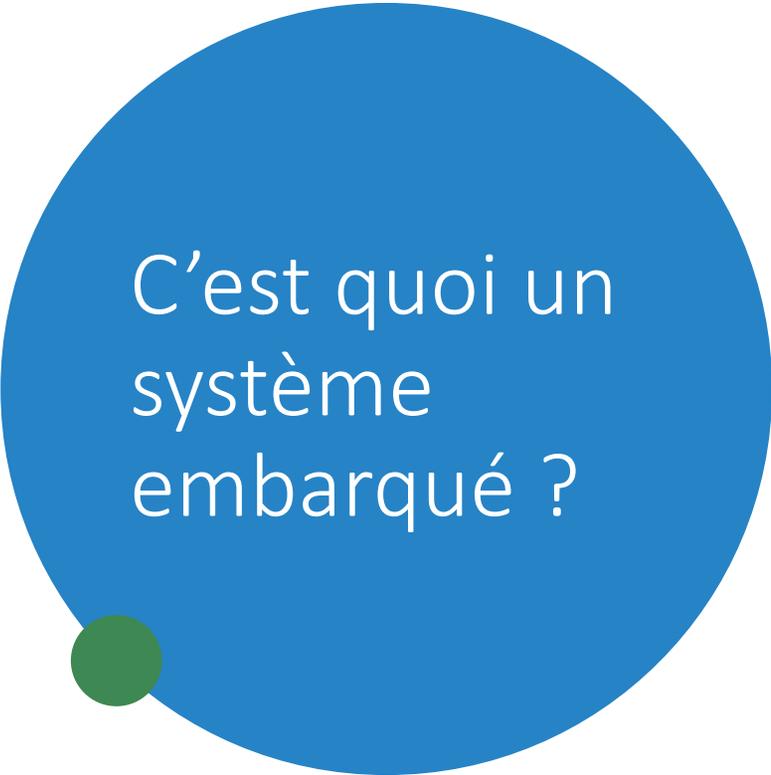
Standards STERIS  
encore plus élevés



Obligation de  
démontrer la  
conformité des laveurs

# Les systèmes embarqués





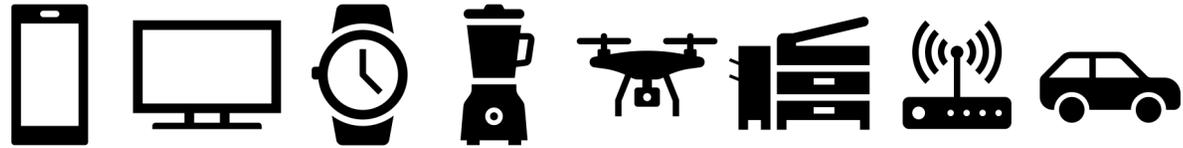
C'est quoi un  
système  
embarqué ?



Un système électronique et informatique autonome, souvent temps réel, spécialisé dans une tâche précise. Ses ressources sont généralement limitées spatialement (encombrement réduit) et énergétiquement (consommation restreinte).

# Programmation Embarquée Omniprésente

Produits Public  
Général



Procédés  
industriels



Produits  
réglementés

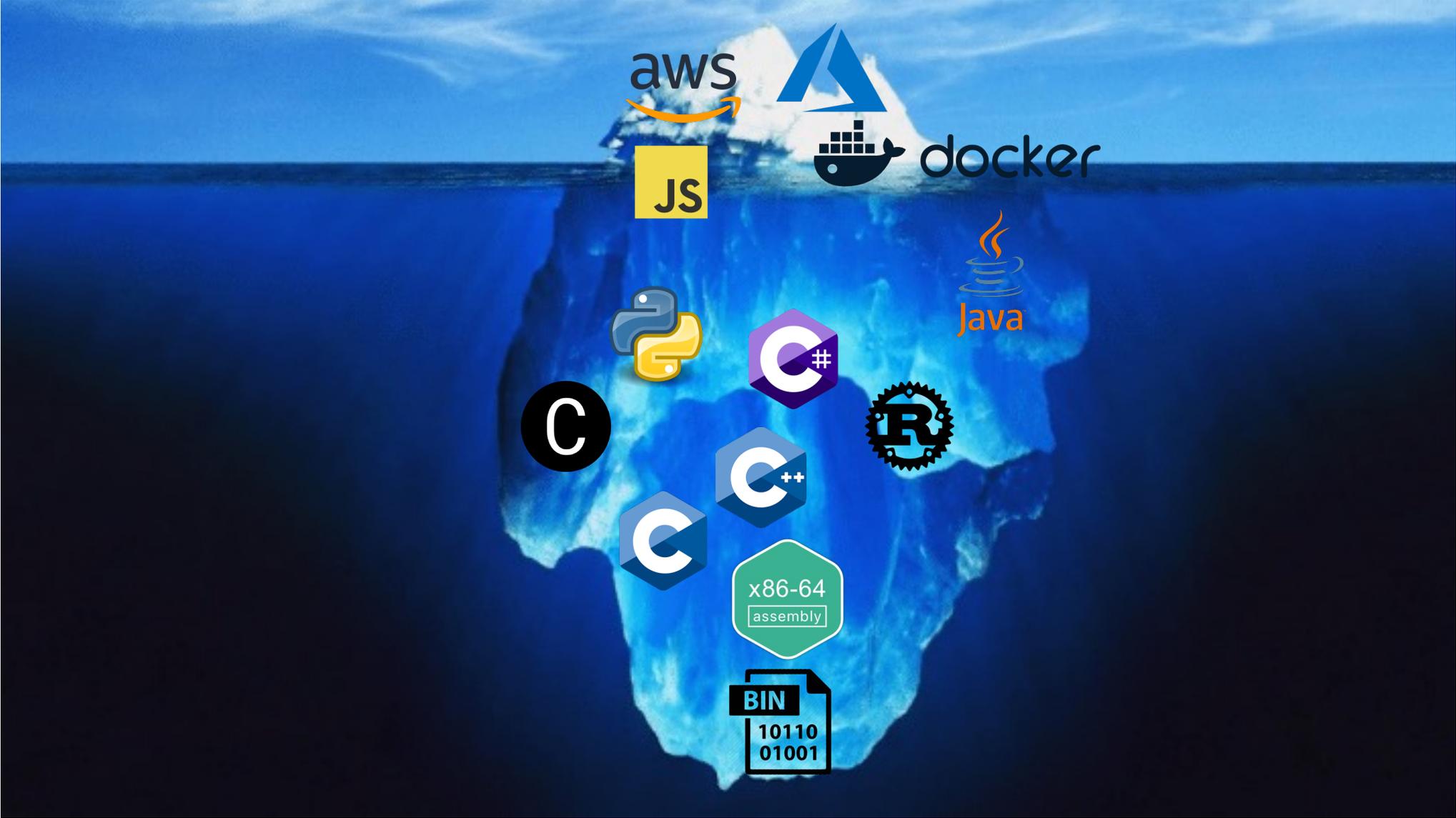


Infrastructures



Firmwares /  
Drivers





aws



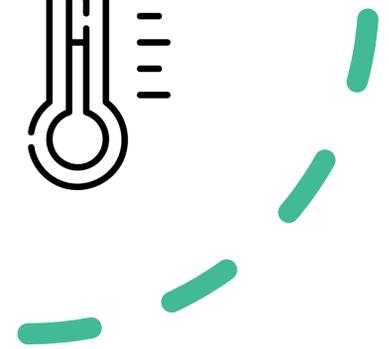
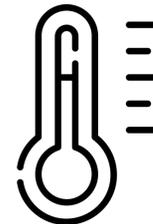
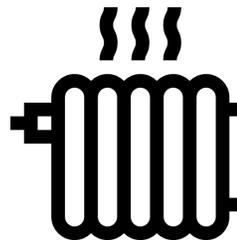
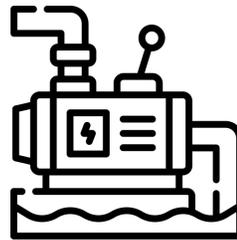
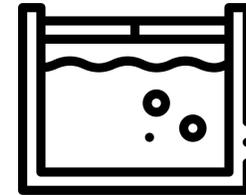
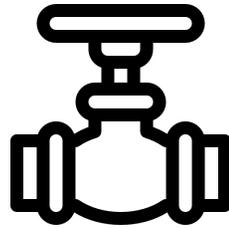
JS



docker



Exemples de dépendances matérielles



# Composants des systèmes embarqués

Microcontrôleur

Mémoire

Interfaces d'entrée/sortie

Capteurs / Actionneurs

Logiciel embarqué

Interface utilisateur

Alimentation

Connectivité

Communication

# Défis des Systèmes Embarqués dans le médical

## Besoins Spécifiques

- Rapidité
- Criticité (normes)
- Fiabilité critique
- Sécurité (usagers + données)
- Coûts
- Requis Physique (Taille/Poids)

## Limitations matérielles / logiciels

- Mémoire
- Performance, précision d'horloge
- Consommation énergétique
- RTOS vs OS
- Accès à des libraires (même les libraires standard)

## Considérations de design

- Utilisation de mémoire « *Heap / Stack* »
- Intégration avec des phénomènes physiques (Choc Thermique)

# Protocoles de communication



# C'est quoi?

Un ensemble de règles standardisées qui déterminent comment les données sont échangées entre les composants ou appareils, assurant une transmission **fiable** et **cohérente** des données.

Niveau logiciel	Niveau matériel
WebSockets	I2C
HTTP Requeste RestAPI	RS232
MessagesQueue	SPI
ZMQ	CAN

# Contraintes dans le domaine médical

Fiabilité

Systèmes robustes et  
tolérants au risque

Bande-passante et  
latence

Interopérabilité

# I2C



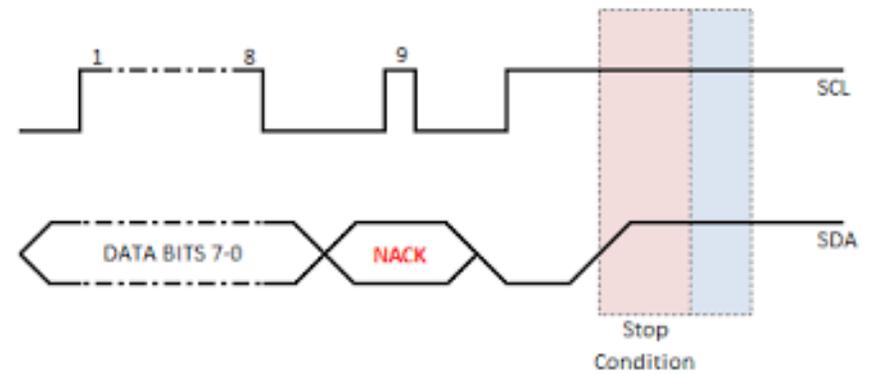
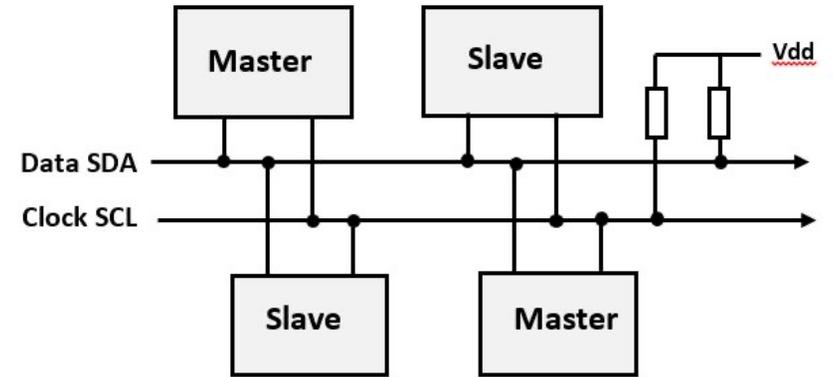
Communication  
Maître-Esclave



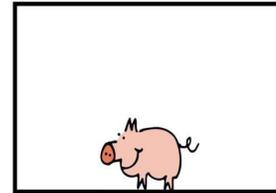
2 lignes de  
communication (SDA –  
SCL)



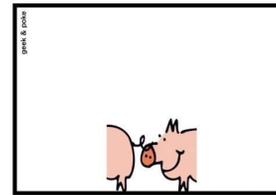
Bus Multi-Master



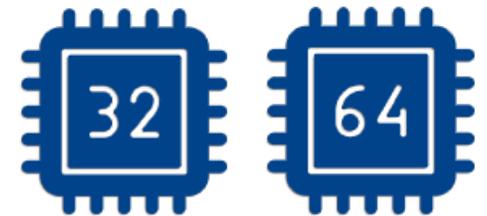
# Exemples de problématiques



BIG-ENDIAN



LITTLE-ENDIAN



# Gestion de mémoire



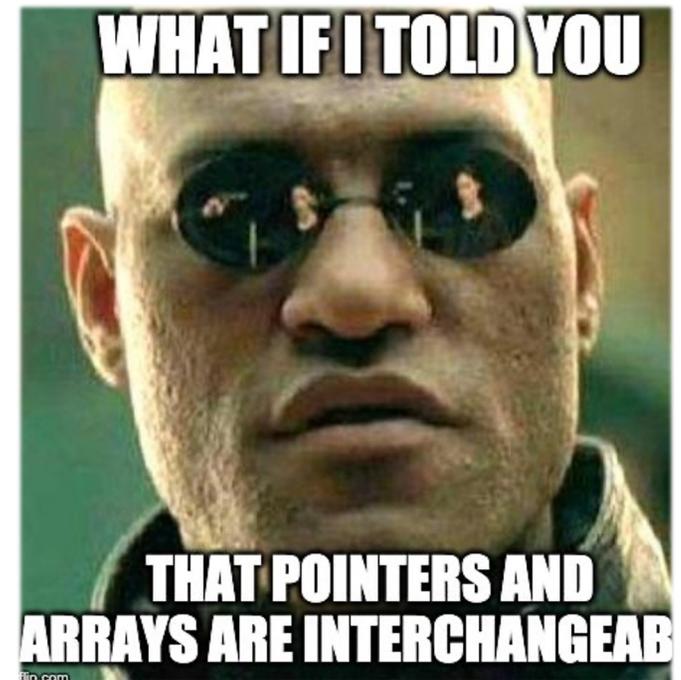
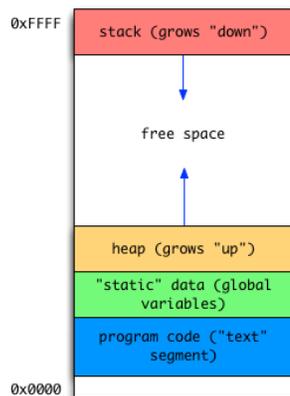
## Ce qui distingue la mémoire dans la prog. embarquée

- Différence de taille
  - Système embarqué: KB & MB
  - Serveur/Ordinateur: GB & TB
- Arrangement de la mémoire dépendant du compilateur
- Temps de réponse critique



# La memoire en rafale

- Pointeurs
  - Avantages: Efficacité & Flexibilité
  - Inconvénients: Complexité & Sécurité
- Mémoire stack vs mémoire heap
- Quel est la signification de “volatile” en C++ embarqué





# Problématiques

---

- Norme pour les système médicaux de limiter la taille de la heap.
- Certain choix d'architectures peuvent être très exigent pour la memoire.
  - Design patterns
  - Utilisation de certaines composantes
- Mauvaise analyse ou utilisation de librairie du système.
  - Pointeurs pour les singletons
  - ...

# Solutions

- Fichier .map qui donne des indications sur la mémoire du système
- Intégration des données de mémoire dans le processus de développement logiciel
- Bonne compréhension des concepts obscurs de C/C++



# Divers



# Ce qu'on aime de l'embarqué



C'est tangible, unique



Domaine en forte croissance technologique



Projets d'envergures, utiles et durables dans le temps



Dépend beaucoup des efforts à l'interne (- des librairies externes)

# Stages

- Projets des années antérieures
  - Outil de tag
  - Outil d'analyse de données de cycles
  - Outil pour afficher les métriques de projet
  - Automatisation de la validation de langue (OCR)
  - Outil de suivi de l'inventaire pour prévenir les ruptures
  - Intégration aux projets existants



# Activités à venir

---

- IFT-1111 (30 novembre)
- Cocktail de recrutement en TI (23 janvier 2024)





Questions?

---

