



Raptor Dev



Apports de la formation :

- Présentation de la chaîne d'outils et des ECU⁽¹⁾
- Principes de génération de code
- Blocs de la librairie **Raptor dans simulink**
- Prérequis pour assurer la génération de code
- Architectures logicielles et règles de design
- Définition de la cible et choix des ECU
- Configuration du noyau Temps Réel
- Connexions aux E/S du calculateur ciblé
- Gestion des interruptions
- Ordonnancement des tâches
- Configuration des E/S
- Génération du code
- Programmation du code sur ECU

* **Option** : Personnalisation sur cas d'usage client.



Public cible :

- Développeur SW et prototypage de fonction en MBD⁽²⁾
- Connaissance logicielle de base :
 - Simulink, simulation, charge CPU, mémoire embarquée, fonction de transfert
- Connaissance de base en automatique : bloqueur, boucle de régulation, temps réel / temps continu



Format animation :

- 2 jours de formation
- Présentiel
- Théorique et pratique



Trame pédagogique :

- Un intervenant expert outil
- Présentation des différents blocs utilisables pour développer son application
- Présentation de la génération de code



Outils utilisés :

- Pc windows version 7 à 11 avec :
 - Mathworks Suite Release n-1
 - Matlab
 - Simulink
 - Stateflow
 - Matlab Coder
 - Simulink Coder
 - Embedded Coder
 - Raptor Dev
 - Raptor Cal
- Clé USB de licence (nécessaire pour la compilation et la désignation de cible)
- Interface KVASER
- ECU avec faisceau de programmation
- Alimentation électrique 12V



Doc support :

- Support de présentation
- Manuel de formation
- Banc de formation



Formation complémentaire :

- RaptorCal
- LIN/CAN
- Kvaser

(1) ECU : Electronic Control Unit
(2) MBD : Model Based Design