

ETUDE DE CAS

ARCHITECTURE FONCTIONNELLE



ARCHITECTURE MATERIEL

Chassis :

- 4 moteurs à courant continu
- ponts en H
- batterie et régulateur de tension

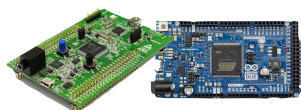


Interface :

- microphone

Calculateur :

- Arduino DUE
- STM32F4 Discovery



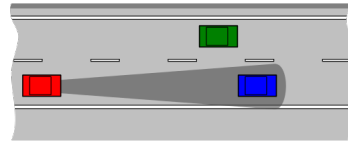
Mesure de distance :

- capteur ultrason

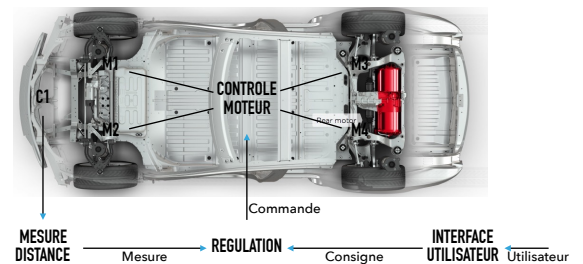


ETUDE DE CAS

- ▶ Adaptive Cruise Control
- ▶ Régulation de distance
- ▶ Issue du projet intégrateur L3 EEA

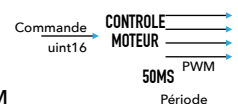


ARCHITECTURE FONCTIONNELLE

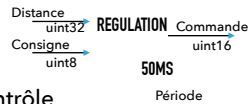


CONTROLE MOTEUR

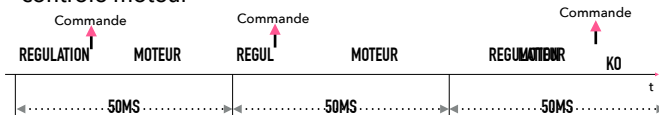
- ▶ Moteurs pilotés par signaux PWM
- ▶ 4 signaux distincts pour compenser défauts des moteurs
- ▶ Commande : vitesse d'avance (entier dans [0; 1023])
 - ▶ pas de changement de sens, ni d'orientation
- ▶ Signaux doivent être générés le plus régulièrement possible
- ▶ Doit s'exécuter au plus tôt après réception de la commande



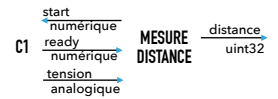
REGULATION



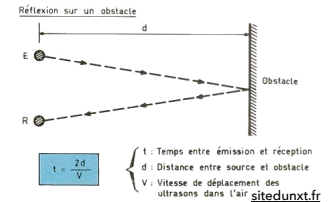
- ▶ Consigne : choix d'une loi de contrôle
- ▶ Commande : 0 -> vitesse nulle, 1023 -> vitesse max
- ▶ Mesure : distance entre capteur et obstacle
- ▶ Doit toujours produire une commande avant démarrage de contrôle moteur



MESURE DE DISTANCE



- ▶ start : démarrage d'une mesure sur capteur ultrason
- ▶ ready : signale mesure prête
- ▶ tension : expression de la distance par tension analogique
- ▶ Principe de la mesure :



INTERFACE UTILISATEUR



- ▶ Conversion d'une entrée de l'utilisateur en une consigne pour la régulation
- ▶ Interprétation d'un signal sonore pour faire évoluer une machine à état qui produit la consigne pour la régulation
- ▶ Décomposition en 3 fonctions élémentaires :



ECHANTILLONNAGE (SAMPLING)



- ▶ Un échantillonnage par secondes
- ▶ 100 échantillons par échantillonnage
- ▶ 100 μs entre chaque échantillon (doit être le plus régulier possible)



TRAITEMENT



- ▶ Traitement d'un échantillonnage
- ▶ Demande beaucoup de temps de calcul => ne doit pas bloquer les autres traitements (contrôle moteur, regulation,...)
- ▶ Une fréquence sonore interpréter en un ordre
- ▶ Traitement activé seulement si un nouvelle échantillonnage est disponible
- ▶ Principe DTMF : utilisation un couple de fréquence pour représenter un symbole (touche d'un téléphone). Ici, symbole sera associé à un ordre

	1 209 Hz	1 336 Hz	1 477 Hz	1 633 Hz
697 Hz	1	2	3	A
770 Hz	4	5	6	B
852 Hz	7	8	9	C
941 Hz	*	0	#	D

Wikipedia

EVOLUTION MACHINE A ETAT



- ▶ Chaque nouvel ordre fait évoluer la machine à état
- ▶ L'état détermine la consigne à transmettre à la régulation
- ▶ Evolution activée seulement si nouvelle ordre disponible

SPECIFICATION DE L'ARCHITECTURE LOGICIELLE

- ▶ Lister toutes les contraintes et dépendances entre les fonctions
- ▶ En déduire des contraintes d'ordonnancement
 - ▶ priorité, périodicité, précédence
- ▶ Etablir des séquences de fonctionnement correctes

t

TRAMPOLINE (OSEK/VDX)

- ▶ Système d'exploitation temps réel pour l'embarqué
- ▶ Compatible avec les normes OSEK/VDX et AUTOSAR
- ▶ Proposer une architecture logicielle pour TRAMPOLINE
 - ▶ Tâches, niveaux de priorité, messages, ressources partagées, alarmes, chaînage de tâches, interruptions, événements...

