Selecting a V_{DD} value

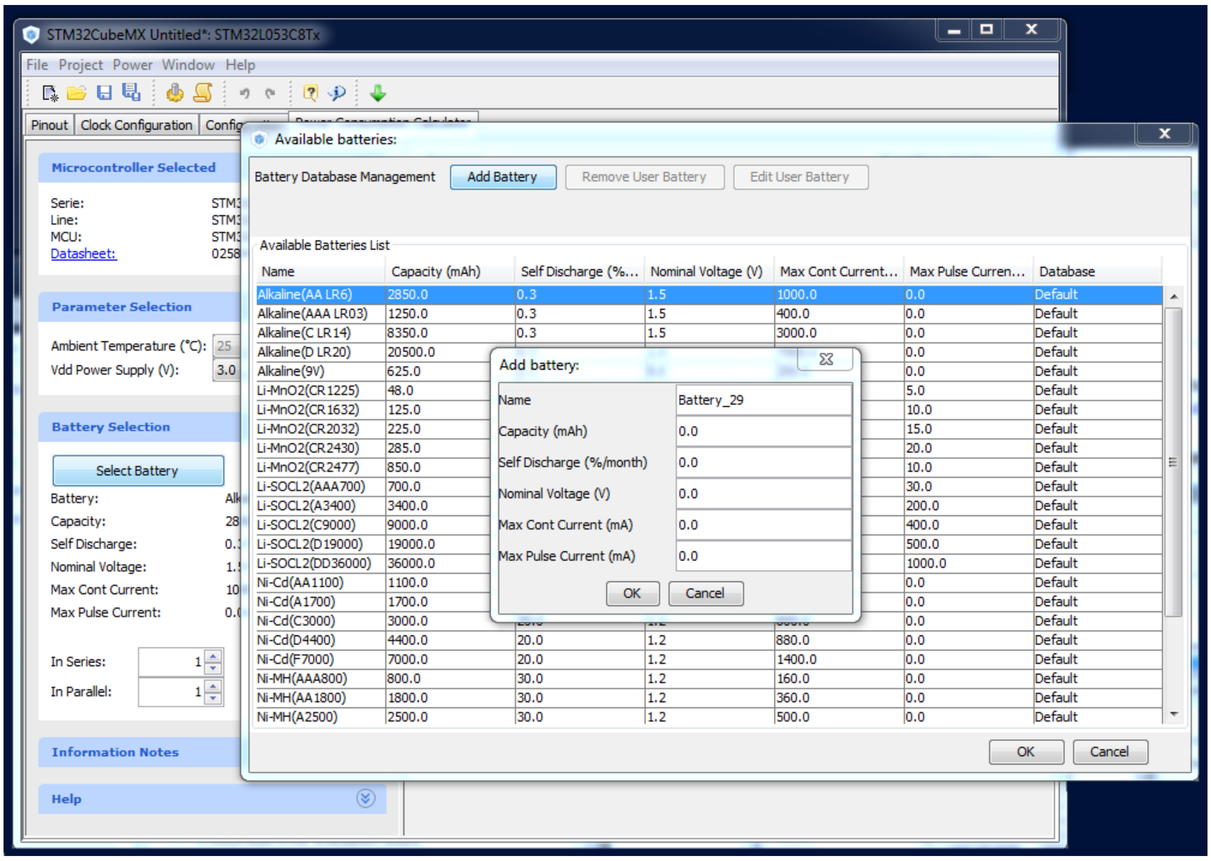
From this view and when multiple choices are available, the user must select a V_{DD} value.

Selecting a battery model (optional)

Optionally, the user can select a battery model. This can also be done once the power consumption sequence is configured.

The user can select a predefined battery or choose to specify a new battery that best matches his application (see *Figure 93*).

Figure 93. Battery selection



Serie: STM3
Line: STM3
MCU: STM3
Datasheet: 0258

Ambient Temperature (°C): 25
Vdd Power Supply (V): 3.0

Battery: Alk
Capacity: 28
Self Discharge: 0.
Nominal Voltage: 1.
Max Cont Current: 10
Max Pulse Current: 0.

In Series: 1
In Parallel: 1

Battery Database Management

Available Batteries List

Name	Capacity (mAh)	Self Discharge (%...	Nominal Voltage (V)	Max Cont Current...	Max Pulse Curren...	Database
Alkaline(AA LR6)	2850.0	0.3	1.5	1000.0	0.0	Default
Alkaline(AAA LR03)	1250.0	0.3	1.5	400.0	0.0	Default
Alkaline(C LR14)	8350.0	0.3	1.5	3000.0	0.0	Default
Alkaline(D LR20)	20500.0	[illegible]	[illegible]	[illegible]	0.0	Default
Alkaline(9V)	625.0	[illegible]	[illegible]	[illegible]	0.0	Default
Li-MnO2(CR1225)	48.0	[illegible]	[illegible]	[illegible]	5.0	Default
Li-MnO2(CR1632)	125.0	[illegible]	[illegible]	[illegible]	10.0	Default
Li-MnO2(CR2032)	225.0	[illegible]	[illegible]	[illegible]	15.0	Default
Li-MnO2(CR2430)	285.0	[illegible]	[illegible]	[illegible]	20.0	Default
Li-MnO2(CR2477)	850.0	[illegible]	[illegible]	[illegible]	10.0	Default
Li-SOCL2(AAA700)	700.0	[illegible]	[illegible]	[illegible]	30.0	Default
Li-SOCL2(A3400)	3400.0	[illegible]	[illegible]	[illegible]	200.0	Default
Li-SOCL2(C9000)	9000.0	[illegible]	[illegible]	[illegible]	400.0	Default
Li-SOCL2(D19000)	19000.0	[illegible]	[illegible]	[illegible]	500.0	Default
Li-SOCL2(DD36000)	36000.0	[illegible]	[illegible]	[illegible]	1000.0	Default
Ni-Cd(AA1100)	1100.0	[illegible]	[illegible]	[illegible]	0.0	Default
Ni-Cd(A1700)	1700.0	[illegible]	[illegible]	[illegible]	0.0	Default
Ni-Cd(C3000)	3000.0	[illegible]	[illegible]	[illegible]	0.0	Default
Ni-Cd(D4400)	4400.0	20.0	1.2	880.0	0.0	Default
Ni-Cd(F7000)	7000.0	20.0	1.2	1400.0	0.0	Default
Ni-MH(AAA800)	800.0	30.0	1.2	160.0	0.0	Default
Ni-MH(AA1800)	1800.0	30.0	1.2	360.0	0.0	Default
Ni-MH(A2500)	2500.0	30.0	1.2	500.0	0.0	Default

Add battery:

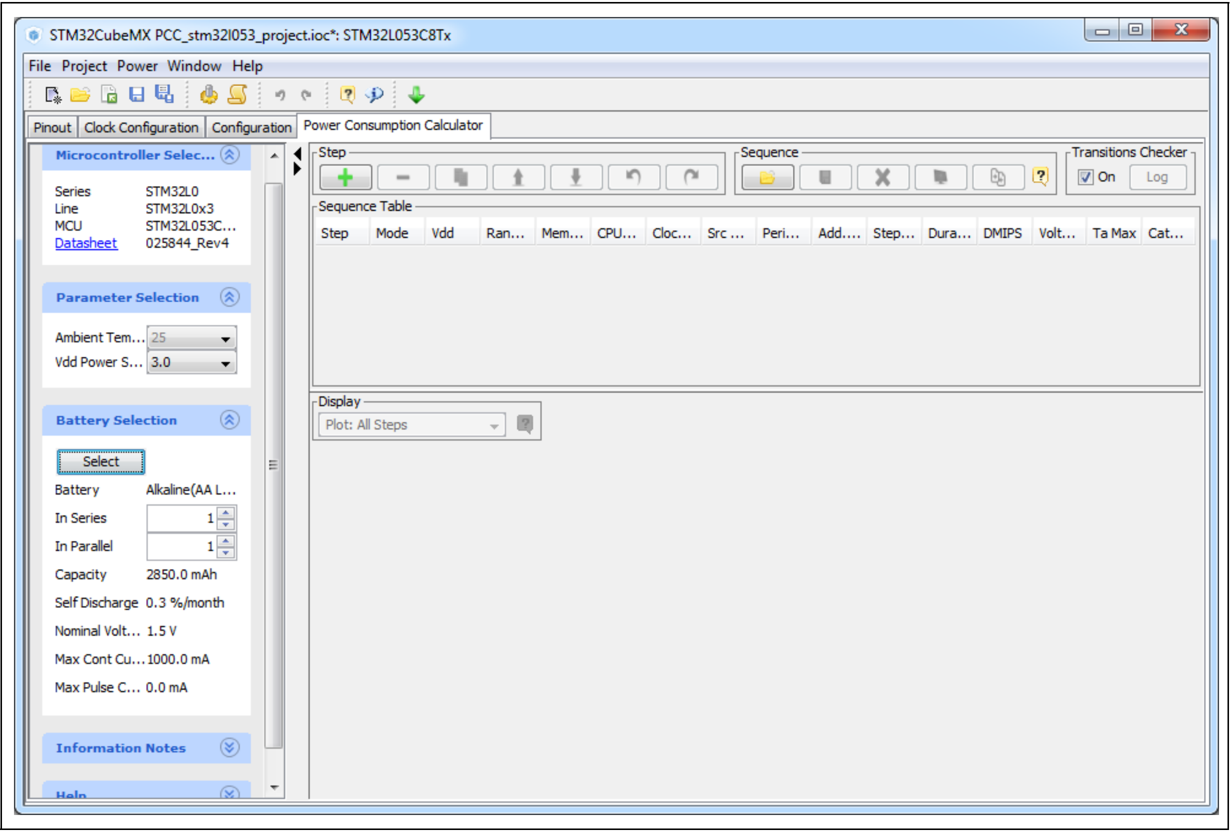
Name	Battery_29
Capacity (mAh)	0.0
Self Discharge (%/month)	0.0
Nominal Voltage (V)	0.0
Max Cont Current (mA)	0.0
Max Pulse Current (mA)	0.0

OK Cancel

Power sequence default view

The user can now proceed and build a power sequence (see *Figure 94*).

Figure 94. Building a power consumption sequence



Series	STM32L0
Line	STM32L0x3
MCU	STM32L053C...
Datasheet	025844_Rev4

Ambient Tem...	25
Vdd Power S...	3.0

Battery	Alkaline(AA L...
In Series	1
In Parallel	1
Capacity	2850.0 mAh
Self Discharge	0.3 %/month
Nominal Volt...	1.5 V
Max Cont Cu...	1000.0 mA
Max Pulse C...	0.0 mA

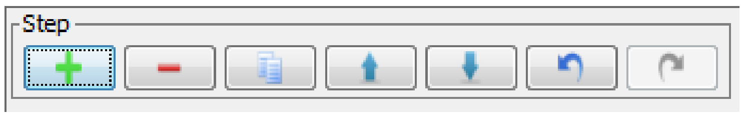
Step	Mode	Vdd	Ran...	Mem...	CPU...	Cloc...	Src ...	Peri...	Add....	Step...	Dura...	DMIPS	Volt...	Ta Max	Cat...

Managing sequence steps

Steps can be reorganized within a sequence (**Add** new, **Delete** a step, **Duplicate** a step, move **Up** or **Down** in the sequence) using the set of Step buttons (see *Figure 95*).

The user can undo or redo the last configuration actions by clicking the **Undo** button in the PCC view or the Undo icon from the main toolbar

Figure 95. Step management functions



Adding a step

There are two ways to add a new step:

- Click **Add** in the Power Consumption panel. The **New Step** window opens with empty step settings.
- Or, select a step from the sequence table and click **Duplicate**. A **New Step** window opens duplicating the step settings. (see *Figure 96*).

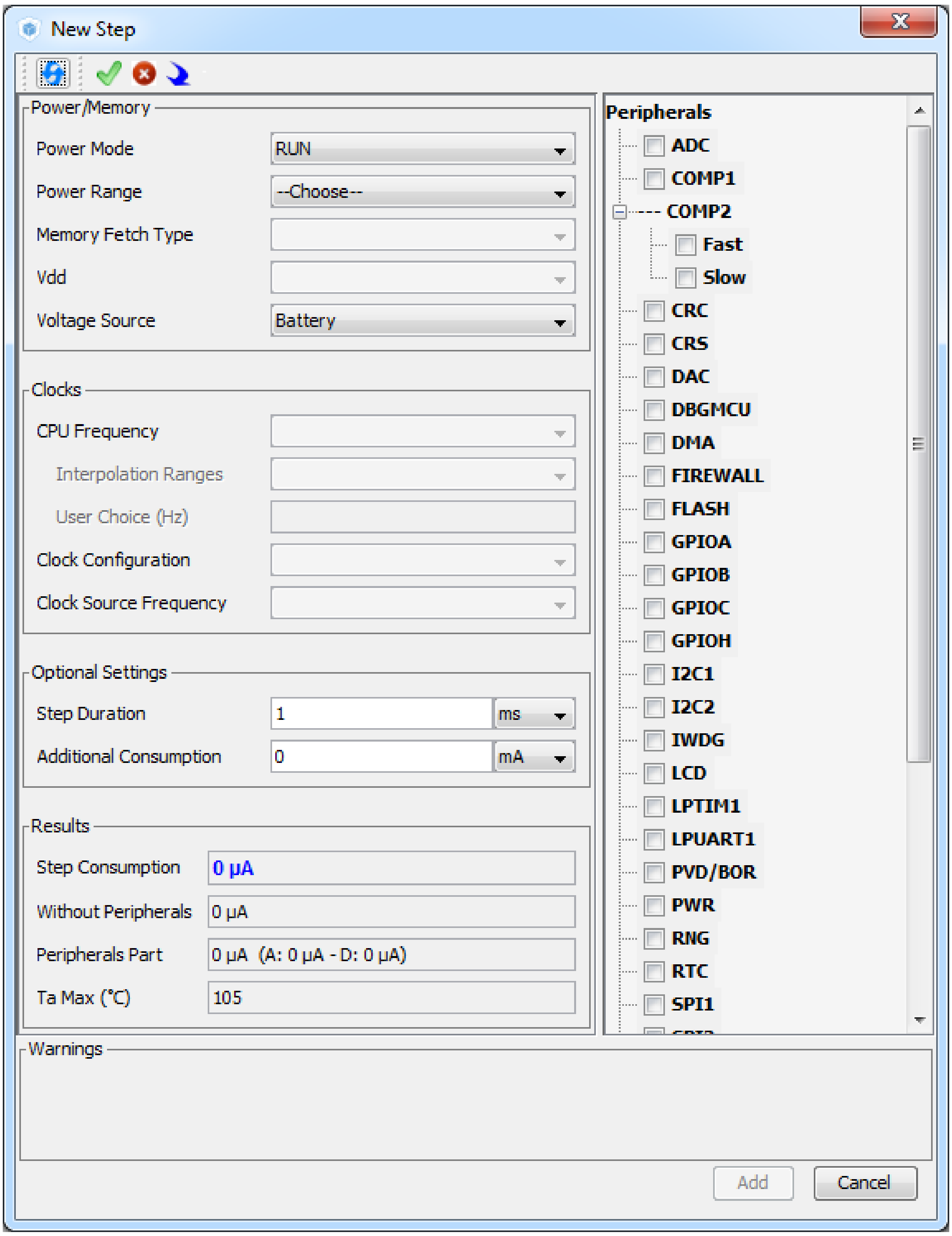


Figure 96. Power consumption sequence: new step default view

New Step

Power/Memory

Power Mode	RUN
Power Range	--Choose--
Memory Fetch Type	
Vdd	
Voltage Source	Battery

Clocks

CPU Frequency	
Interpolation Ranges	
User Choice (Hz)	
Clock Configuration	
Clock Source Frequency	

Optional Settings

Step Duration	1	ms
Additional Consumption	0	mA

Results

Step Consumption	0 µA
Without Peripherals	0 µA
Peripherals Part	0 µA (A: 0 µA - D: 0 µA)
Ta Max (°C)	105

Peripherals
- [ ] ADC
- [ ] COMP1
- COMP2
  - [ ] Fast
  - [ ] Slow
- [ ] CRC
- [ ] CRS
- [ ] DAC
- [ ] DBGMCU
- [ ] DMA
- [ ] FIREWALL
- [ ] FLASH
- [ ] GPIOA
- [ ] GPIOB
- [ ] GPIOC
- [ ] GPIOH
- [ ] I2C1
- [ ] I2C2
- [ ] IWDG
- [ ] LCD
- [ ] LPTIM1
- [ ] LPUART1
- [ ] PVD/BOR
- [ ] PWR
- [ ] RNG
- [ ] RTC
- [ ] SPI1

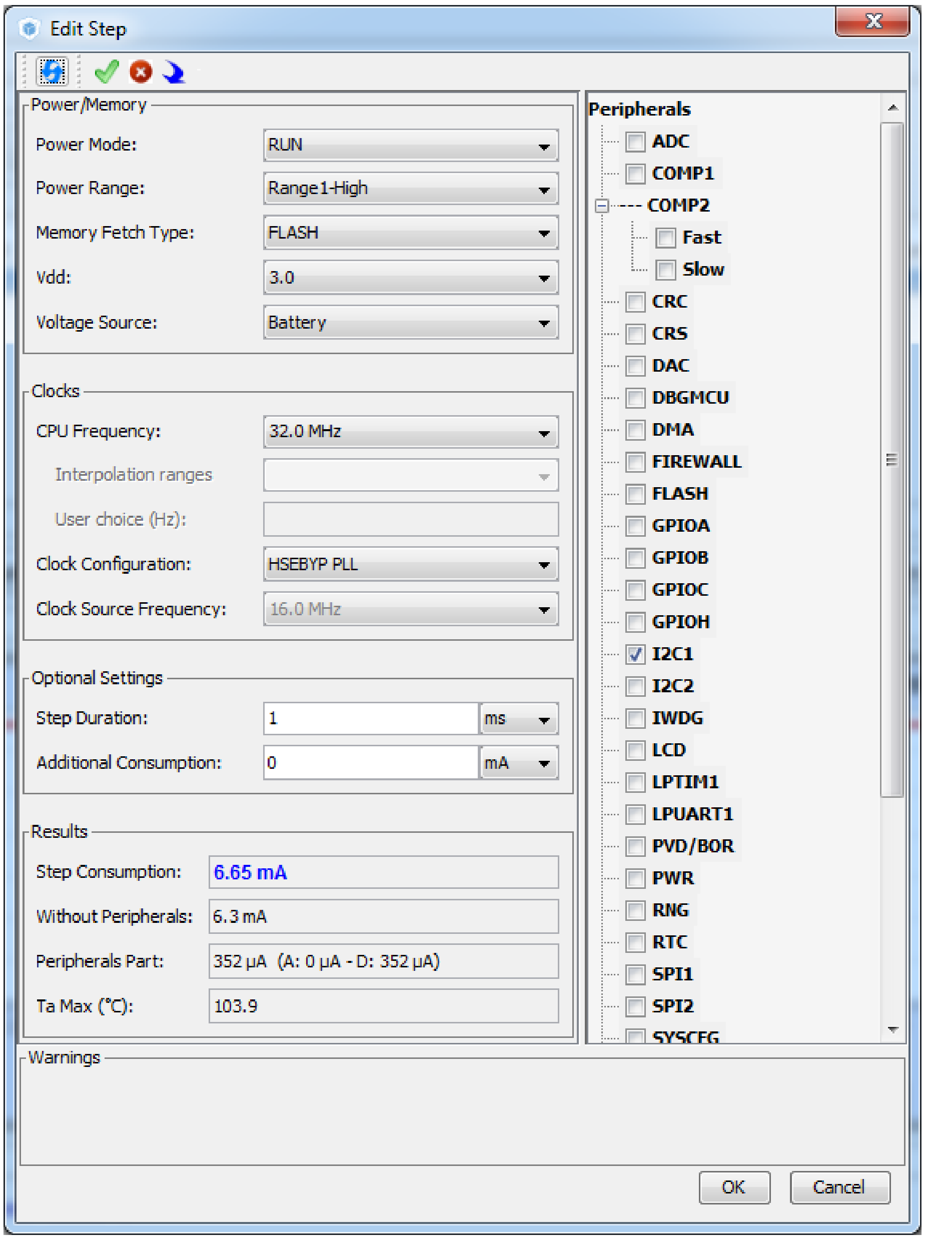
Warnings

Once a step is configured, resulting current consumption and T_{AMAX} values are provided in the window.

Editing a step

To edit a step, double-click it in the sequence table. The **Edit Step** window opens (see *Figure 97*).

Figure 97. Edit Step window



Power/Memory

Power Mode:	RUN
Power Range:	Range1-High
Memory Fetch Type:	FLASH
Vdd:	3.0
Voltage Source:	Battery

Clocks

CPU Frequency:	32.0 MHz
Interpolation ranges	
User choice (Hz):	
Clock Configuration:	HSEBYP PLL
Clock Source Frequency:	16.0 MHz

Optional Settings

Step Duration:	1	ms
Additional Consumption:	0	mA

Results

Step Consumption:	6.65 mA
Without Peripherals:	6.3 mA
Peripherals Part:	352 µA (A: 0 µA - D: 352 µA)
Ta Max (°C):	103.9

Warnings

Peripherals
- ☐ ADC
- ☐ COMP1
- COMP2
  - ☐ Fast
  - ☐ Slow
- ☐ CRC
- ☐ CRS
- ☐ DAC
- ☐ DBGMCU
- ☐ DMA
- ☐ FIREWALL
- ☐ FLASH
- ☐ GPIOA
- ☐ GPIOB
- ☐ GPIOC
- ☐ GPIOH
- ✓ I2C1
- ☐ I2C2
- ☐ IWDG
- ☐ LCD
- ☐ LPTIM1
- ☐ LPUART1
- ☐ PVD/BOR
- ☐ PWR
- ☐ RNG
- ☐ RTC
- ☐ SPI1
- ☐ SPI2
- ☐ SYSCFG

Moving a step

By default, a new step is added at the end of a sequence.

Click the step in the sequence table to select it and use the **Up** and **Down** buttons to move it elsewhere in the sequence.

Deleting a step

Select the step to be deleted and click the **Delete** button.

Using the transition checker

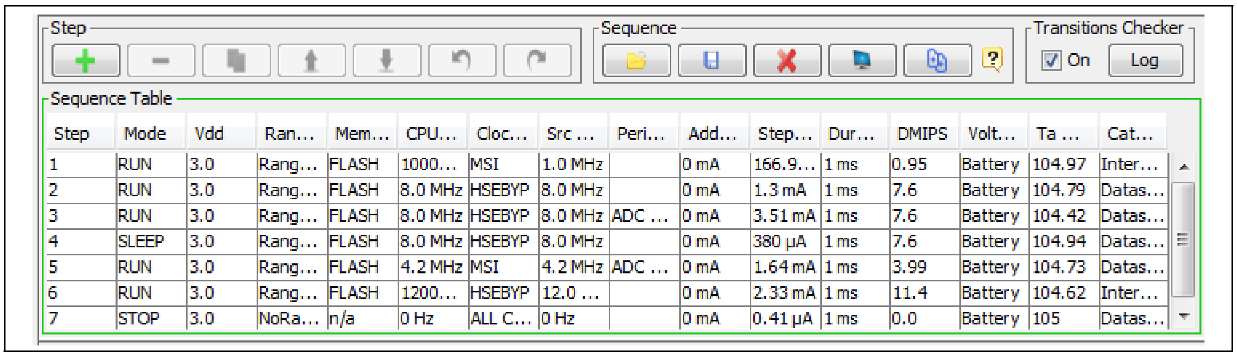
Not all transitions between power modes are possible. PCC proposes a transition checker to detect invalid transitions or restrict the sequence configuration to only valid transitions.

Enabling the transition checker option prior to sequence configuration ensures the user will be able to select only valid transition steps.

Enabling the transition checker option on an already configured sequence will highlight the sequence in green (green frame) if all transitions are valid (see *Figure 98*), or in red if at least one transition is invalid (red frame with description of invalid step highlighted in red) (see *Figure 99*).

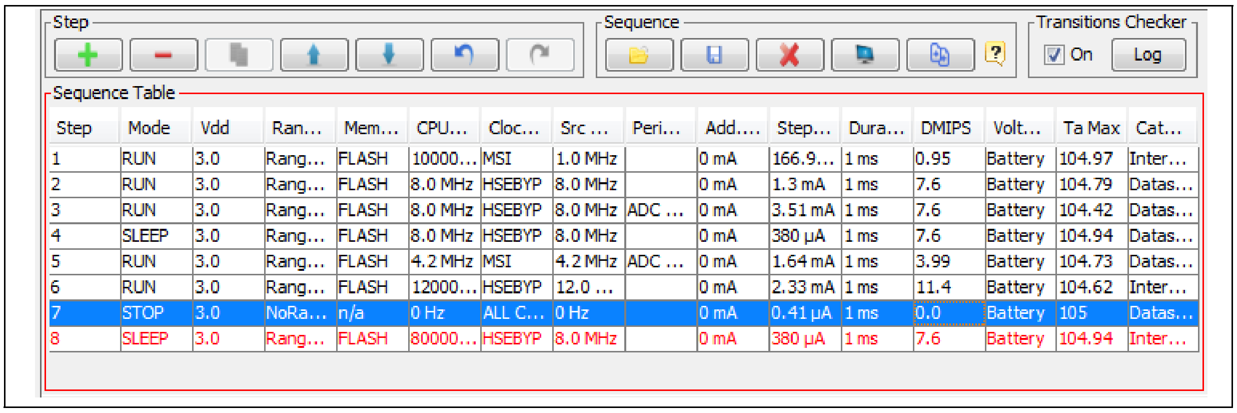
In this case, the user can click the **Show log** button to find out how to solve the transition issue (see *Figure 100*).

Figure 98. Enabling the transition checker option on an already configured sequence - all transitions valid



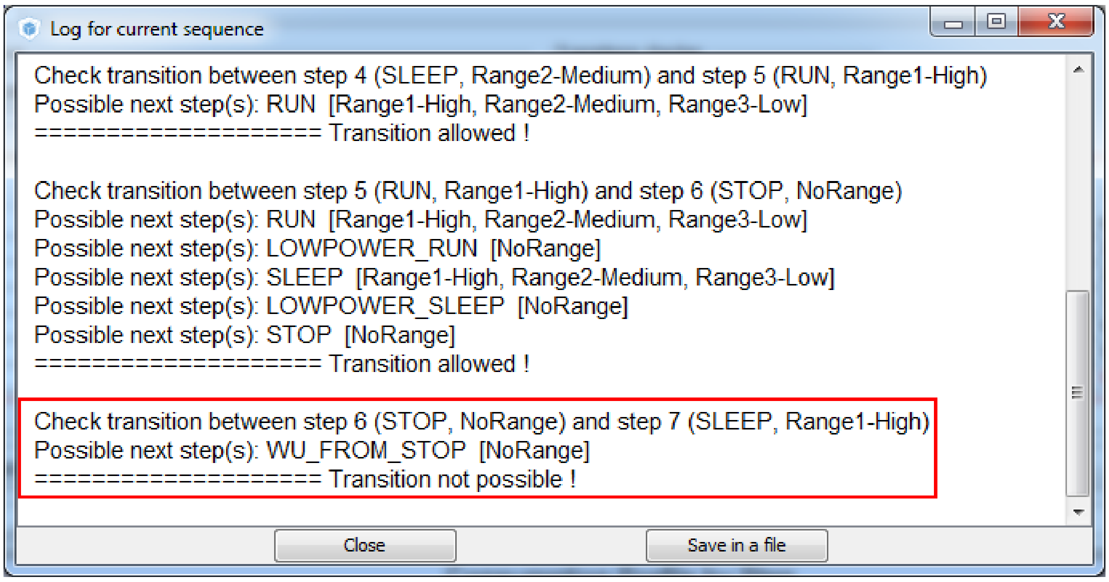
Step	Mode	Vdd	Ran...	Mem...	CPU...	Cloc...	Src ...	Peri...	Add...	Step...	Dur...	DMIPS	Volt...	Ta ...	Cat...
1	RUN	3.0	Rang...	FLASH	1000...	MSI	1.0 MHz		0 mA	166.9...	1 ms	0.95	Battery	104.97	Inter...
2	RUN	3.0	Rang...	FLASH	8.0 MHz	HSEBYP	8.0 MHz		0 mA	1.3 mA	1 ms	7.6	Battery	104.79	Datas...
3	RUN	3.0	Rang...	FLASH	8.0 MHz	HSEBYP	8.0 MHz	ADC ...	0 mA	3.51 mA	1 ms	7.6	Battery	104.42	Datas...
4	SLEEP	3.0	Rang...	FLASH	8.0 MHz	HSEBYP	8.0 MHz		0 mA	380 µA	1 ms	7.6	Battery	104.94	Datas...
5	RUN	3.0	Rang...	FLASH	4.2 MHz	MSI	4.2 MHz	ADC ...	0 mA	1.64 mA	1 ms	3.99	Battery	104.73	Datas...
6	RUN	3.0	Rang...	FLASH	1200...	HSEBYP	12.0 ...		0 mA	2.33 mA	1 ms	11.4	Battery	104.62	Inter...
7	STOP	3.0	NoRa...	n/a	0 Hz	ALL C...	0 Hz		0 mA	0.41 µA	1 ms	0.0	Battery	105	Datas...

Figure 99. Enabling the transition checker option on an already configured sequence - at least one transition invalid



Step	Mode	Vdd	Ran...	Mem...	CPU...	Cloc...	Src ...	Peri...	Add....	Step...	Dura...	DMIPS	Volt...	Ta Max	Cat...
1	RUN	3.0	Rang...	FLASH	10000...	MSI	1.0 MHz		0 mA	166.9...	1 ms	0.95	Battery	104.97	Inter...
2	RUN	3.0	Rang...	FLASH	8.0 MHz	HSEBYP	8.0 MHz		0 mA	1.3 mA	1 ms	7.6	Battery	104.79	Datas...
3	RUN	3.0	Rang...	FLASH	8.0 MHz	HSEBYP	8.0 MHz	ADC ...	0 mA	3.51 mA	1 ms	7.6	Battery	104.42	Datas...
4	SLEEP	3.0	Rang...	FLASH	8.0 MHz	HSEBYP	8.0 MHz		0 mA	380 µA	1 ms	7.6	Battery	104.94	Datas...
5	RUN	3.0	Rang...	FLASH	4.2 MHz	MSI	4.2 MHz	ADC ...	0 mA	1.64 mA	1 ms	3.99	Battery	104.73	Datas...
6	RUN	3.0	Rang...	FLASH	12000...	HSEBYP	12.0 ...		0 mA	2.33 mA	1 ms	11.4	Battery	104.62	Inter...
7	STOP	3.0	NoRa...	n/a	0 Hz	ALL C...	0 Hz		0 mA	0.41 µA	1 ms	0.0	Battery	105	Datas...
8	SLEEP	3.0	Rang...	FLASH	80000...	HSEBYP	8.0 MHz		0 mA	380 µA	1 ms	7.6	Battery	104.94	Inter...

Figure 100. Transition checker option -show log



```
Check transition between step 4 (SLEEP, Range2-Medium) and step 5 (RUN, Range1-High)
Possible next step(s): RUN  [Range1-High, Range2-Medium, Range3-Low]
==================== Transition allowed !

Check transition between step 5 (RUN, Range1-High) and step 6 (STOP, NoRange)
Possible next step(s): RUN  [Range1-High, Range2-Medium, Range3-Low]
Possible next step(s): LOWPOWER_RUN  [NoRange]
Possible next step(s): SLEEP  [Range1-High, Range2-Medium, Range3-Low]
Possible next step(s): LOWPOWER_SLEEP  [NoRange]
Possible next step(s): STOP  [NoRange]
==================== Transition allowed !

Check transition between step 6 (STOP, NoRange) and step 7 (SLEEP, Range1-High)
Possible next step(s): WU_FROM_STOP  [NoRange]
==================== Transition not possible !
```

4.14.2 Configuring a step in the power sequence

The step configuration is performed from the **Edit Step** and **New Step** windows. The graphical interface guides the user by forcing a predefined order for setting parameters.

Their naming may differ according to the selected MCU series. For details on each parameter, refer to *Section 4.14.4: Power sequence step parameters glossary* glossary and to *Appendix D: STM32 microcontrollers power consumption parameters* or to the electrical characteristics section of the MCU datasheet.

The parameters are set automatically by the tool when there is only one possible value (in this case, the parameter cannot be modified and is grayed out). The tool proposes only the configuration choices relevant to the selected MCU.

Proceed as follow to configure a new step:

1. Click **Add** or **Duplicate** to open the **New step** window or double-click a step from the sequence table to open the **Edit step** window.
2. Within the open step window, select in the following order:
   – The **Power Mode**

     Changing the Power Mode resets the whole step configuration.

   – The **Peripherals**

     Peripherals can be selected/unselected at any time after the Power Mode is configured.

   – The **Power scale**

     The power scale corresponds to the power consumption range (STM32L1) or the power scale (STM32F4).

     Changing the Power Mode or the Power Consumption Range discards all subsequent configurations.

   – The **Memory Fetch Type**
   – The V_{DD} value if multiple choices available
   – The voltage source (battery or VBUS)
   – A **Clock Configuration**

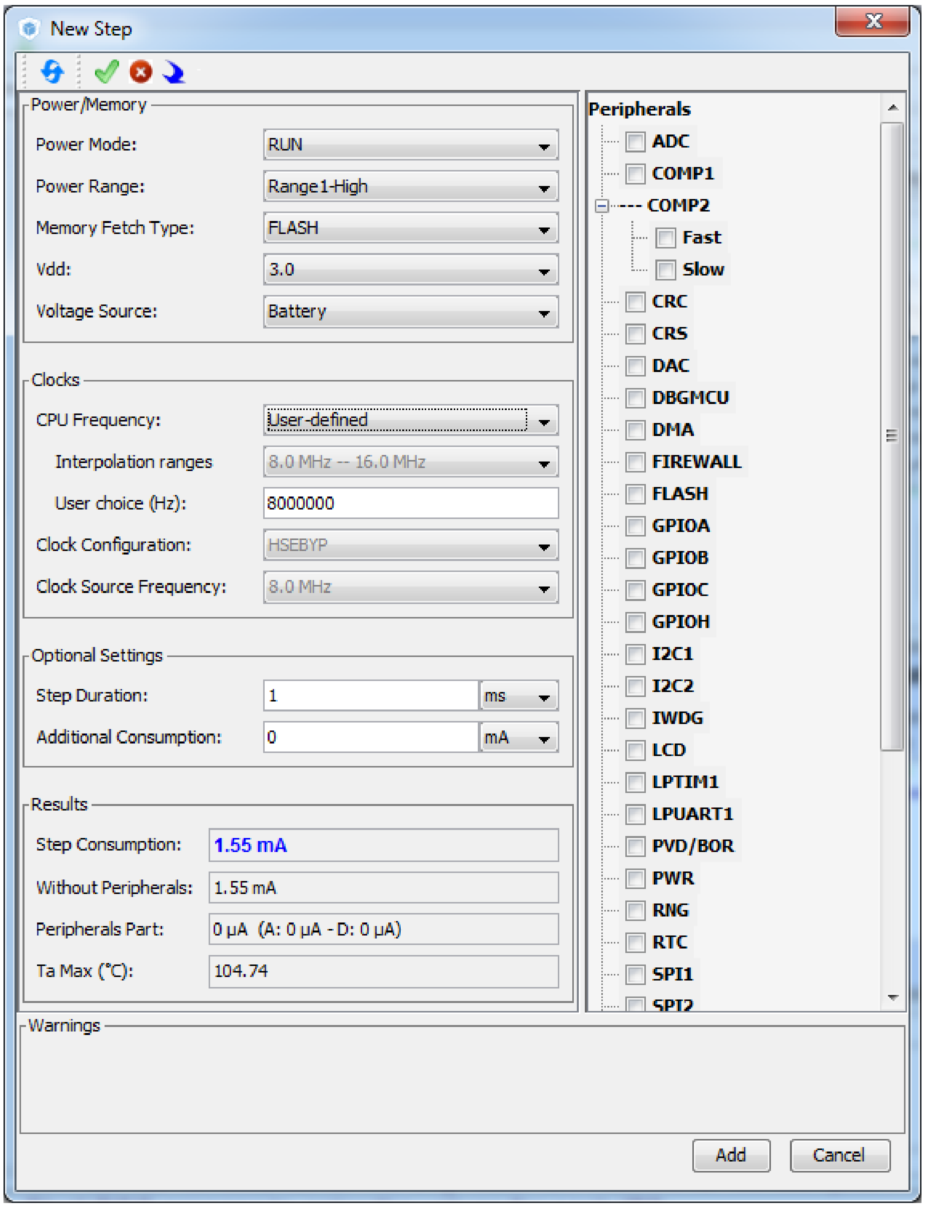
     Changing the Clock Configuration resets the frequency choices further down.

   – When multiple choices are available, the **CPU Frequency** (STM32F4) and the **AHB Bus Frequency/CPU Frequency**(STM32L1) or, for active modes, a user specified frequency. In this case, the consumption value will be interpolated (see *Section : Using interpolation*).
3. Optionally set
   – A **step duration** (1 ms is the default value)
   – An **additional consumption** value (expressed in mA) to reflect, for example, external components used by the application (external regulator, external pull-up, LEDs or other displays). This value added to the microcontroller power consumption will impact the step overall power consumption.
4. Once the configuration is complete, the **Add** button becomes active. Click it to create the step and add it to the sequence table.

Using interpolation

For steps configured for active modes (Run, Sleep), frequency interpolation is supported by selecting CPU frequency as User Defined and entering a frequency in Hz (see *Figure 101*).

Figure 101. Interpolated Power Consumption



Power/Memory

Power Mode:	RUN
Power Range:	Range1-High
Memory Fetch Type:	FLASH
Vdd:	3.0
Voltage Source:	Battery

Clocks

CPU Frequency:	User-defined
Interpolation ranges	8.0 MHz -- 16.0 MHz
User choice (Hz):	8000000
Clock Configuration:	HSEBYP
Clock Source Frequency:	8.0 MHz

Optional Settings

Step Duration:	1	ms
Additional Consumption:	0	mA

Results

Step Consumption:	1.55 mA
Without Peripherals:	1.55 mA
Peripherals Part:	0 µA (A: 0 µA - D: 0 µA)
Ta Max (°C):	104.74

Peripherals

- ☐ ADC
- ☐ COMP1
- COMP2
  - ☐ Fast
  - ☐ Slow
- ☐ CRC
- ☐ CRS
- ☐ DAC
- ☐ DBGMCU
- ☐ DMA
- ☐ FIREWALL
- ☐ FLASH
- ☐ GPIOA
- ☐ GPIOB
- ☐ GPIOC
- ☐ GPIOH
- ☐ I2C1
- ☐ I2C2
- ☐ IWDG
- ☐ LCD
- ☐ LPTIM1
- ☐ LPUART1
- ☐ PVD/BOR
- ☐ PWR
- ☐ RNG
- ☐ RTC
- ☐ SPI1
- ☐ SPI2

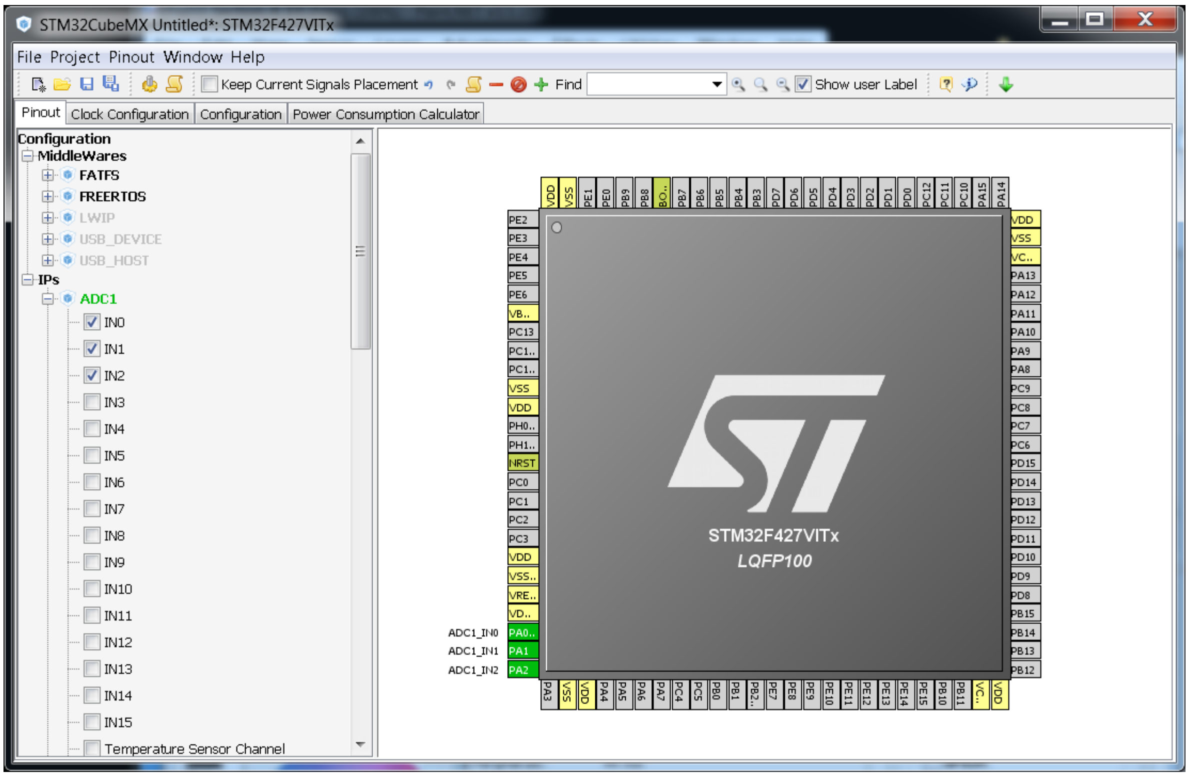
Warnings

Importing pinout

Figure 102 illustrates the example of the ADC configuration in the **Pinout** view: clicking **Import Pinout** in the PCC view selects the ADC IP and GPIO A (*Figure 103*).

The **Import pinout** button allows to automatically select the IPs that have been configured in the **Pinout** view.

Figure 102. ADC selected in Pinout view


STM32CubeMX Untitled*: STM32F427VITx
File Project Pinout Window Help
Keep Current Signals Placement
Find
Show user Label
Pinout
Clock Configuration
Configuration
Power Consumption Calculator
ADC1_IN0
ADC1_IN1
ADC1_IN2
STM32F427VITx
LQFP100

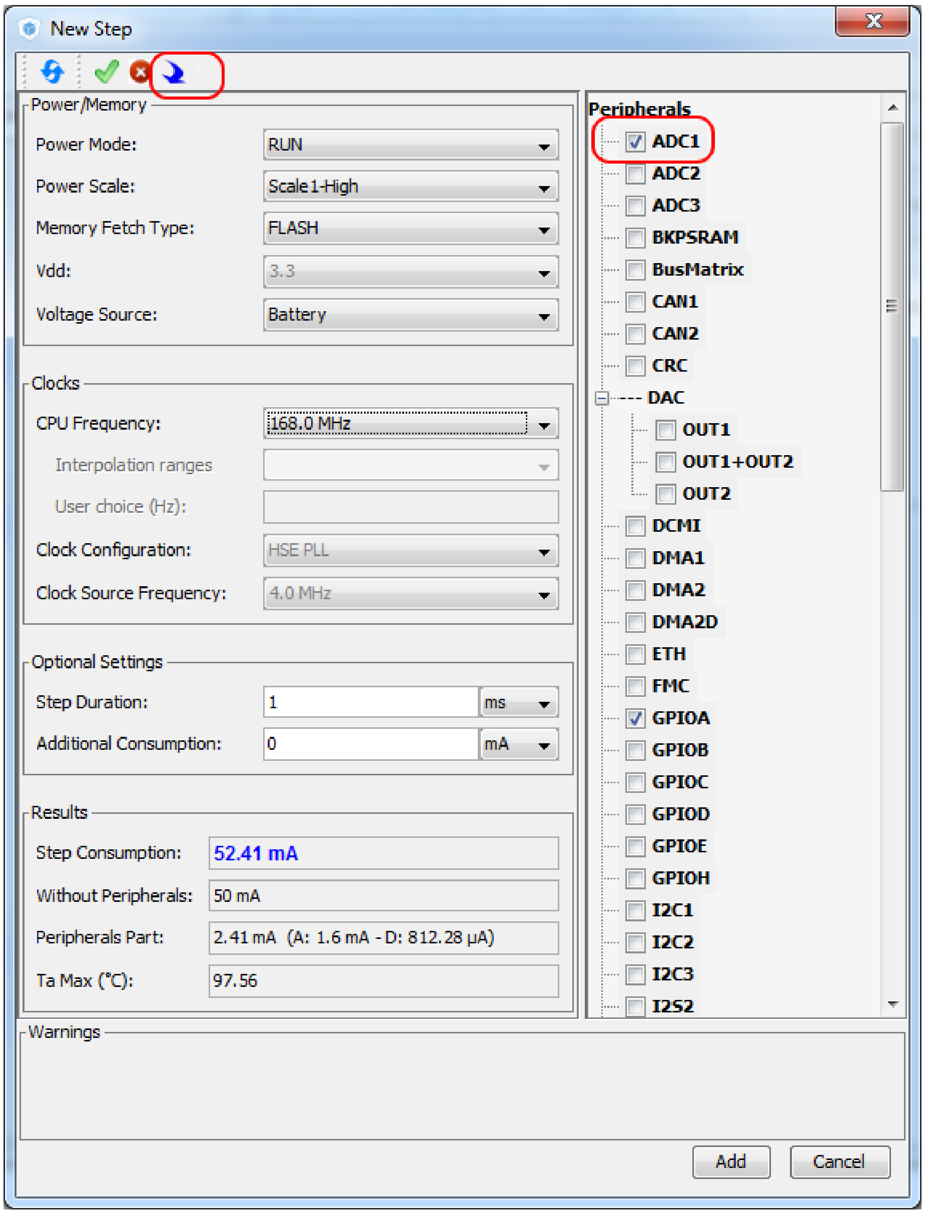
- Configuration
  - MiddleWares
    - FATFS
    - FREERTOS
    - LWIP
    - USB_DEVICE
    - USB_HOST
  - IPs
    - ADC1
      - [✓] IN0
      - [✓] IN1
      - [✓] IN2
      - [ ] IN3
      - [ ] IN4
      - [ ] IN5
      - [ ] IN6
      - [ ] IN7
      - [ ] IN8
      - [ ] IN9
      - [ ] IN10
      - [ ] IN11
      - [ ] IN12
      - [ ] IN13
      - [ ] IN14
      - [ ] IN15
      - [ ] Temperature Sensor Channel

Selecting/deselecting all peripherals

Clicking the **Select All** button ✓ allows selecting all peripherals at once.

Clicking **Deselect All** ✗ removes them as contributors to the step consumption.

Figure 103. PCC Step configuration window: ADC enabled using import pinout



New Step

Power/Memory

Power Mode:	RUN
Power Scale:	Scale1-High
Memory Fetch Type:	FLASH
Vdd:	3.3
Voltage Source:	Battery

Clocks

CPU Frequency:	168.0 MHz
Interpolation ranges	
User choice (Hz):	
Clock Configuration:	HSE PLL
Clock Source Frequency:	4.0 MHz

Optional Settings

Step Duration:	1	ms
Additional Consumption:	0	mA

Results

Step Consumption:	52.41 mA
Without Peripherals:	50 mA
Peripherals Part:	2.41 mA (A: 1.6 mA - D: 812.28 µA)
Ta Max (°C):	97.56

Peripherals

- [✓] ADC1
- [ ] ADC2
- [ ] ADC3
- [ ] BKPSRAM
- [ ] BusMatrix
- [ ] CAN1
- [ ] CAN2
- [ ] CRC
- DAC
  - [ ] OUT1
  - [ ] OUT1+OUT2
  - [ ] OUT2
- [ ] DCMI
- [ ] DMA1
- [ ] DMA2
- [ ] DMA2D
- [ ] ETH
- [ ] FMC
- [✓] GPIOA
- [ ] GPIOB
- [ ] GPIOC
- [ ] GPIOD
- [ ] GPIOE
- [ ] GPIOH
- [ ] I2C1
- [ ] I2C2
- [ ] I2C3
- [ ] I2S2

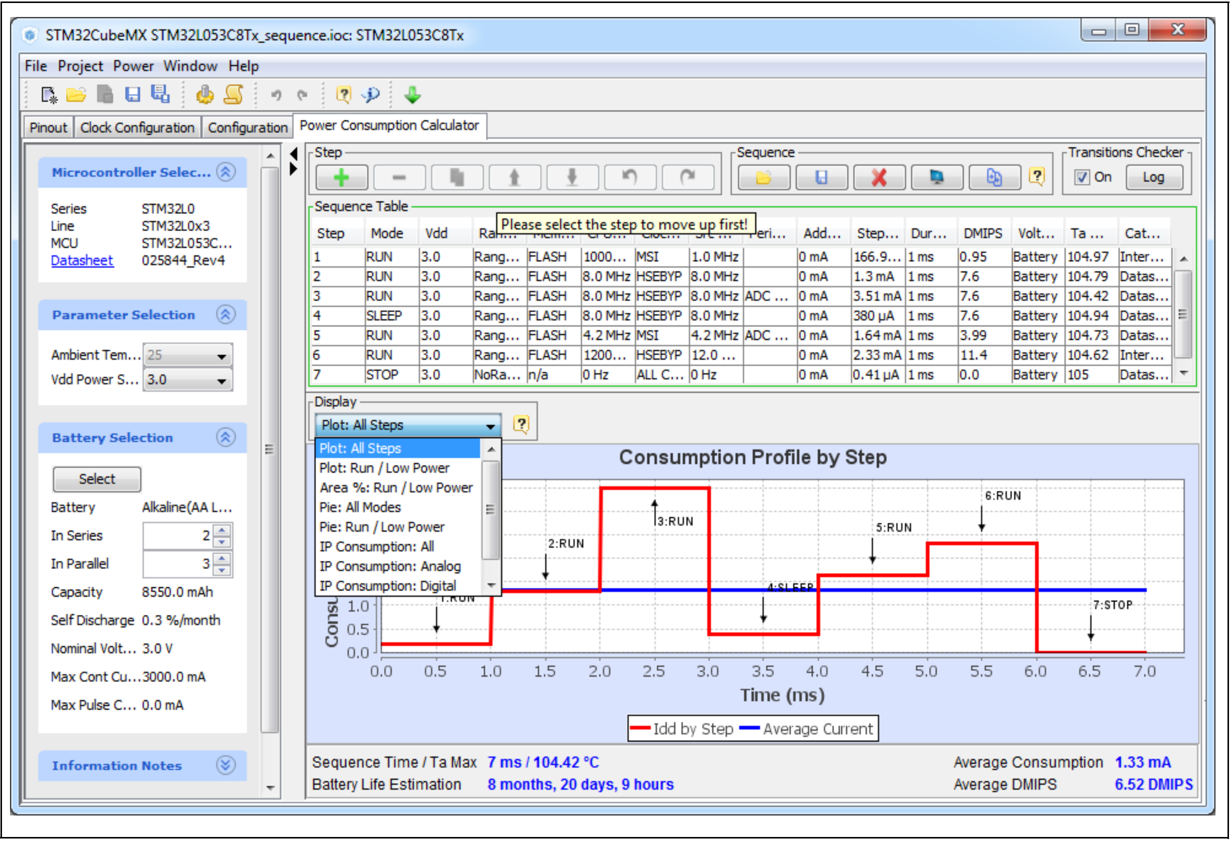
Warnings

4.14.3 Managing user-defined power sequence and reviewing results

The configuration of a power sequence leads to an update of the PCC view (see *Figure 104*):

- The sequence table shows all steps and step parameters values. A category column indicates whether the consumption values are taken from the datasheet or are interpolated.
- The sequence chart area shows different views of the power sequence according to a display type (e.g. plot all steps, plot low power versus run modes, ..)
- The results summary provides the total sequence time, the maximum ambient temperature (T_{AMAX}), plus an estimate of the average power consumption, DMIPS, and battery lifetime provided a valid battery configuration has been selected.

Figure 104. Power Consumption Calculator view after sequence building



STM32CubeMX STM32L053C8Tx_sequence.ioc: STM32L053C8Tx

Series	STM32L0
Line	STM32L0x3
MCU	STM32L053C...
Datasheet	025844_Rev4

Ambient Tem...	25
Vdd Power S...	3.0

Battery	Alkaline(AA L...
In Series	2
In Parallel	3
Capacity	8550.0 mAh
Self Discharge	0.3 %/month
Nominal Volt...	3.0 V
Max Cont Cu...	3000.0 mA
Max Pulse C...	0.0 mA

Please select the step to move up first!

Step	Mode	Vdd	Ra...	Mem...	CPU...	Cloc...	Src ...	Peri...	Add...	Step...	Dur...	DMIPS	Volt...	Ta ...	Cat...
1	RUN	3.0	Rang...	FLASH	1000...	MSI	1.0 MHz		0 mA	166.9...	1 ms	0.95	Battery	104.97	Inter...
2	RUN	3.0	Rang...	FLASH	8.0 MHz	HSEBYP	8.0 MHz		0 mA	1.3 mA	1 ms	7.6	Battery	104.79	Datas...
3	RUN	3.0	Rang...	FLASH	8.0 MHz	HSEBYP	8.0 MHz	ADC ...	0 mA	3.51 mA	1 ms	7.6	Battery	104.42	Datas...
4	SLEEP	3.0	Rang...	FLASH	8.0 MHz	HSEBYP	8.0 MHz		0 mA	380 µA	1 ms	7.6	Battery	104.94	Datas...
5	RUN	3.0	Rang...	FLASH	4.2 MHz	MSI	4.2 MHz	ADC ...	0 mA	1.64 mA	1 ms	3.99	Battery	104.73	Datas...
6	RUN	3.0	Rang...	FLASH	1200...	HSEBYP	12.0 ...		0 mA	2.33 mA	1 ms	11.4	Battery	104.62	Inter...
7	STOP	3.0	NoRa...	n/a	0 Hz	ALL C...	0 Hz		0 mA	0.41 µA	1 ms	0.0	Battery	105	Datas...

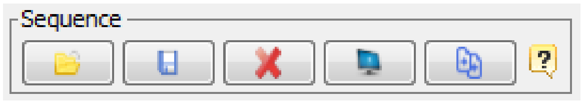
Sequence Time / Ta Max 7 ms / 104.42 °C
Battery Life Estimation 8 months, 20 days, 9 hours
Average Consumption 1.33 mA
Average DMIPS 6.52 DMIPS

Managing the whole sequence (load, save and compare)

The current sequence can be saved or deleted by clicking and , respectively.

In addition, a previously saved sequence can be either loaded in the current view or opened for comparison by clicking (see *Figure 105*).

Figure 105. Sequence table management functions



To load a previously saved sequence:

1. Click the load button .
2. Browse to select the sequence to load.

To open a previously saved sequence for comparison:

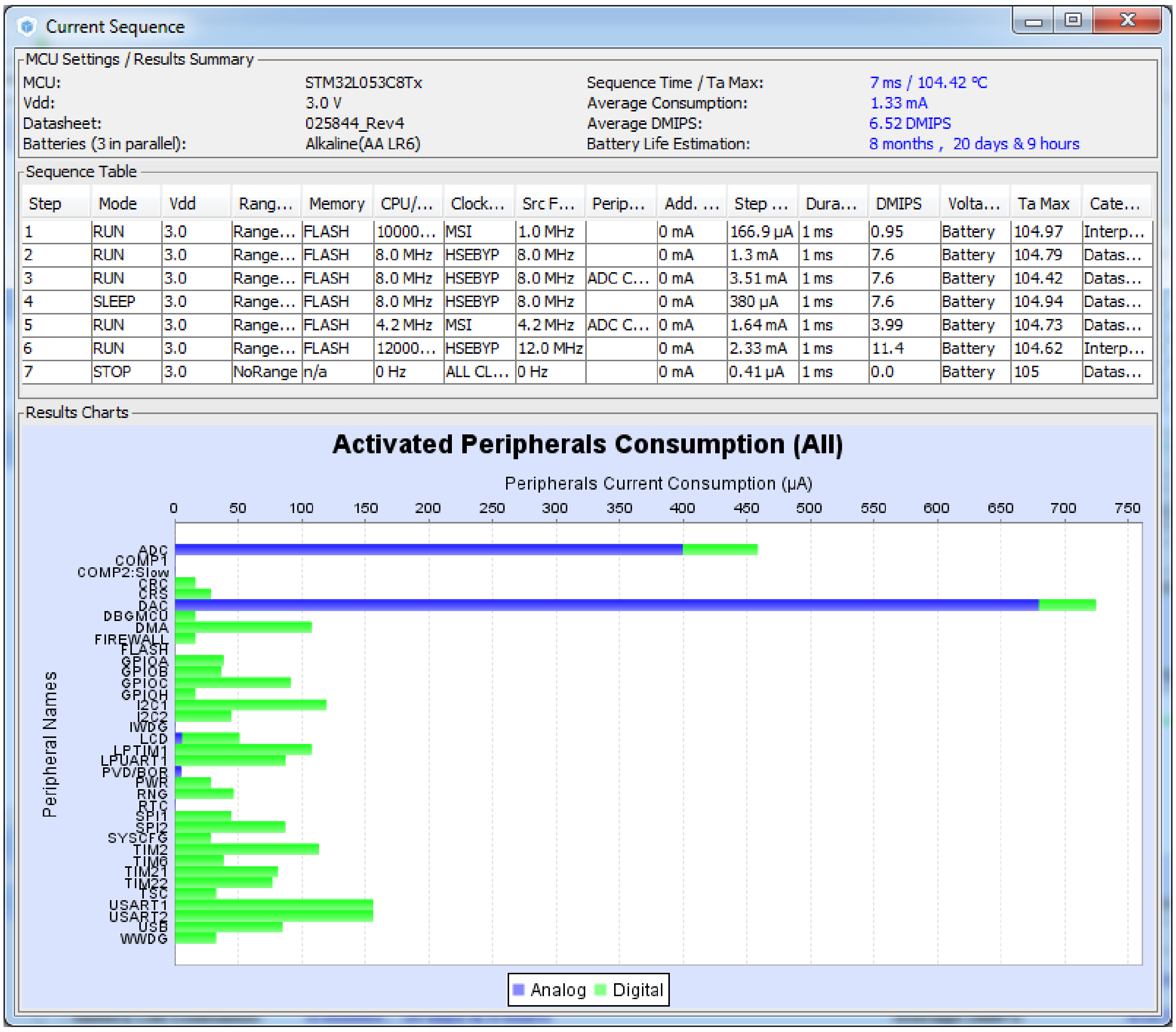
1. Click the **Compare** button .
2. Browse and select the .pcs sequence file to be compared with the current sequence. A new window opens showing the selected sequence details.

Managing the results charts and display options

In the Display area, select the type of chart to display (sequence steps, pie charts, consumption per IPs, ...). You can also click **External Display** to open the charts in dedicated windows (see *Figure 106*).

Right-click on the chart to access the contextual menus: **Properties**, **Copy**, **Save** as png picture file, **Print**, **Zoom** menus, and **Auto Range** to reset to the original view before zoom operations. **Zooming** can also be achieved by mouse selecting from left to right a zone in the chart and **Zoom reset** by clicking the chart and dragging the mouse to the left.

Figure 106. Power Consumption: Peripherals Consumption Chart



MCU Settings / Results Summary

MCU:	STM32L053C8Tx	Sequence Time / Ta Max:	7 ms / 104.42 °C
Vdd:	3.0 V	Average Consumption:	1.33 mA
Datasheet:	025844_Rev4	Average DMIPS:	6.52 DMIPS
Batteries (3 in parallel):	Alkaline(AA LR6)	Battery Life Estimation:	8 months , 20 days & 9 hours

Sequence Table

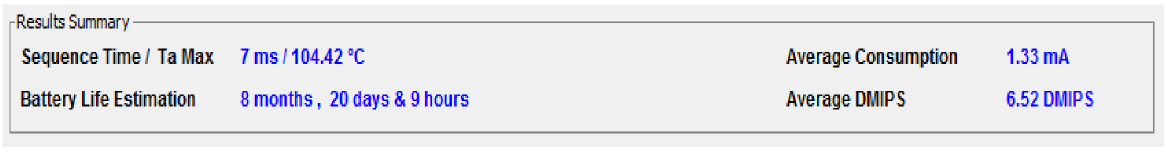
Step	Mode	Vdd	Rang…	Memory	CPU/…	Clock…	Src F…	Perip…	Add. …	Step …	Dura…	DMIPS	Volta…	Ta Max	Cate…
1	RUN	3.0	Range…	FLASH	10000…	MSI	1.0 MHz		0 mA	166.9 µA	1 ms	0.95	Battery	104.97	Interp…
2	RUN	3.0	Range…	FLASH	8.0 MHz	HSEBYP	8.0 MHz		0 mA	1.3 mA	1 ms	7.6	Battery	104.79	Datas…
3	RUN	3.0	Range…	FLASH	8.0 MHz	HSEBYP	8.0 MHz	ADC C…	0 mA	3.51 mA	1 ms	7.6	Battery	104.42	Datas…
4	SLEEP	3.0	Range…	FLASH	8.0 MHz	HSEBYP	8.0 MHz		0 mA	380 µA	1 ms	7.6	Battery	104.94	Datas…
5	RUN	3.0	Range…	FLASH	4.2 MHz	MSI	4.2 MHz	ADC C…	0 mA	1.64 mA	1 ms	3.99	Battery	104.73	Datas…
6	RUN	3.0	Range…	FLASH	12000…	HSEBYP	12.0 MHz		0 mA	2.33 mA	1 ms	11.4	Battery	104.62	Interp…
7	STOP	3.0	NoRange	n/a	0 Hz	ALL CL…	0 Hz		0 mA	0.41 µA	1 ms	0.0	Battery	105	Datas…

Overview of the Results summary area

This area provides the following information (see *Figure 107*):

- Total sequence time as the sum of the sequence steps durations.
- Average consumption as the sum of each step consumption weighed by the step duration.
- The average DMIPS (Dhrystone Million Instructions per Second) based on Dhrystone benchmark, highlighting the CPU performance for the defined sequence.
- Battery life estimation for the selected battery model, based on the average power consumption and the battery self-discharge.
- T_{AMAX}: highest maximum ambient temperature value encountered during the sequence.

Figure 107. Description of the Results area



Results Summary

Sequence Time / Ta Max	7 ms / 104.42 °C	Average Consumption	1.33 mA
Battery Life Estimation	8 months , 20 days & 9 hours	Average DMIPS	6.52 DMIPS

4.14.4 Power sequence step parameters glossary

The parameters that characterize power sequence steps are the following (refer to *Appendix D: STM32 microcontrollers power consumption parameters* for more details):

- Power modes

  To save energy, it is recommended to switch the microcontroller operating mode from running mode, where a maximum power is required, to a low-power mode requiring limited resources.
- V_{CORE} range (STM32L1) or Power scale (STM32F4)

  These parameters are set by software to control the power supply range for digital peripherals.
- Memory Fetch Type

  This field proposes the possible memory locations for application C code execution. It can be either RAM, FLASH or FLASH with ART ON or OFF (only for families that feature a proprietary Adaptive real-time (ART) memory accelerator which increases the program execution speed when executing from Flash memory).

The performance achieved thanks to the ART accelerator is equivalent to 0 wait state program execution from Flash memory. In terms of power consumption, it is equivalent to program execution from RAM. In addition, STM32CubeMX uses the same selection choice to cover both settings, RAM and Flash with ART ON.

- Clock Configuration

  This operation sets the AHB bus frequency or the CPU frequency that will be used for computing the microcontroller power consumption. When there is only one possible choice, the frequencies are automatically configured.

  The clock configuration drop-down list allows to configure the application clocks:

  – The internal or external oscillator sources: MSI, HSI, LSI, HSE or LSE),

  – The oscillator frequency,

  – Other determining parameters: PLL ON, LSE Bypass, AHB prescaler value, LCD with duty...

- Peripherals

  The peripheral list shows the peripherals available for the selected power mode. The power consumption is given assuming that peripherals are only clocked (e.g. not in use by a running program). Each peripheral can be enabled or disabled. Peripherals individual power consumptions are displayed in a tooltip. An overall consumption due to peripheral analog and digital parts is provided in the step Results area (see *Figure 108*).

  The user can select the peripherals relevant for the application:

  – None (**Disable All)**,

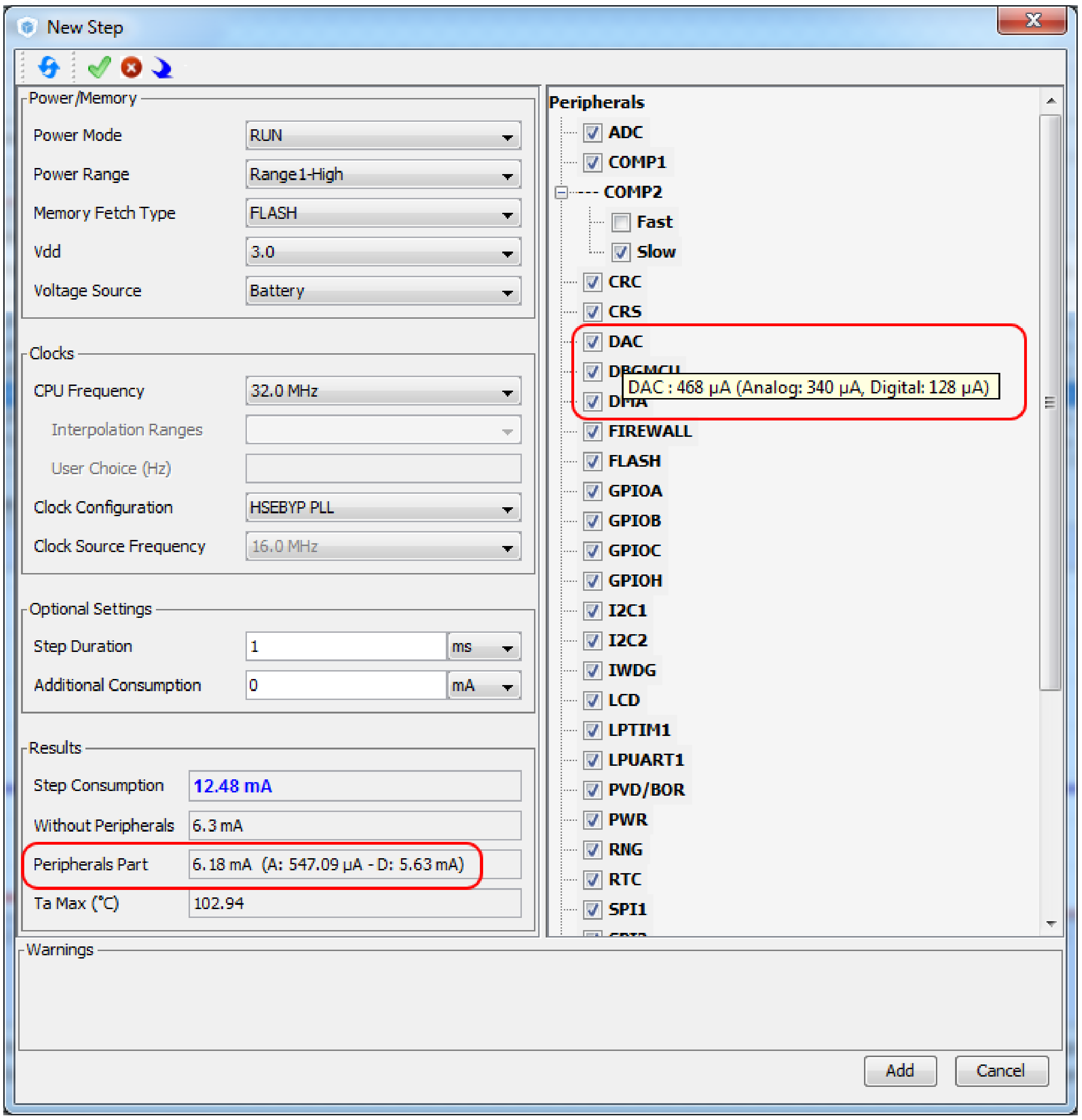
  – Some (using IP individual checkbox),

  – All (**Activate All**),

  – Or all from the previously defined pinout configuration (**Import Pinout**).

  Only the selected and enabled peripherals are taken into account when computing the power consumption.

Figure 108. Peripheral power consumption tooltip


New Step

Power/Memory

Power Mode	RUN
Power Range	Range1-High
Memory Fetch Type	FLASH
Vdd	3.0
Voltage Source	Battery

Clocks

CPU Frequency	32.0 MHz
Interpolation Ranges	
User Choice (Hz)	
Clock Configuration	HSEBYP PLL
Clock Source Frequency	16.0 MHz

Optional Settings

Step Duration	1	ms
Additional Consumption	0	mA

Results

Step Consumption	12.48 mA
Without Peripherals	6.3 mA
Peripherals Part	6.18 mA (A: 547.09 µA - D: 5.63 mA)
Ta Max (°C)	102.94

Warnings

Peripherals

- [✓] ADC
- [✓] COMP1
- COMP2
  - [ ] Fast
  - [✓] Slow
- [✓] CRC
- [✓] CRS
- [✓] DAC
- [✓] DBGMCU
- [✓] DMA
- [✓] FIREWALL
- [✓] FLASH
- [✓] GPIOA
- [✓] GPIOB
- [✓] GPIOC
- [✓] GPIOH
- [✓] I2C1
- [✓] I2C2
- [✓] IWDG
- [✓] LCD
- [✓] LPTIM1
- [✓] LPUART1
- [✓] PVD/BOR
- [✓] PWR
- [✓] RNG
- [✓] RTC
- [✓] SPI1

DAC : 468 µA (Analog: 340 µA, Digital: 128 µA)

- Step duration

  The user can change the default step duration value. When building a sequence, the user can either create steps according to the application actual power sequence or define them as a percentage spent in each mode. For example, if an application spends 30% in Run mode, 20% in Sleep and 50% in Stop, the user must configure a 3-step sequence consisting in 30 ms in Run, 20 ms in Sleep and 50 ms in Stop.
- Additional Consumption

  This field allows entering an additional consumption resulting from specific user configuration (e.g. MCU providing power supply to other connected devices).

4.14.5 Battery glossary

- Capacity (mAh)

  Amount of energy that can be delivered in a single battery discharge.
- Self-discharge (%/month)

  This percentage, over a specified period, represents the loss of battery capacity when the battery is not used (open-circuit conditions), as a result of internal leakage.
- Nominal voltage (V)

  Voltage supplied by a fully charged battery.
- Max. Continuous Current (mA)

  This current corresponds to the maximum current that can be delivered during the battery lifetime period without damaging the battery.
- Max. Pulse Current (mA)

  This is the maximum pulse current that can be delivered exceptionally, for instance when the application is switched on during the starting phase.

5 STM32CubeMX C Code generation overview

Refer to *Section 4.4.2: Project menu* for code generation and C project settings related topics.

5.1 Standard STM32Cube code generation

During the C code generation process, STM32CubeMX performs the following actions:

1. If it is missing, it downloads the relevant STM32Cube firmware package from the user repository. STM32CubeMX repository folder is specified in the **Help > Updater settings** menu.
2. It copies from the firmware package, the relevant files in *Drivers/CMSIS* and *Drivers/STM32F4_HAL_Driver* folders and in the *Middleware* folder if a middleware was selected.
3. It generates the initialization C code (.c/.h files) corresponding to the user MCU configuration and stores it in the *Inc* and *Src* folders. By default, the following files are included:
   - **stm32f4xx_hal_conf.h** file: this file defines the enabled HAL modules and sets some parameters (e.g. External High Speed oscillator frequency) to predefined default values or according to user configuration (clock tree).
   - **stm32f4xx_hal_msp.c** (MSP = MCU Support package): this file defines all initialization functions to configure the IP instances according to the user configuration (pin allocation, enabling of clock, use of DMA and Interrupts).
   - **main.c** is in charge of:

     Resetting the MCU to a known state by calling the *HAL_init()* function that resets all peripherals, initializes the Flash memory interface and the SysTick.

     Configuring and initializing the system clock.

     Configuring and initializing the GPIOs that are not used by IPs.

     Defining and calling, for each configured IP, an IP initialization function that defines a handle structure that will be passed to the corresponding IP *HAL init* function which in turn will call the IP HAL MSP initialization function. Note that when LwIP (respectively USB) middleware is used, the initialization C code for the underlying Ethernet (respectively USB IP) is moved from main.c to LwIP (respectively USB) initialization C code itself.
   - **mxconstants.h** file:

     This file contains the define statements corresponding to the pin labels set from the **Pinout** tab, as well as the user project constants added from the **Configuration** tab (refer to *Figure 109* and *Figure 110* for examples):

```
#define MyTimeOut        10
#define LD4_Pin          GPIO_PIN_12
#define LD4_GPIO_Port    GPIOD
#define LD3_Pin          GPIO_PIN_13
#define LD3_GPIO_Port    GPIOD
#define LD5_Pin          GPIO_PIN_14
#define LD5_GPIO_Port    GPIOD
#define LD6_Pin          GPIO_PIN_15
```

```
#define LD6_GPIO_Port    GPIOD
```

Figure 109. Labels for pins generating define statements

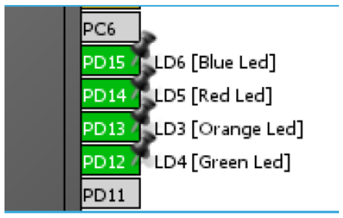
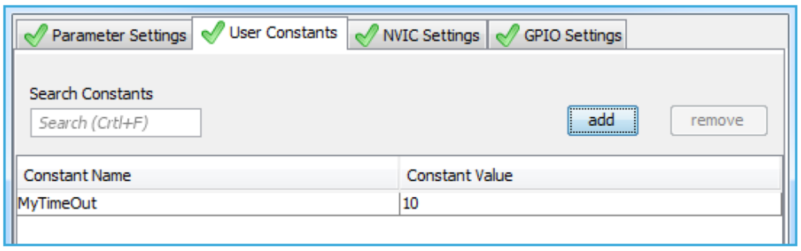


Figure 110. User constant generating define statements



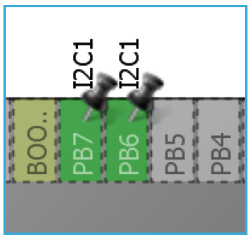
Constant Name	Constant Value
MyTimeOut	10

In case of duplicate labels, a unique suffix, consisting of the pin port letter and the pin index number, is added and used for the generation of the associated define statements.

In the example of a duplicate I2C1 labels shown in *Figure 111*, the code generation produces the following code, keeping the I2C1 label on the original port B pin 6 define statements and adding B7 suffix on pin 7 define statements:

```
#define I2C1_Pin           GPIO_PIN_6
#define I2C1_GPIO_Port     GPIOB
#define I2C1B7_Pin         GPIO_PIN_7
#define I2C1B7_GPIO_Port   GPIOB
```

Figure 111. Duplicate labels



In order for the generated project to compile, define statements shall follow strict naming conventions. They shall start with a letter or an underscore as well as the corresponding label. In addition, they shall not include any special character such as minus sign, parenthesis or brackets. Any special character within the label will be automatically replaced by an underscore in the define name.

If the label contains character strings between "[]" or "()", only the first string listed is used for the define name. As an example, the label "**LD6** [Blue Led]" corresponds the following define statements:

```
#define LD6_Pin   GPIO_PIN_15
#define LD6_GPIO_Port   GPIOD
```

The define statements are used to configure the GPIOs in the generated initialization code. In the following example, the initialization of the pins labeled *Audio_RST_Pin* and *LD4_Pin* is done using the corresponding define statements:

```
/*Configure GPIO pins : LD4_Pin Audio_RST_Pin */
GPIO_InitStruct.Pin = LD4_Pin | Audio_RST_Pin;
GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
GPIO_InitStruct.Pull = GPIO_NOPULL;
GPIO_InitStruct.Speed = GPIO_SPEED_LOW;
HAL_GPIO_Init(GPIOD, &GPIO_InitStruct);
```

4. Finally it generates a *Projects* folder that contains the toolchain specific files that match the user project settings. Double-clicking the IDE specific project file launches the IDE and loads the project ready to be edited, built and debugged.