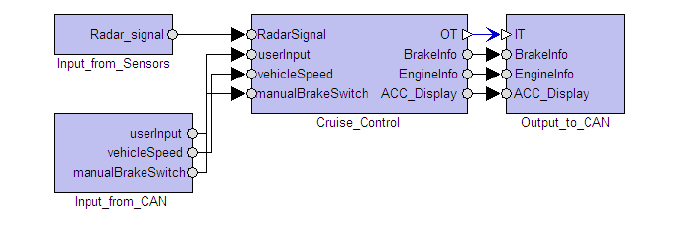
Recall that we have considered an entity which model the network behavior. This entity resolves the non-determinism on message arrivals to each actor. Actors can be either connected directly or via network. Currently, our network implements the behavior of the CAN BUS protocol. Thus, messages passed over the network are delivered from actors to CAN and from CAN to the actors. CAN applies the delay of a message communication over network based on the sender and receiver.

We have decided to give priority to data messages (e.g. most messages sent by physical actors) over other messages\*. These messages are of a setter type: upon serving such messages, its respective message server sets the respective discrete value of the receiving actor.

\* We wanted to map port behavior to our message queue. Intuition for our decision is that in ECUs when a control signal is simultaneous with some data signals, upon processing the control signal, The ECUs can access to data values by reading the ports. An Example of a ECU model is give bellow.



Our actors are executed concurrently until they reach to an action which is not executed in zero-time (i.e., delay statements, network communication delays, and execution of equations in physical actors). Due to non-deterministic execution of actors, they may be scenarios that such priority-based handling of messages are not preserved. Please see the example below.

Imagine we have 3 actor A, B and C.

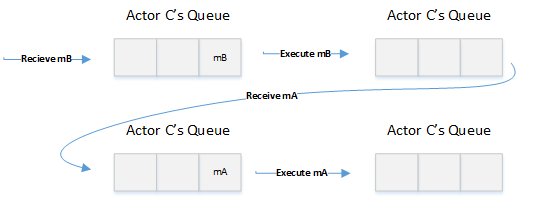
A wants to send message m­A to actor C.

B wants to send message m­B to actor C.

There is no message in actor C.

**Assume m­A is a data message and mB is not.**

Even if we give higher priority to pure data messages there are some scenarios that mB will be executed before m­A . A scenario is given bellow.



(The ideal scenario is that both messages are first received by actor C and after that actor C starts its execution.)