Benefits

**Improve System Level Performance**
Simulate and improve the dynamic behavior of multi-disciplinary systems. Easily model, simulate and validate smart systems where users can incorporate functions of sensing, actuation, and control coming from diverse components.

**Leverage Model-Based Development**
Altair Activate provides a common framework for functional product assessment and communication throughout the product development process. Perform what-if analyses at the system level to quickly test numerous designs and investigate the interactions of all components and subsystems comprising a system.

Altair Activate users can easily leverage the large library of Modelica physical components to further describe the plant and the controller.

Capabilities

**Build Diagrams Intuitively**
- Drag, drop and connect paradigm to rapidly construct models
- Multiple window configuration with the ability to modify diagrams between windows using the drag-and-drop and copy-and-paste operations
- Support for concurrent loading of multiple models in a session

**Hybrid Modeling**
Model and simulate continuous and discrete dynamic systems.

**Multi-disciplinary Modeling**
Altair Activate allows users to model and simulate the combined system behavior of real-world systems with support for multiple domains such as Mechanical, Electrical, and more.
Hierarchical and Parametric Modeling
- Build hierarchical component-based models of a real-world system using 1D block diagram modeling libraries
- Mix signal-based and physical modeling blocks in the same model
- When modeling large or complex systems, easily create super blocks by encapsulating multiple blocks in a diagram into a single block
- Super blocks are modular, reusable, can be masked, and fundamentally behave like regular blocks allowing more flexibility
- Since a model can be hierarchical and parameters can be defined at different levels, Altair Activate provides an ‘all available parameters’ option which lets users navigate in a diagram and get a report of all parameters that are known or defined at a current level

Built-in Block-based Model Libraries
Altair Activate includes a large variety of predefined blocks available in an easy-to-use library of palettes. Users can also create their own custom blocks in C or math scripts in OML and save them to new or existing libraries.

- Signal Generators
- Signal Viewers
- Signal Importers
- Signal Exporters
- Signal Conversions
- Signal Properties
- Math Operations
- Dynamic
- Hybrid
- Routing
- Logical Operations
- Activation Operations
- Matrix Operations
- Lookup Tables
- Ports
- Buffers
- Bus Operations
- Optimization
- Cosimulation
- FlipFlops
- Custom Blocks

Physical Component Modeling Using Modelica and SPICE
In addition to the signal-based blocks listed above, Altair Activate comes with the Modelica standard library (MSL) – a collection of blocks describing the physical behavior of Electrical, Electromagnetic, Mechanical, Thermal components. These blocks can be extended by user-defined Modelica blocks. Furthermore, users can provide SPICE netlists to model electrical circuits.

Library Management
Easily create components and assemble custom applications. Use Altair Activate’s library manager to create and edit custom libraries. Altair Activate also provides an IDE along with API functions for users to further leverage library management.

Hybrid Simulator
Altair Activate’s simulator provides users with several high-performance numerical solvers that accurately and robustly solve dynamic systems including continuous, discrete-time, and event-based behaviors.

<table>
<thead>
<tr>
<th>Solver Type</th>
<th>Stiffness</th>
<th>Solver Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixed step-size</td>
<td>Non-stiff ODE</td>
<td>Forward Euler</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Explicit Trapezoidal</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Classical Runge-Kutta</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Runge-Kutta</td>
</tr>
<tr>
<td>Stiff ODE</td>
<td></td>
<td>Backward Euler</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Implicit Trapezoidal</td>
</tr>
<tr>
<td>Non-stiff ODE</td>
<td></td>
<td>CVODE-BDF-Functional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CVODE-ADAMS-Functional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DOPRI (Dormand-prince)</td>
</tr>
<tr>
<td>Variable step-size</td>
<td>Stiff ODE</td>
<td>LSode</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CVODE-BDF-NEWTON</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CVODE-ADAMS-NEWTON</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RADAU-IIA for ODE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CVODE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DAE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IDA RADAU-IIA for DAE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DASKIR</td>
</tr>
</tbody>
</table>

Optimization
Formulate optimization problems to improve the system parameters and design robust control strategies with multiple options.

Graphical optimization tool:
- The simplest way to formulate and solve optimization problems

Script-based optimization:
- A powerful mechanism for solving general optimization problems where the cost and constraints may be obtained from a combination of Altair Activate simulation results and math scripts

BOBYA Optimizer block:
- This optimization block can be used directly in a model and doesn’t require any external calling function or link-up
- Cascade multiple optimization blocks to formulate max-min and min-max problems

Model Exchange and Co-simulation via Functional Mock-up Interface (FMI)
Altair Activate supports FMI 2.0 standard for both model exchange and co-simulation of dynamic systems including the ability to import and export FMUs (Functional Mock-up Units).

Co-simulation with Multi-body Models
The co-simulation interface lets users simulate a complex system that includes a multi-body system (MBS) and one or more control subsystems. In order to effectively simulate the entire system, the MBS is simulated with a multi-body simulation solver while the control subsystem is simulated with Altair Activate.

Linearization
Altair Activate allows users to create linear models from Altair Activate blocks by linearization. The operating point can be computed either by running the simulation at a given time instant or by computing a steady-state point by imposing constraints on inputs, outputs, states and state derivatives.

Compiling models into executable code
Altair Activate supports code generation for system performance & IP protection.