

Latest Features in Simulink Coder

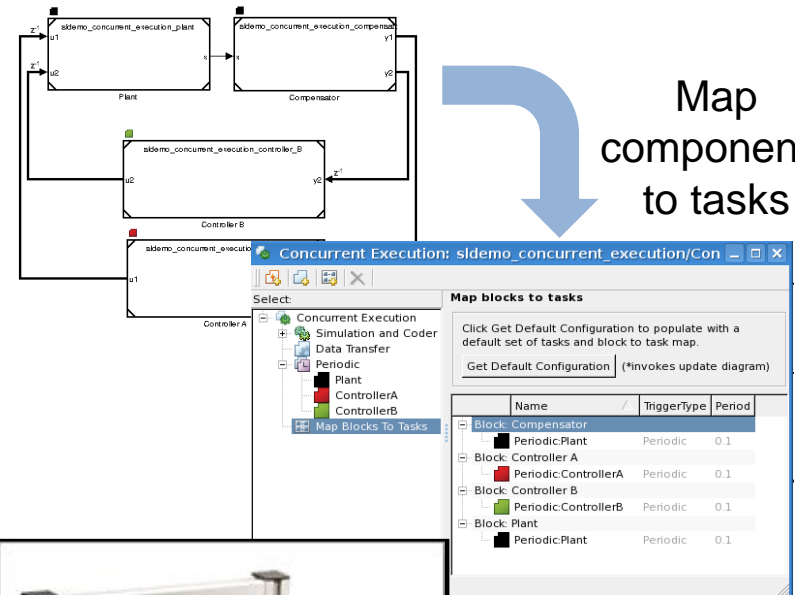
September 2011

R2011b

Multicore Support for xPC Target and Desktop Simulation

Leverage multiple cores to enable faster execution

- Specify tasks that can run concurrently on multicore processors
- Configure task-to-task communication
- Choose to enable multicore support for xPC Target, POSIX threads (Pthreads), or Windows threads

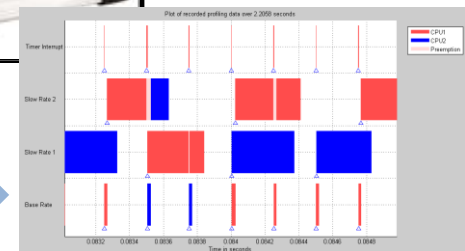


Map components to tasks



Generate code for multicore xPC machine

Profile task execution time

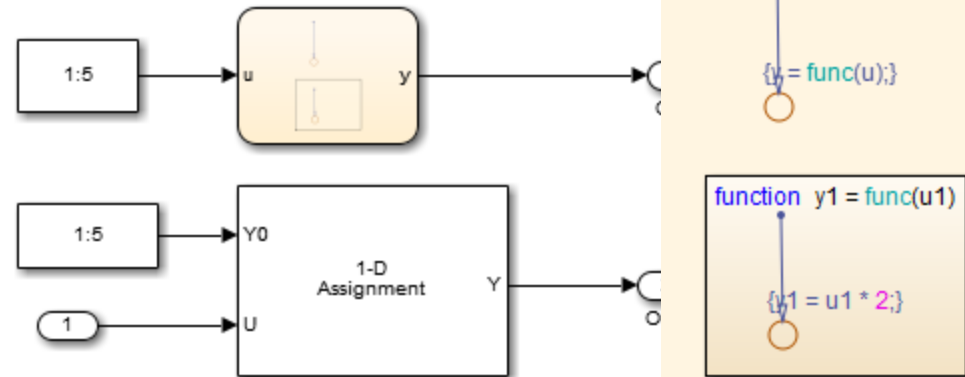


» sldemo_concurrent_execution

Improved Parameter Pooling for Stateflow and MATLAB Functions

Reduce RAM and ROM consumption by pooling more parameter data

- Parameter pooling includes matrix constant parameter associated with Simulink, Stateflow, and MATLAB function input arguments.



Stateflow declares separate memory from Simulink

```
static real_T tmp[5] = {
    1.0, 2.0, 3.0, 4.0, 5.0 };

func(tmp, rtY.Out1);

for (i = 0; i < 5; i++) {
    rtY.Out3[i] =
        rtConstP.pooled1[i];
}
```

R2011a

Stateflow pools memory with Simulink

```
func(rtConstP.pooled1,
    rtY.Out1);

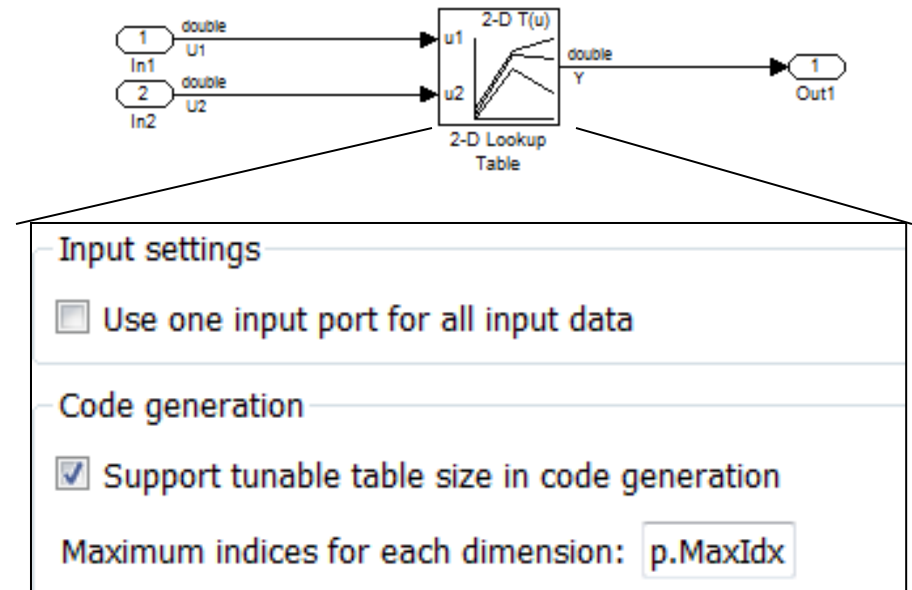
for (i = 0; i < 5; i++) {
    rtY.Out3[i] =
        rtConstP.pooled1[i];
}
```

R2011b

Tunable Lookup Table Size in Generated Code

Support calibration workflows by enabling table size tuning in generated code

- Lookup table size in code generation adjustable at link time or startup and background-only operations.



Tunable table data and size

```
void output(void)
{
    Y = look2_binlcpw(U1, U2, p.BP1, p.BP2, p.Table, p.MaxIdx, p.MaxIdx[0] + 1U);
}
```