

## 2nd Home Assignment – Structural Modeling of Functional Units

Based on a meeting with the project customer, the lead system architect highlighted the most important functional components that we plan to design. As one of our design teams, you will be tasked with pilot projects involving a couple of important subsystems.

### Tasks

Extend the initial model by completing the following tasks:

- a. First, a simple warm-up task is related to the *Automated passenger counter (APC)* functionality. Based on the high-level requirements in the package *IntelliBus System/2 - Requirements/Automated Passenger Counting*, we have previously created an architecture overview in the package *IntelliBus System/4 - Functional Architecture/4.1 Functional decomposition/Automated Passenger Counting*, in the form of a *Block Definition Diagram (BDD)* and an *Internal Block Diagram (IBD)* of the *Autonomous Vehicle*. However, we have forgotten to create the traceability relationships between the requirements and design artifacts. The first task of the team is to investigate these packages, and reconstruct the appropriate traceability links. Extend the provided block diagram to visualize these links.
- b. The next task is the bottom-up design of an *intelligent intersection control (IIC)* subsystem, where an intersection control unit automatically controls pedestrian and vehicular traffic lights based on the input of pedestrian call buttons and vehicle sensors, as well as communicating with a central traffic management system. For realizing such intersections, we are planning on using the off-the-shelf components showcased in the vendor catalog BDD in *IntelliBus System/4 - Functional Architecture/4.1 Functional decomposition/Intelligent Intersection Control/Manufacturer Catalog*. The task is to realize the specific junction J42 using these components; where J42 is a simple T-intersection (a road coming from the west meets a north-to-south road), with each of the three directions having a single incoming vehicle lane and a pedestrian crossing. Please extend package *IntelliBus System/4 - Functional Architecture/4.1 Functional decomposition/Intelligent Intersection Control/J42* by completing the IBD of J42 (a block that specializes Intersection), specifying details of the intersection and each necessary component.
- c. The final task is the top-down design of the *traffic and route planning (TRP)* functionality, according to requirements provided in package *IntelliBus System/2 - Requirements/Traffic and Route Planning*. Please complete the pre-created BDD in package *IntelliBus System/4 - Functional Architecture/4.1 Functional decomposition/Traffic and Route Planning* with the definition of block types representing functional units that are necessary for satisfying the requirements. Specify the logical containment structure, associations, communication ports (with their types) and data flow connections between the new functional blocks on one or more IBDs (one such IBD is expected to depict the entire *IntelliBus System*, of course focusing on the TRP aspect only). Where meaningful and necessary, add subcomponents to the design such as user interfaces or databases even if not directly mentioned in the requirements; but do not yet overspecify implementation details. Make sure that the blocks correspond to functional units, not yet allocated to physical devices; in particular, it has not yet been decided whether autonomous vehicles select their next destination (bus stop or station) autonomously, or have to be instructed by a central intelligent traffic planner. Keep in mind that components or ports created for this task or previous tasks may be reused in several roles; make sure to use blocks or ports from the APC or IIS subsystems wherever applicable.
- d. (for extra IMSc credits) Specify non-trivial well-formedness constraints (design rules) for the system model (in relation to any of the three tasks) in English, Hungarian or in any formal constraint language (e.g. VIATRA QUERY, OCL).

Hint: we suggest going through the individual tasks in order. The reason is that the final task is the most complex one, and it will have to interconnect with the models that correspond to the first two tasks.