

RES - Resolver

SD101 | Posted on November 8, 2021 | Updated on May 27, 2025



Simon STROBL

Product Director

imperix • in

Table of Contents

- [Simulink block](#)
 - [Signal specification](#)
 - [Parameters](#)
- [PLECS block](#)
 - [Signal specification](#)
 - [Parameters](#)
- [C++ functions](#)
- [Excitation frequency admissible range](#)

The resolver interface block decodes the feedback from a resolver and provides the position of the rotor for drive applications.

The B-Box RCP supports up to two resolvers through the [Motor Interface for B-Box RCP](#). Usually, a resolver-to-digital converter IC handles this type of sensor by generating the excitation signal (EXC) and extracting the position of the rotor from the quadrature signals (SIN and COS). The resolver module configures the dedicated IC inside the Motor Interface, including the excitation frequency and resolution, and retrieves the decoded position angle. The latter is latched similarly and simultaneously to the sample-&-hold feature of the ADC inputs. Additionally, the Motor Interface also provides the estimated rotation speed.

The RES block is available starting from [version 3.7.1.4](#) of the SDK. The Motor Interface for B-Box RCP is **required** to use this driver.

Simulink block

Signal specification

- The output signal is the mechanical angle in the range $[0; 2\pi]$.
- The `sim` input signal is used in simulation and represents the actual angle value in radian, computed by the simulation plant model.
- The `>` input signal needs to be connected to the sampling clock generated by the [CONFIG block](#) to account for the exact sampling instant in simulation.



Parameters

- Device ID selects which B-Box/B-Board to address when used in a multi-device configuration.
- Motor Interface input selects which connector of the Motor Interface is used as an input.
- Number of pole pairs defines the number of pole pairs of the resolver.
- Direction defines if an increasing angle corresponds to a *clockwise* or *counterclockwise* rotation.
- Excitation frequency defines the excitation frequency, in kHz. The excitation frequency admissible range depends on the resolution (cf. [Excitation frequency admissible range](#) section) and is set by increments of 250 Hz.
- Resolution defines the resolution of the acquired position (10, 12, 14 or 16 bits).

Block Parameters: RES

Resolver
Configures a resolver-to-digital converter of the Motor Interface and retrieves the measured angle.
The output signal is the mechanical angle in the range $[0; 2\pi]$.

Addressing
Device ID (default=0)
Motor Interface input

Resolver configuration
Number of pole pairs
Direction

Resolver-to-digital configuration
Note: Machines A and B share the same resolver-to-digital configuration.
Excitation frequency [kHz]
Resolution

OK Cancel Help Apply

PLECS block

Signal specification

- The output signal is the mechanical angle in the range $[0; 2\pi]$.
- The `sim` input signal is used in simulation and represents the actual angle value in radian, computed by the simulation plant model.
- The `>` input signal needs to be connected to the sampling clock generated by the [CONFIG block](#) to account for the exact sampling instant in simulation.



Parameters

- Device ID selects which B-Box/B-Board to address when used in a multi-device configuration.
- Motor Interface input selects which connector of the Motor Interface is used as an input.
- Number of pole pairs defines the number of pole pairs of the resolver.
- Direction defines if an increasing angle corresponds to a *clockwise* or *counterclockwise* rotation.
- Excitation frequency defines the excitation frequency, in kHz. The excitation frequency admissible range depends on the resolution (cf. [Excitation frequency admissible range](#) section) and is set by increments of 250 Hz.
- Resolution defines the resolution of the acquired position (10, 12, 14 or 16 bits).

Block Parameters: AD2S imperix_template/Imperix controller... X

RES - Resolver (mask) (link)

Configures a resolver-to-digital converter of the Motor Interface and retrieves the measured angle.

The output signal is the mechanical angle in the range $[0; 2\pi]$.

Note: Machines A and B share the same resolver-to-digital configuration. The minimal excitation frequency increment is 250 Hz.

The admissible excitation frequency range depends on the resolution:

- [10; 20] kHz with a 10-bit resolution ;
- [6; 20] kHz with a 12-bit resolution ;
- [3; 12] kHz with a 14-bit resolution ;
- [2; 10] kHz with a 16-bit resolution.

Addressing Resolver configuration

Device ID [default=0]:
0 ☐

Motor interface input:
Machine A (connector X2) ☐

OK Cancel Apply Help

Block Parameters: AD2S imperix_template/Imperix controller... X

RES - Resolver (mask) (link)

Configures a resolver-to-digital converter of the Motor Interface and retrieves the measured angle.

The output signal is the mechanical angle in the range $[0; 2\pi]$.

Note: Machines A and B share the same resolver-to-digital configuration. The minimal excitation frequency increment is 250 Hz.

The admissible excitation frequency range depends on the resolution:

- [10; 20] kHz with a 10-bit resolution ;
- [6; 20] kHz with a 12-bit resolution ;
- [3; 12] kHz with a 14-bit resolution ;
- [2; 10] kHz with a 16-bit resolution.

Addressing Resolver configuration

Number of pole pairs:
1 ☐

Direction:
Clockwise ☐

Excitation frequency [kHz]:
10 ☐

Resolution:
12-bit ☐

OK Cancel Apply Help

C++ functions

MotInt_EnableMotorInterface — Enable the drivers of the Motor Interface

`void MotInt_EnableMotorInterface(unsigned int device=0);` Code language: C++ (cpp)

Enables the drivers of the Motor Interface.

It has to be called in `UserInit()`.

Parameters

- device: the id of the addressed device (optional, used in multi-device configuration only).

Res_SetNumberOfPolePairs — Set the number of pole pairs

`void Res_SetNumberOfPolePairs(tMotIntMachine machine, unsigned int polePairs, unsigned int device=0);` Code language: C++ (cpp)

Defines the number of pole pairs of the resolver.

It has to be called in `UserInit()`.

Parameters

- machine: the machine to configure (*MACHINE_A* or *MACHINE_B*).

- polePairs: the number of pair of poles of the resolver.
- device: the id of the addressed device (optional, used in multi-device configuration only).

Res_ConfigureDirection — Set the rotation direction

```
void Res_ConfigureDirection(tMotIntMachine machine, tMotIntDirection direction, unsigned int device=0);
```

Code language

Configures what rotation direction leads to an increasing angle (positive direction).

It has to be called in UserInit().

Parameters

- machine: the machine to configure (*MACHINE_A* or *MACHINE_B*).
- direction: the positive direction, clockwise (*CW*) or counter-clockwise (*CCW*).
- device: the id of the addressed device (optional, used in multi-device configuration only).

Res_ConfigureExcitationFrequency — Set the excitation frequency

```
void Res_ConfigureExcitationFrequency(float frequency, unsigned int device);
```

Code language: C++ (cpp)

Configures the excitation frequency of the resolvers, in kHz. The excitation frequency is set by increments of 250 Hz. Both resolvers (machine A and B) of a motor interface share the same excitation frequency.

It has to be called in UserInit().

Parameters

- frequency: the excitation frequency to configure, in kHz and by increments of 250 Hz.
- device: the id of the addressed device (optional, used in multi-device configuration only).

Res_ConfigureResolution — Set the resolution

```
void Res_ConfigureResolution(tResResolution resolution, unsigned int device);
```

Code language: C++ (cpp)

Configures the resolution of the acquired position, as 10, 12, 14 or 16 bits. Both resolvers (machine A and B) of a motor interface share the same resolution.

It has to be called in UserInit().

Parameters

- resolution: the resolution to configure (*RES10BIT*, *RES12BIT*, *RES14BIT* or *RES16BIT*).
- device: the id of the addressed device (optional, used in multi-device configuration only).

Res_GetPosition — Get the position

```
float Res_GetPosition(tMotIntMachine machine, unsigned int device=0);
```

Code language: C++ (cpp)

Returns the mechanical angle (position) of the resolver.

It has to be called during the control interrupt.

Parameters

- machine: the machine to configure (*MACHINE_A* or *MACHINE_B*).
- device: the id of the addressed device (optional, used in multi-device configuration only).

Excitation frequency admissible range

The admissible range of the excitation frequency depends on the resolution.

The excitation frequency must be within:

- [10; 20] kHz when the resolution is set to 10 bits;
- [6; 20] kHz when the resolution is set to 12 bits;
- [3; 12] kHz when the resolution is set to 14 bits;
- [2; 10] kHz when the resolution is set to 16 bits.