Timing info tab in BB Control

PN156 | Posted on March 29, 2021 | Updated on May 7, 2025

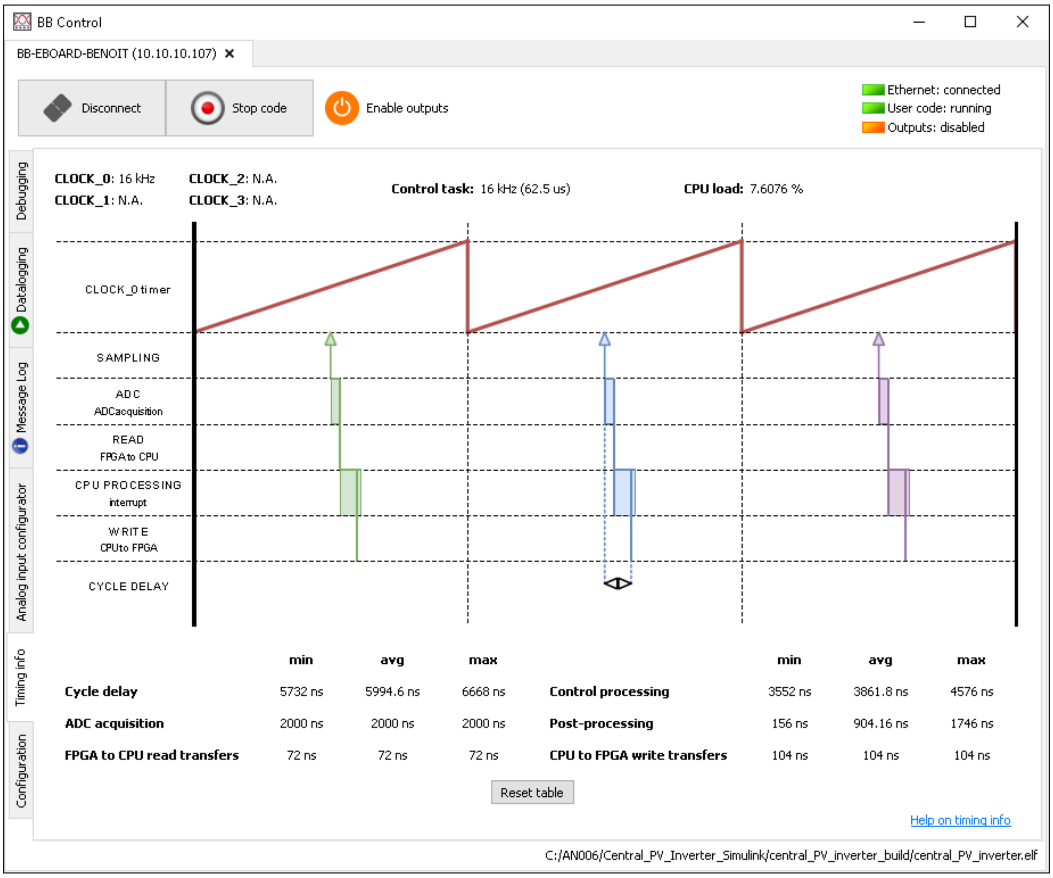
Benoît STEINMANN
Software Team Leader
imperix • in

Discontinuation Notice
From release v.3.7 of both ACG SDK and CPP SDK, BB control has been discontinued. It has been replaced by imperix Cockpit which offers an extended set of features and a superior user experience. The same timing information can be found in Cockpit. To learn more please visit the Cockpit user guide page.

The BB Control *Timing info* tab provides a graphical representation of the various computation and communication delays involved in the B-Board PRO and B-Box RCP power electronics controllers during run-time.

It is particularly useful to observe the delays involved in the control dynamics of the system as explained in Identifying the discrete control delay (PN142) and adjust the control parameters accordingly, such as the K_p and K_i of a PI control implementation as shown in Basic PI control implementation (TN105).

Below are screenshots of the *Timing info* tab when running the model Central PV inverter (AN006) in two scenarios.



🟩 Ethernet: connected
🟩 User code: running
🟧 Outputs: disabled

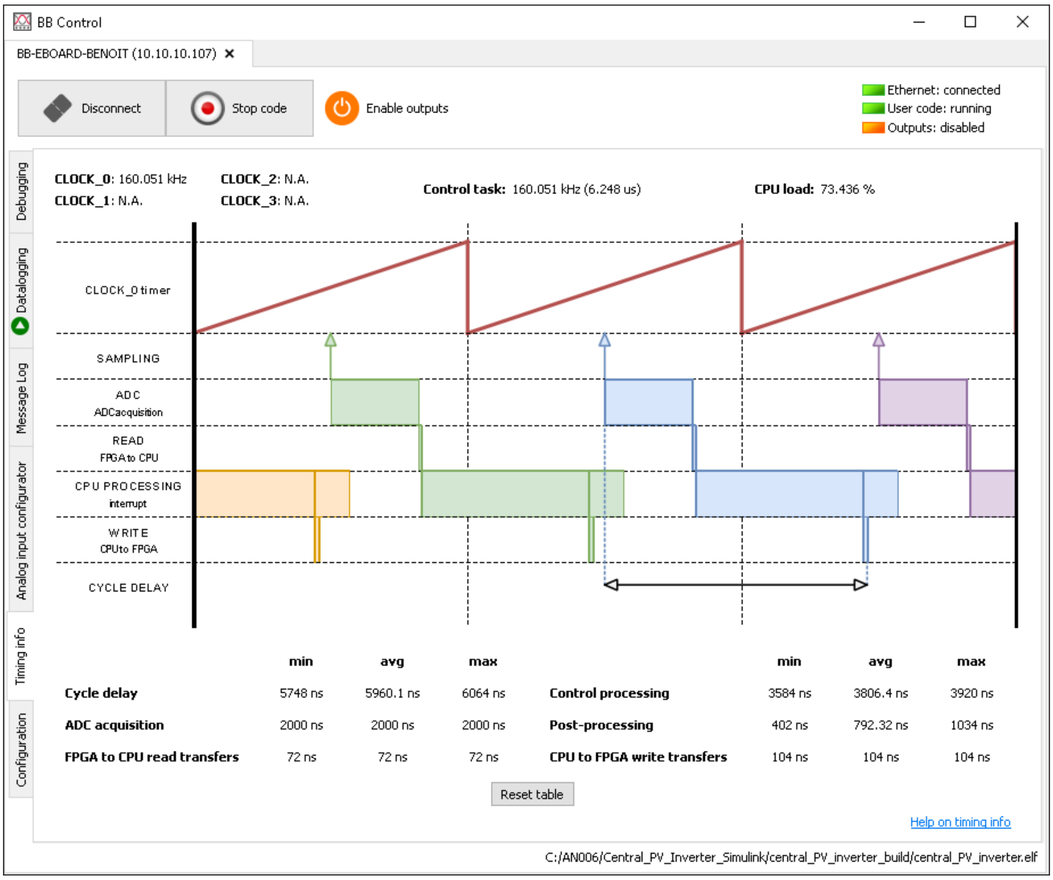
CLOCK_0: 16 kHz **CLOCK_2**: N.A.
CLOCK_1: N.A. **CLOCK_3**: N.A.

Control task: 16 kHz (62.5 us) **CPU load:** 7.6076 %

	min	avg	max		min	avg	max
Cycle delay	5732 ns	5994.6 ns	6668 ns	**Control processing**	3552 ns	3861.8 ns	4576 ns
ADC acquisition	2000 ns	2000 ns	2000 ns	**Post-processing**	156 ns	904.16 ns	1746 ns
FPGA to CPU read transfers	72 ns	72 ns	72 ns	**CPU to FPGA write transfers**	104 ns	104 ns	104 ns

C:/AN006/Central_PV_Inverter_Simulink/central_PV_inverter_build/central_PV_inverter.elf

Central PV inverter running at 16 kHz



🟩 Ethernet: connected
🟩 User code: running
🟧 Outputs: disabled

CLOCK_0: 160.051 kHz **CLOCK_2**: N.A.
CLOCK_1: N.A. **CLOCK_3**: N.A.

Control task: 160.051 kHz (6.248 us) **CPU load:** 73.436 %

	min	avg	max		min	avg	max
Cycle delay	5748 ns	5960.1 ns	6064 ns	**Control processing**	3584 ns	3806.4 ns	3920 ns
ADC acquisition	2000 ns	2000 ns	2000 ns	**Post-processing**	402 ns	792.32 ns	1034 ns
FPGA to CPU read transfers	72 ns	72 ns	72 ns	**CPU to FPGA write transfers**	104 ns	104 ns	104 ns

C:/AN006/Central_PV_Inverter_Simulink/central_PV_inverter_build/central_PV_inverter.elf

Central PV inverter running at 160 kHz

Clock generators, control task routine, and CPU load

At the top of the window, the following information are shown:

CLOCK_0, **CLOCK_1**, **CLOCK_2,** and **CLOCK_3** show the configured frequency of the 4 Clock generator.

The **Control task** routine execution frequency and period are also displayed. Note that the control task is always mapped on *CLOCK_0.*

The **CPU load** represents how much time the CPU spends in the interrupt routine. Safety mechanisms are implemented to detect CPU overload. An overload can result from a control algorithm too complex or an execution frequency too high.

The timing graph

The timing graph accurately represents the delays involved in the execution of the user control code.

CLOCK_0 timer represents the value of the clock generator counter. *CLOCK_0* is used as the time-base for the sampling events and the control task routine execution. In most cases, the PWM modulators are also based on *CLOCK_0*.

SAMPLING shows the sampling events, i.e. the instants where the ADCs sample the input analog signals. In the example above, the sampling phase is set 0.5 so the sampling occurs in the middle of the period of the PWMs that are based on *CLOCK_0*. The oversampling enables the possibility to set up multiple sampling events between each control task execution as explained in Oversampling (PN154).

ADC acquisition shows the delay between a sampling event and the availability of the values in the FPGA. It comprises the analog-to-digital delay and the results transfer to the FPGA delay. This value can be set to 2000 ns (B-Box RCP or B-Board PRO) or to 500 ns (B-Board PRO).

FPGA-to-CPU transfers are executed right after the ADC results are available in the FPGA. The values transferred are typically the ADC measurements, the GPI values, or the angle decoder output.

CPU PROCESSING shows the time the CPU spends executing the interrupt routine. To execute the *CPU-to-FPGA transfers* as early as possible (and reducing the overall response delay), the interrupt routine is separated into two phases:

- The **control task execution** is the delay to execute the user control model and update the modulation parameters and other FPGA values. The *CPU-to-FPGA transfers* are executed right after, in parallel with the *post-processing execution*.
- The **post-processing execution** is the delay necessary to perform all the tasks that are not directly involved in the control algorithm. It includes the datalogging execution, the CAN communication, the Simulink external mode execution, etc.

CPU-to-FPGA transfers are performed once the *control task execution* is over. The values transferred are typically the PWM modulation parameters (duty-cycle, phase), the GPO values, or the DAC values.

CYCLE DELAY represents the delay between the sampling event and when the newly computed data are available in FPGA (*ADC acquisition + FPGA-to-CPU transfers + control task execution + CPU-to-FPGA transfers*). This value can be used by the user to compute the *control delay* and the *total loop delay*, as explained in Identifying the discrete control delay (PN142).

The measurements table

The table at the bottom of the window provides the values used to generate the timing graph *(avg),* and a saves the *min* and *max* of each value. The *reset table* button resets the values of the table.

The **ADC acquisition**, the **FPGA to CPU read transfers,** and **CPU to FPGA write transfers** are fixed values computed at code launch.

The **control processing** and **post-processing** values are measured from the CPU using the CPU physical counter.

The **cycle delay** value is precisely measured directly from within the FPGA.