SoMachine Basic Example Guide

Using Pulse Train Output and Motion Task Tables xSample_PTO_Motion_Table.smbe

12/2016

EIO0000002472.00

www.schneider-electric.com

Schneider Electric

The information provided in this documentation contains general descriptions and/or technical characteristics of the performance of the products contained herein. This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither Schneider Electric nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information contained herein. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

No part of this document may be reproduced in any form or by any means, electronic or mechanical, including photocopying, without express written permission of Schneider Electric.

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

Failure to use Schneider Electric software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this information can result in injury or equipment damage.

Table of Contents

Safety Information 5

About the Book 9

Chapter 1 Example Description 13

Overview 14

Configuring the Pulse Train Output Generator 15

Motion and Administrative Function Blocks 17

Example Description 19

Project Template Description 28

Safety Information

Important Information

NOTICE

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, service, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

⚡ The addition of this symbol to a “Danger” or “Warning” safety label indicates that an electrical hazard exists which will result in personal injury if the instructions are not followed.

⚠ This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates a hazardous situation which, if not avoided, **will result in** death or serious injury.

⚠ WARNING

WARNING indicates a hazardous situation which, if not avoided, **could result in** death or serious injury.

⚠ CAUTION

CAUTION indicates a hazardous situation which, if not avoided, **could result** in minor or moderate injury.

NOTICE

NOTICE is used to address practices not related to physical injury.

PLEASE NOTE

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and its installation, and has received safety training to recognize and avoid the hazards involved.

BEFORE YOU BEGIN

Do not use this product on machinery lacking effective point-of-operation guarding. Lack of effective point-of-operation guarding on a machine can result in serious injury to the operator of that machine.

⚠ WARNING

UNGUARDED EQUIPMENT

- Do not use this software and related automation equipment on equipment which does not have point-of-operation protection.
- Do not reach into machinery during operation.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

This automation equipment and related software is used to control a variety of industrial processes. The type or model of automation equipment suitable for each application will vary depending on factors such as the control function required, degree of protection required, production methods, unusual conditions, government regulations, etc. In some applications, more than one processor may be required, as when backup redundancy is needed.

Only you, the user, machine builder or system integrator can be aware of all the conditions and factors present during setup, operation, and maintenance of the machine and, therefore, can determine the automation equipment and the related safeties and interlocks which can be properly used. When selecting automation and control equipment and related software for a particular application, you should refer to the applicable local and national standards and regulations. The National Safety Council's Accident Prevention Manual (nationally recognized in the United States of America) also provides much useful information.

In some applications, such as packaging machinery, additional operator protection such as point-of-operation guarding must be provided. This is necessary if the operator's hands and other parts of the body are free to enter the pinch points or other hazardous areas and serious injury can occur. Software products alone cannot protect an operator from injury. For this reason the software cannot be substituted for or take the place of point-of-operation protection.

Ensure that appropriate safeties and mechanical/electrical interlocks related to point-of-operation protection have been installed and are operational before placing the equipment into service. All interlocks and safeties related to point-of-operation protection must be coordinated with the related automation equipment and software programming.

NOTE: Coordination of safeties and mechanical/electrical interlocks for point-of-operation protection is outside the scope of the Function Block Library, System User Guide, or other implementation referenced in this documentation.

START-UP AND TEST

Before using electrical control and automation equipment for regular operation after installation, the system should be given a start-up test by qualified personnel to verify correct operation of the equipment. It is important that arrangements for such a check be made and that enough time is allowed to perform complete and satisfactory testing.

⚠ WARNING

EQUIPMENT OPERATION HAZARD

- Verify that all installation and set up procedures have been completed.
- Before operational tests are performed, remove all blocks or other temporary holding means used for shipment from all component devices.
- Remove tools, meters, and debris from equipment.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Follow all start-up tests recommended in the equipment documentation. Store all equipment documentation for future references.

Software testing must be done in both simulated and real environments.

Verify that the completed system is free from all short circuits and temporary grounds that are not installed according to local regulations (according to the National Electrical Code in the U.S.A, for instance). If high-potential voltage testing is necessary, follow recommendations in equipment documentation to prevent accidental equipment damage.

Before energizing equipment:

- Remove tools, meters, and debris from equipment.
- Close the equipment enclosure door.
- Remove all temporary grounds from incoming power lines.
- Perform all start-up tests recommended by the manufacturer.

OPERATION AND ADJUSTMENTS

The following precautions are from the NEMA Standards Publication ICS 7.1-1995 (English version prevails):

- Regardless of the care exercised in the design and manufacture of equipment or in the selection and ratings of components, there are hazards that can be encountered if such equipment is improperly operated.
- It is sometimes possible to misadjust the equipment and thus produce unsatisfactory or unsafe operation. Always use the manufacturer’s instructions as a guide for functional adjustments. Personnel who have access to these adjustments should be familiar with the equipment manufacturer’s instructions and the machinery used with the electrical equipment.
- Only those operational adjustments actually required by the operator should be accessible to the operator. Access to other controls should be restricted to prevent unauthorized changes in operating characteristics.

About the Book

At a Glance

Document Scope

This document describes how to configure the Pulse Train Output of M221 Logic Controller and use the Motion and Administrative function blocks. It also demonstrates the advantages of using Motion tables.

The example described in this document is intended for learning purposes only. It must not be used directly on products that are part of a machine or process.

⚠ **WARNING**

UNINTENDED EQUIPMENT OPERATION

Do not include the code from this example in your machine or process and put into service without thoroughly testing your entire application.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

This document and its related SoMachine Basic project file focus on specific instructions and function blocks provided with SoMachine Basic, and on specific features available in SoMachine Basic. They are intended to help you understand how to develop, test, commission, and integrate applicative software of your own design in your control systems.

The example is intended for new SoMachine Basic users who already have some degree of expertise in the design and programming of control systems.

Validity Note

This document has been updated for the release of SoMachine Basic V1.5.

Related Documents

Title of Documentation	Reference Number
LXM32M AC Servo Drive Product Manual	*0198441113767*
SoMachine Basic - Operating Guide	*EIO0000001354 (ENG)* *EIO0000001355 (FRA)* *EIO0000001356 (GER)* *EIO0000001357 (SPA)* *EIO0000001358 (ITA)* *EIO0000001359 (CHS)* *EIO0000001366 (POR)* *EIO0000001367 (TUR)*
SoMachine Basic Generic Functions - Library Guide	*EIO0000001474 (ENG)* *EIO0000001475 (FRE)* *EIO0000001476 (GER)* *EIO0000001477 (SPA)* *EIO0000001478 (ITA)* *EIO0000001479 (CHS)* *EIO0000001480 (POR)* *EIO0000001481 (TUR)*
Modicon M221 Logic Controller Advanced Functions - Library Guide	*EIO0000002007 (ENG)* *EIO0000002008 (FRE)* *EIO0000002009 (GER)* *EIO0000002010 (SPA)* *EIO0000002011 (ITA)* *EIO0000002012 (CHS)* *EIO0000002013 (POR)* *EIO0000002014 (TUR)*

You can download these technical publications and other technical information from our website at http://www.schneider-electric.com/ww/en/download

Product Related Information

⚠ WARNING

LOSS OF CONTROL

- The designer of any control scheme must consider the potential failure modes of control paths and, for certain critical control functions, provide a means to achieve a safe state during and after a path failure. Examples of critical control functions are emergency stop and overtravel stop, power outage and restart.
- Separate or redundant control paths must be provided for critical control functions.
- System control paths may include communication links. Consideration must be given to the implications of unanticipated transmission delays or failures of the link.
- Observe all accident prevention regulations and local safety guidelines.[1]
- Each implementation of this equipment must be individually and thoroughly tested for proper operation before being placed into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

[1] For additional information, refer to NEMA ICS 1.1 (latest edition), "Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control" and to NEMA ICS 7.1 (latest edition), "Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems" or their equivalent governing your particular location.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Only use software approved by Schneider Electric for use with this equipment.
- Update your application program every time you change the physical hardware configuration.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Do not include the code from this example in your machine or process and put into service without thoroughly testing your entire application.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Chapter 1
Example Description

What Is in This Chapter?

This chapter contains the following topics:

Topic	Page
Overview	14
Configuring the Pulse Train Output Generator	15
Motion and Administrative Function Blocks	17
Example Description	19
Project Template Description	28

Overview

General

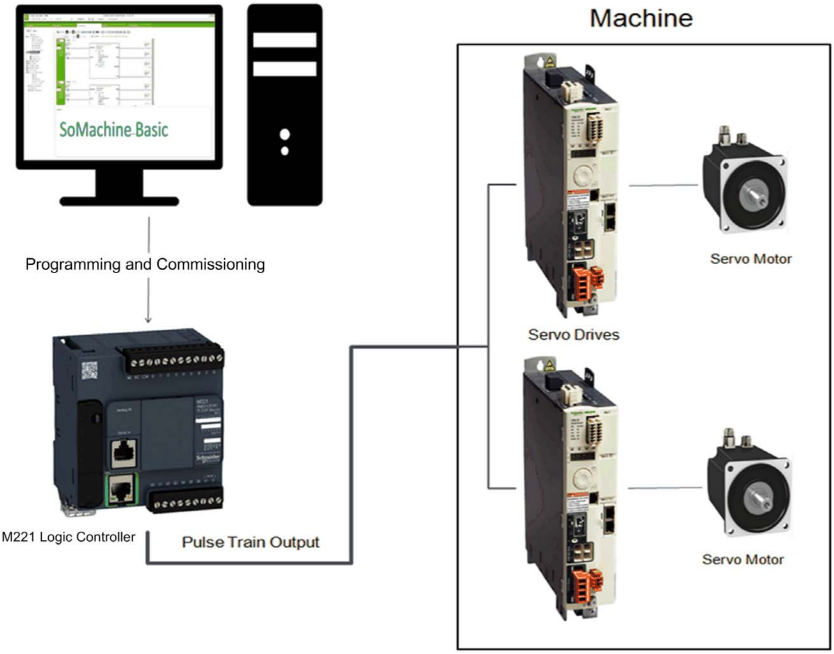
This example guide and its corresponding project template, included with SoMachine Basic, help you to implement the Pulse Train Output (PTO) functionality of the M221 Logic Controller in your application. This guide helps you to:

- Configure the PTO output with SoMachine Basic.
- Implement the PTO function blocks.
- Use Motion tables to simplify the use of multiple Motion function blocks.

This example guide also describes an application that uses the `PTO` function blocks.

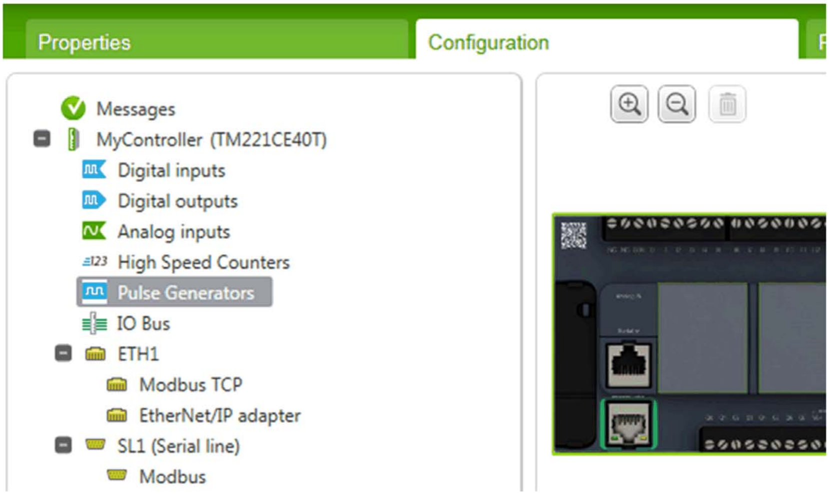
The SoMachine Basic project template is associated with the `xSample_PTO_Motion_Table.smbe` project file. This template is compatible with SoMachine Basic V1.5.

This graphic illustrates a `PTO` setup:

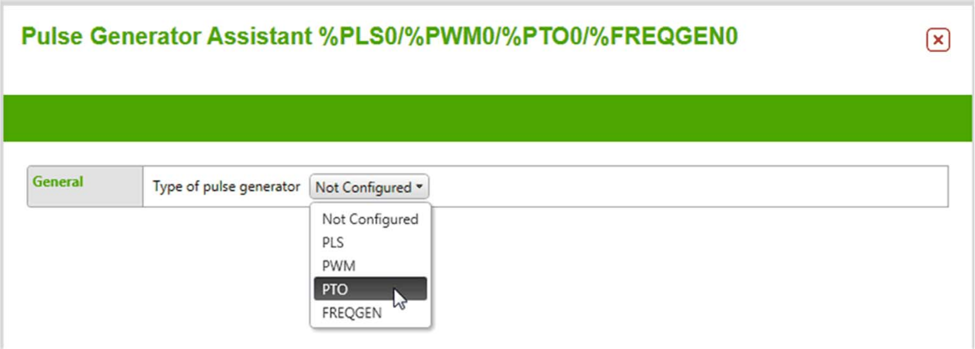


Configuring the Pulse Train Output Generator

To use the Pulse Train Output, first configure the corresponding **Pulse Generators** in SoMachine Basic.

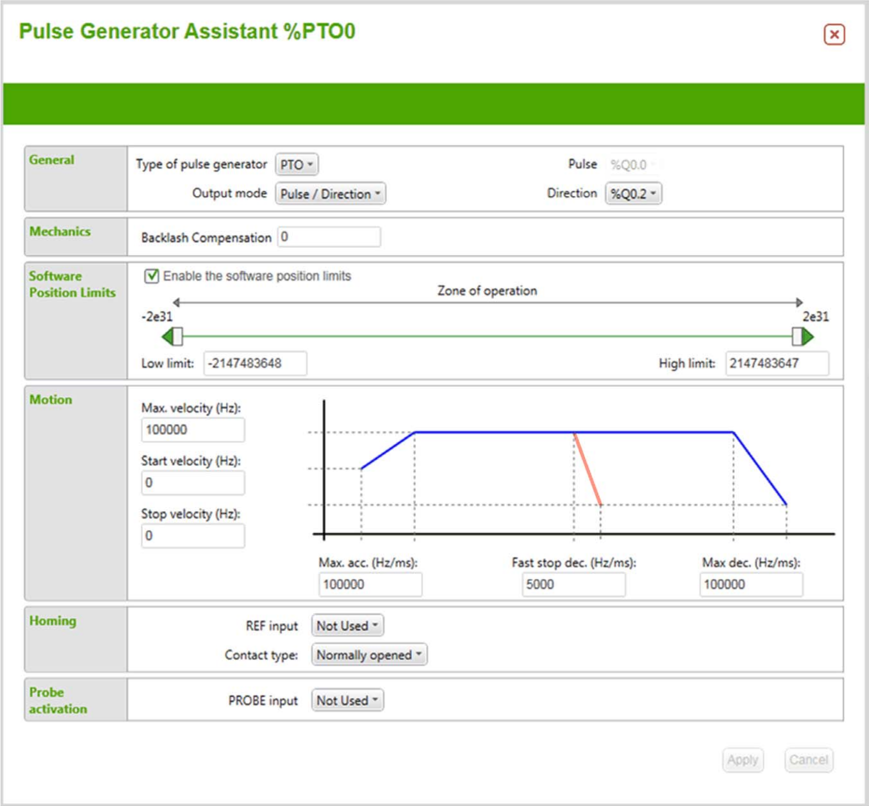


Except for the controller references with relay outputs, there are two **Pulse Generators** in the M221 Logic Controller that you can configure for various applications. You can configure both of them as PTOs and each output is called an axis in SoMachine Basic.


Pulse Generator Assistant %PLS0/%PWM0/%PTO0/%FREQGEN0

General | Type of pulse generator: Not Configured

The **Pulse Generator Assistant %PTOx** enables you to configure the software parameters. A servo drive must be configured separately, refer to Configuring a Servo Drive with SoMove *(see page 21)*. You must adapt the software parameters according to the parameters of your servo drive and motor.



Pulse Generator Assistant %PTO0

General

Type of pulse generator: PTO

Output mode: Pulse / Direction

Pulse: %Q0.0

Direction: %Q0.2

Mechanics

Backlash Compensation: 0

Software Position Limits

☑ Enable the software position limits

Low limit: -2147483648

High limit: 2147483647

Motion

Max. velocity (Hz): 100000

Start velocity (Hz): 0

Stop velocity (Hz): 0

Max. acc. (Hz/ms): 100000

Fast stop dec. (Hz/ms): 5000

Max dec. (Hz/ms): 100000

Homing

REF input: Not Used

Contact type: Normally opened

Probe activation

PROBE input: Not Used

Each axis uses two outputs of the M221 Logic Controller. For the **Pulse/Direction** configuration, `Q0.0` and `Q0.2` are the pulses sent by each axis. `Q0.1` and `Q0.3` are the directional signals sent by each axis. To use the `PTO`, follow the diagram shown in M221 Output Wiring Diagram *(see page 19)*.

Configuration of the **Homing** and **Probe activation** areas of the assistant window is only necessary if you are using `MC_Home_PTO` and `MC_TouchProbe_PTO` function blocks, respectively.

Motion and Administrative Function Blocks

Overview

There are two types of PTO function blocks in SoMachine Basic:

- Administrative
- Motion

PTO is managed in an open loop configuration in the M221 Logic Controller. This means that the M221 Logic Controller does not receive feedback signals directly from the servo drive. Instead, Motion function blocks send the appropriate signals, and the Administrative function blocks provide information on the state of the motor by reading the history of the signals sent by the Motion function blocks. The precision of PTO makes it possible to use an open loop configuration.

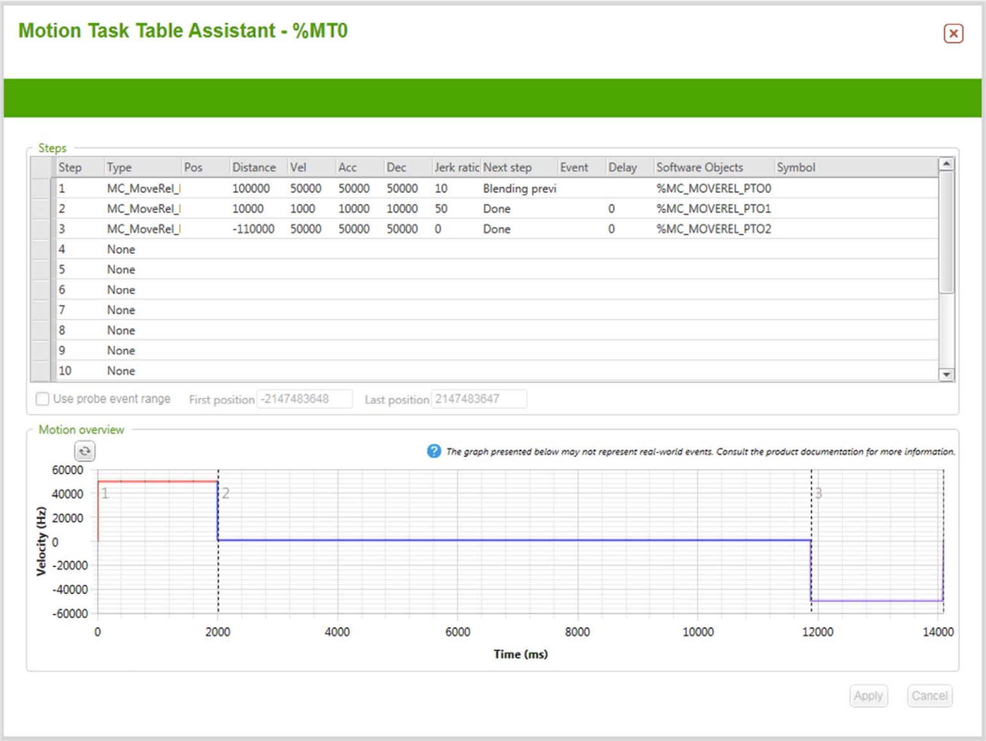
The `MotionTask` function block is a Motion function block that is used to manage the other Motion function blocks. You can find the information on how to use the `MotionTask` function block in Using MotionTask Function Block and Motion Task Tables *(see page 18)*.

Although there are many `PTO` function blocks, it is not necessary to use all of them in your application. It is mandatory to use `MC_Power_PTO` to enable the other function blocks. It is also convenient to use `MC_Reset_PTO` to reset errors detected during execution of the function blocks.

In this template, not all Administrative and Motion function blocks are implemented. For more details on each of these function blocks, refer to the M221 Logic Controller - Advanced Functions Library Guide *(see page 10)*.

Using the `MotionTask` Function Block and Motion Task Tables

The `MotionTask` function block is used in Ladder diagrams. This function block calls Motion Task Table, a table used to regroup Motion function blocks, sequence them, and display the velocity of the motor. For example, in the following example three function blocks are regrouped in a Motion Task Table:



Motion Task Table Assistant - %MT0

Steps

Step	Type	Pos	Distance	Vel	Acc	Dec	Jerk ratic	Next step	Event	Delay	Software Objects	Symbol
1	MC_MoveRel_l		100000	50000	50000	50000	10	Blending previ			%MC_MOVEREL_PTO0	
2	MC_MoveRel_l		10000	1000	10000	10000	50	Done		0	%MC_MOVEREL_PTO1	
3	MC_MoveRel_l		-110000	50000	50000	50000	0	Done		0	%MC_MOVEREL_PTO2	
4	None											
5	None											
6	None											
7	None											
8	None											
9	None											
10	None											

☐ Use probe event range First position -2147483648 Last position 2147483647

Motion overview

The graph presented below may not represent real-world events. Consult the product documentation for more information.

As shown in the graphic above, you can put Motion function blocks (maximum 16) to be executed inside the Motion Task Table. You do not need to program these function blocks as they are managed completely by the Motion Task Table.

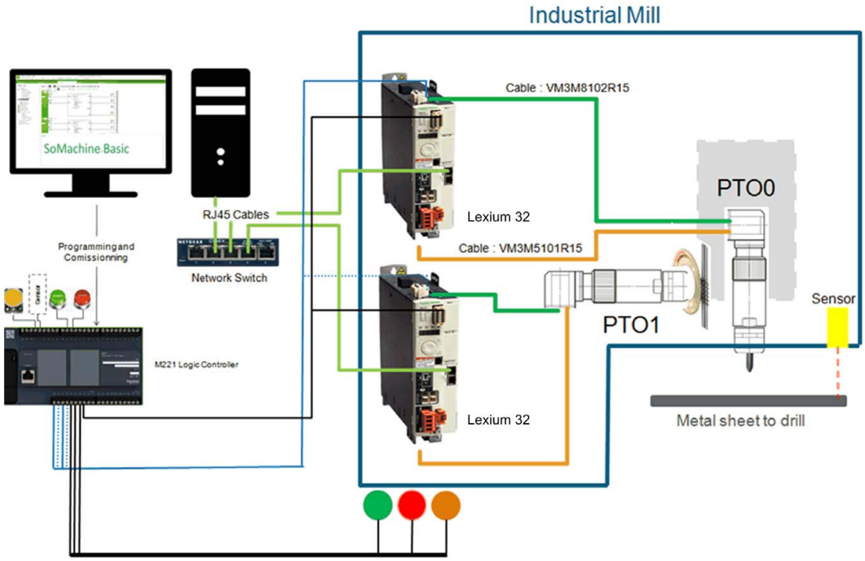
Use a Motion Task Table for a visual representation of the evolution of the velocity of the motor.

You can have up to four Motion Task Tables, each associated with a different axis. Inside a Motion Task Table, you can also assign inputs, called events, which can be used to trigger the next step.

Example Description

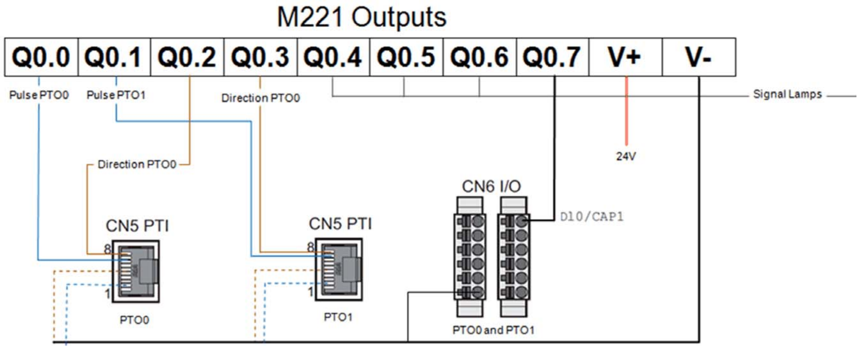
Overview

The example used with the SoMachine Basic project template is illustrated in the following graphic:

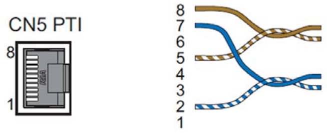


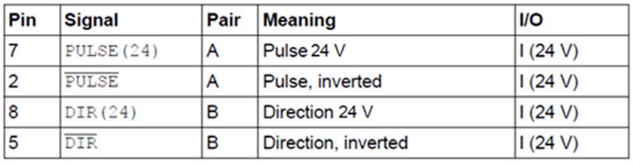
This shows an industrial mill used for drilling thin metal sheets.

This graphic presents the wiring of the outputs of the controller:



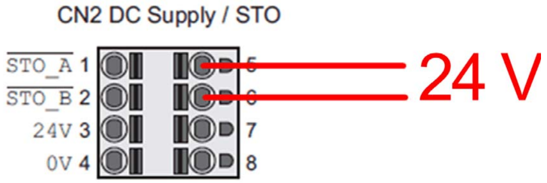
CN5 and **CN6** correspond to the inputs of the Lexium 32 *(see page 10)*. The wiring of the **CN5 Pulse Train Input** of the Lexium needs to be done as explained in the LXM32M AC Servo Drive Product Manual *(see page 10)* and presented in the following graphic:





Pin	Signal	Pair	Meaning	I/O
7	PULSE(24)	A	Pulse 24 V	I (24 V)
2	$\overline{\text{PULSE}}$	A	Pulse, inverted	I (24 V)
8	DIR(24)	B	Direction 24 V	I (24 V)
5	$\overline{\text{DIR}}$	B	Direction, inverted	I (24 V)

You also need to connect the **STO** inputs (**CN2**) to a 24 V power supply:

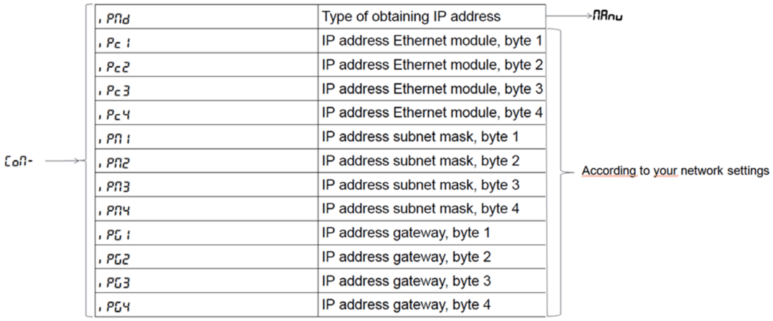


One output of the M221 Logic Controller is connected to one input of the Lexium 32. This input is used for enabling the power stage of the drive and allows the drive to pass from `RDY` state to `RUN` state.

Configuring Servo Drives with SoMove

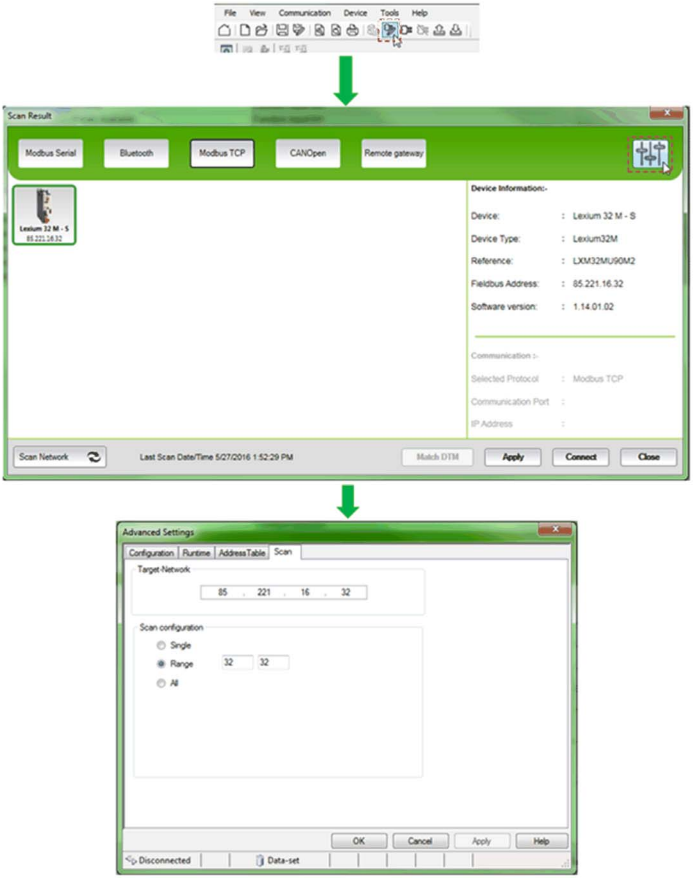
The initial configuration of the Lexium 32 drives is done using a SoMove application. SoMove is a Schneider Electric software that can be used to configure and monitor Schneider Electric variable frequency and servo drives. For Lexium drives, you need to *download the correct library* from the Schneider Electric website and install it.

You can connect your PC to the drives through their respective Ethernet ports. To do so, you need to specify the IP address of the drive in SoMove. The IP address configuration of the Lexium can be found through the human machine interface integrated in the drive:



iPMd	Type of obtaining IP address
iPc1	IP address Ethernet module, byte 1
iPc2	IP address Ethernet module, byte 2
iPc3	IP address Ethernet module, byte 3
iPc4	IP address Ethernet module, byte 4
iPM1	IP address subnet mask, byte 1
iPM2	IP address subnet mask, byte 2
iPM3	IP address subnet mask, byte 3
iPM4	IP address subnet mask, byte 4
iPG1	IP address gateway, byte 1
iPG2	IP address gateway, byte 2
iPG3	IP address gateway, byte 3
iPG4	IP address gateway, byte 4

After configuring the IP address of the drive, the same IP address must be entered in the following fields in SoMove:



Device Information:-

Device: : Lexium 32 M - S

Device Type: : Lexium32M

Reference: : LXM32MU90M2

Fieldbus Address: : 85.221.16.32

Software version: : 1.14.01.02

Communication :-

Selected Protocol : Modbus TCP

Communication Port :

IP Address :

Last Scan Date/Time 5/27/2016 1:52:29 PM

Target-Network

85 . 221 . 16 . 32

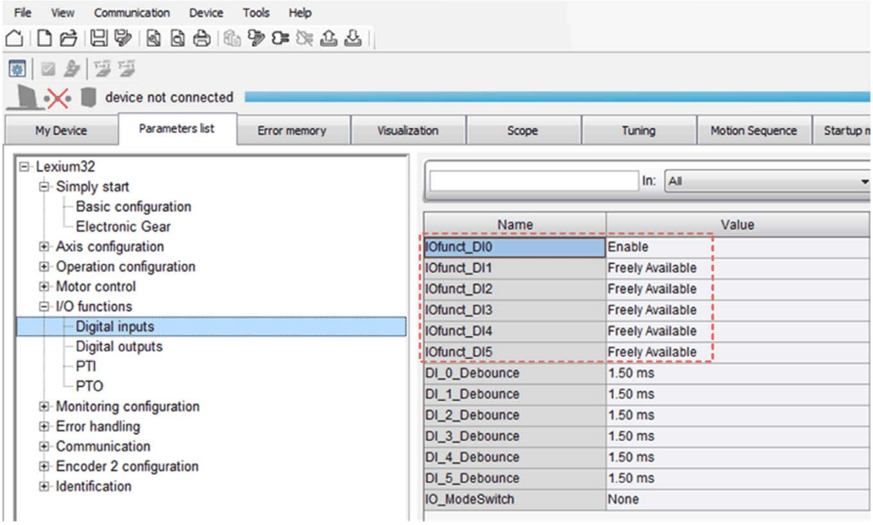
Scan configuration

○ Single

● Range 32 32

○ All

After the connection is established, you need to configure the following parameters (disable inputs DI1, DI2, DI3, DI4, and DI5):

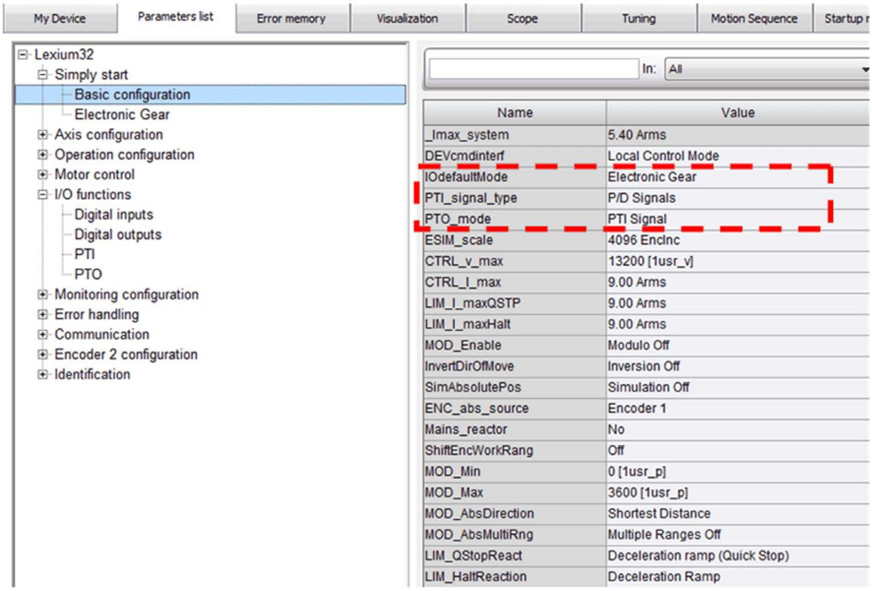


Name	Value
IOfunct_DI0	Enable
IOfunct_DI1	Freely Available
IOfunct_DI2	Freely Available
IOfunct_DI3	Freely Available
IOfunct_DI4	Freely Available
IOfunct_DI5	Freely Available
DI_0_Debounce	1.50 ms
DI_1_Debounce	1.50 ms
DI_2_Debounce	1.50 ms
DI_3_Debounce	1.50 ms
DI_4_Debounce	1.50 ms
DI_5_Debounce	1.50 ms
IO_ModeSwitch	None

To use the PTI input of the drive, choose the electronic gear mode.

The remaining information given hereafter about SoMove is optional. Once the initial configuration is done, you no longer need to be connected to your drive.

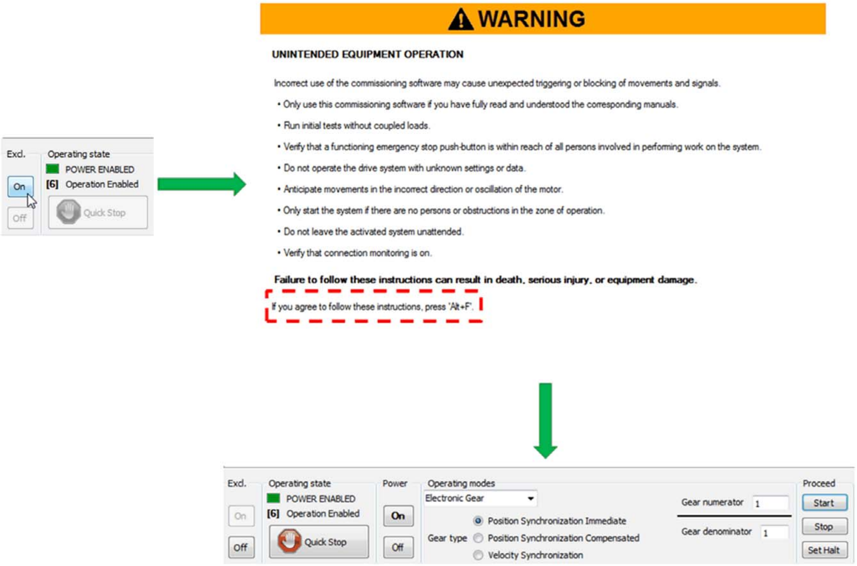
In SoMove, you can find the memory address of the parameters and read them with the M221 Logic Controller using the Modbus protocol. Most of the parameters are under the **Simply start** → **Basic Configuration** menu which is presented in the following graphic:



In: All

Name	Value
_Imax_system	5.40 Arms
DEVcmdinterf	Local Control Mode
IOdefaultMode	Electronic Gear
PTI_signal_type	P/D Signals
PTO_mode	PTI Signal
ESIM_scale	4096 EncInc
CTRL_v_max	13200 [1usr_v]
CTRL_I_max	9.00 Arms
LIM_I_maxQSTP	9.00 Arms
LIM_I_maxHalt	9.00 Arms
MOD_Enable	Modulo Off
InvertDirOfMove	Inversion Off
SimAbsolutePos	Simulation Off
ENC_abs_source	Encoder 1
Mains_reactor	No
ShiftEncWorkRang	Off
MOD_Min	0 [1usr_p]
MOD_Max	3600 [1usr_p]
MOD_AbsDirection	Shortest Distance
MOD_AbsMultiRng	Multiple Ranges Off
LIM_QStopReact	Deceleration ramp (Quick Stop)
LIM_HaltReaction	Deceleration Ramp

You can take the control of the drive and monitor its states by using the executive control option:



⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Incorrect use of the commissioning software may cause unexpected triggering or blocking of movements and signals.

- Only use this commissioning software if you have fully read and understood the corresponding manuals.
- Run initial tests without coupled loads.
- Verify that a functioning emergency stop push-button is within reach of all persons involved in performing work on the system.
- Do not operate the drive system with unknown settings or data.
- Anticipate movements in the incorrect direction or oscillation of the motor.
- Only start the system if there are no persons or obstructions in the zone of operation.
- Do not leave the activated system unattended.
- Verify that connection monitoring is on.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

If you agree to follow these instructions, press 'Alt+F'.

Operating state: 🟩 POWER ENABLED

[6] Operation Enabled

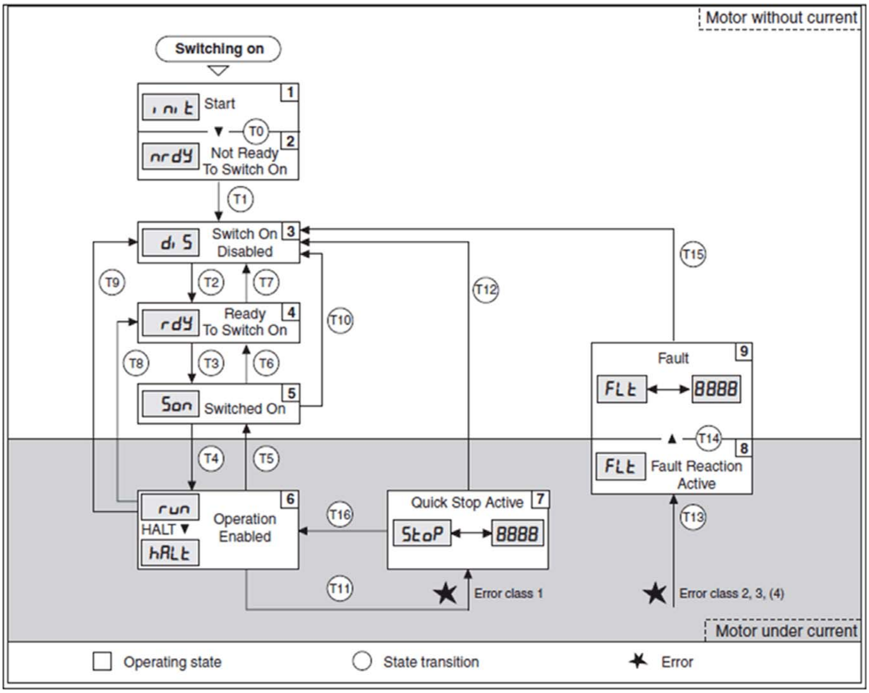
Operating modes: Electronic Gear

Gear type: ● Position Synchronization Immediate ○ Position Synchronization Compensated ○ Velocity Synchronization

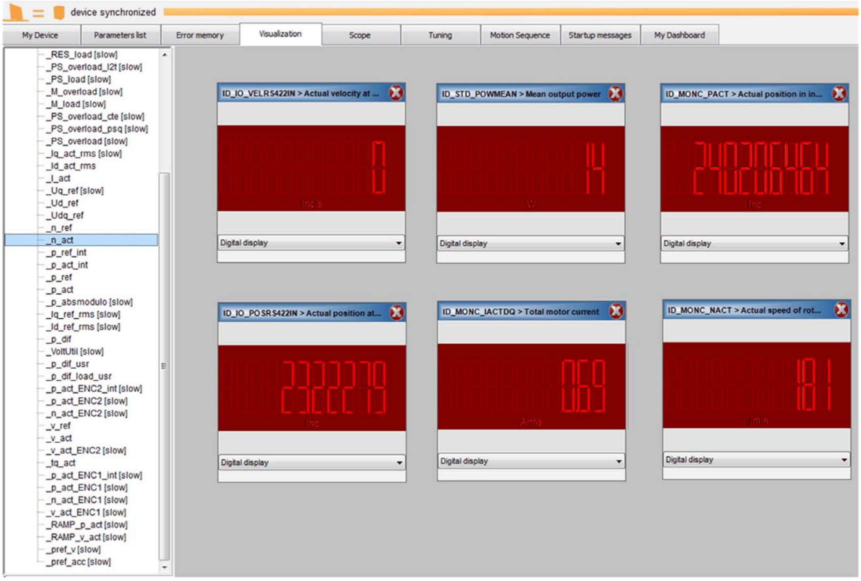
Gear numerator 1

Gear denominator 1

The Lexium 32 implements the following state machine diagram, which is shown under **Operation state** in SoMove:



In SoMove, you can visualize certain parameters (**Velocity**, **Position**, **Errors** and so on) in the **Visualization** tab:



ID_IO_VELRS422IN > Actual velocity at ...
0
Inc/s
Digital display

ID_STD_POWMEAN > Mean output power
14
W
Digital display

ID_MONC_PACT > Actual position in in...
240206464
Inc
Digital display

ID_IO_POSRS422IN > Actual position at...
2322279
Inc
Digital display

ID_MONC_IACTDQ > Total motor current
0.69
Arms
Digital display

ID_MONC_NACT > Actual speed of rot...
181
1/min
Digital display

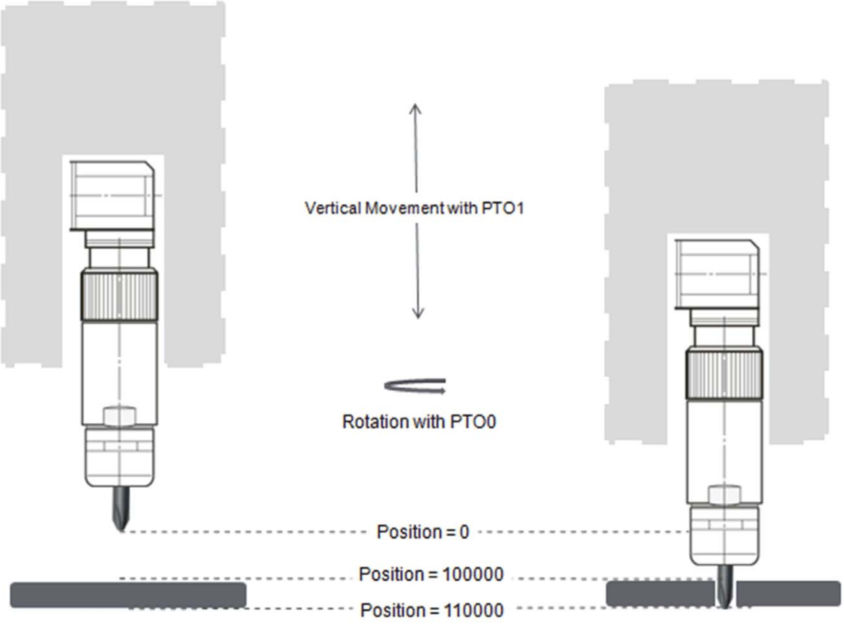
Project Template Description

Overview

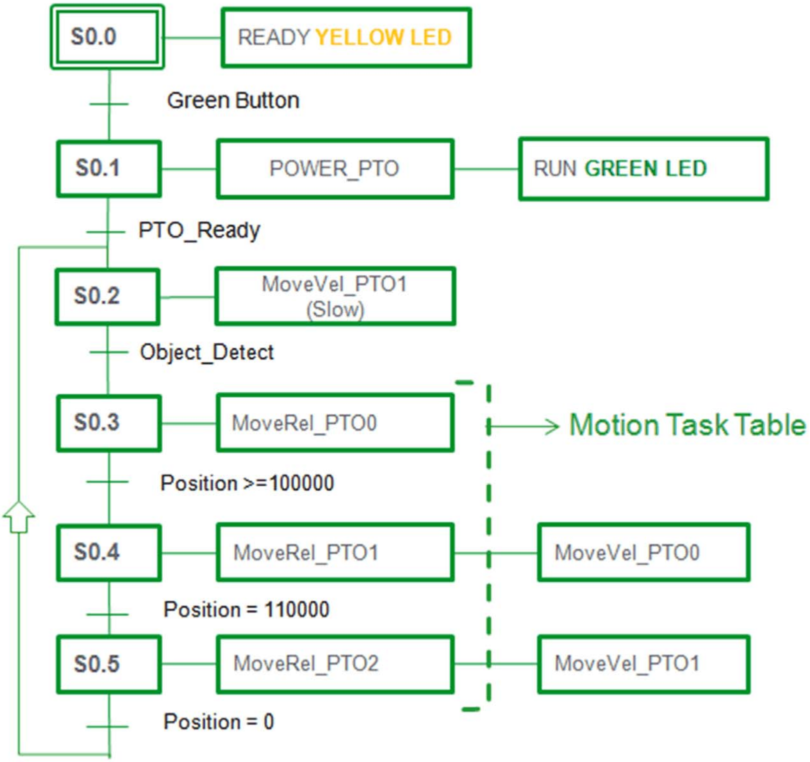
This section describes the operating principles of the project template `xSample_PTO_Motion_Table.smbe`.

This project template is based on the industrial mill described in the Example Description *(see page 19)*.

This graphic illustrates the movement of the drill:



The drill waits for a metal sheet at position 0 while rotating at a low speed. Upon the detection of a sheet, it translates towards the sheet very fast and increases its rotational speed before touching the sheet. Just before touching the sheet (position 100000), it decreases its translation speed and starts drilling. Once the drilling is completed (position 110000), it returns to its initial position with a fast translation speed and decreases its rotational speed. This operation is illustrated with the following Grafcet (SFC) state chart.

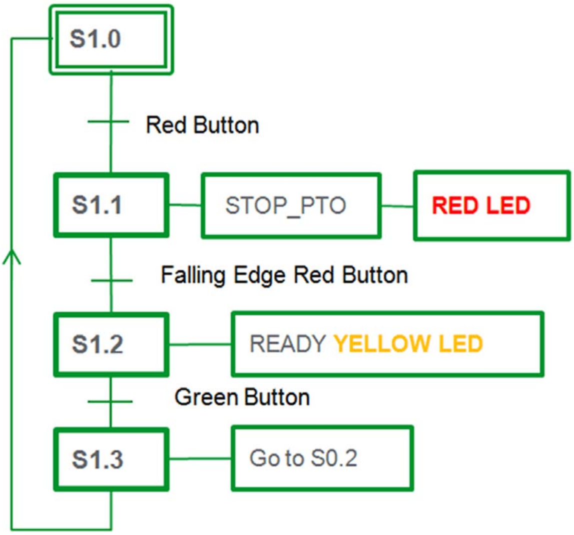


The machine is controlled by an operator. The operator can start the machine with the start button `%I0.7` (green button) or stop it with `%I0.8` (red button). It is possible to monitor the state of the machine with the LEDs associated with the outputs `%Q0.4`, `%Q0.5`, and `%Q0.6`.

A description of the program is:

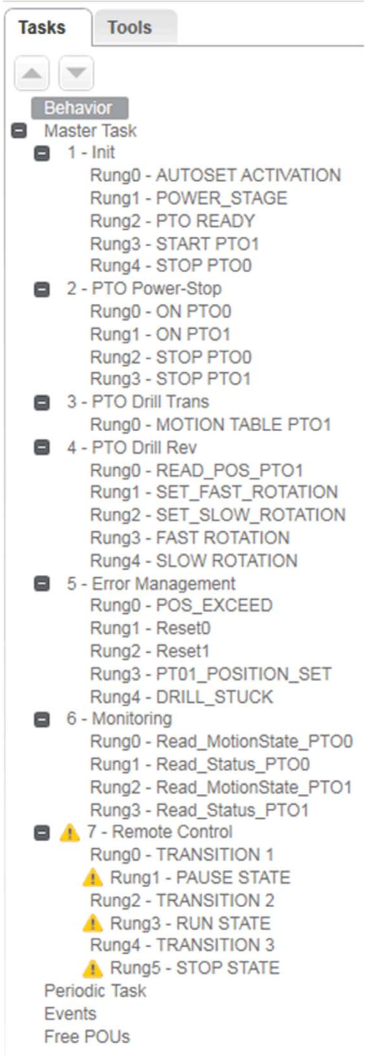
- Pushing the green button twice starts the machine (the second time releases the button). The green LED (**Run**) is illuminated when both `MC_Power_PTO` function blocks are activated. The drill waits to detect an object and then proceeds with the steps given in the Motion Task Table (**S0.3** in the Grafcet (SFC) state chart above).
- Pushing the red button stops the machine. When the button is pushed (first push), the red LED (**Stop**) is illuminated. If the green button is pressed at the same time, the machine will not start.

- When the red button is released, the yellow LED (**Ready**) is illuminated. At in this state, the operator can reset the position of the machine by pushing the `Set_Position` button `I0.0`. Pushing the green button again restarts the machine. This is illustrated in the following Grafcet (SFC) state chart:



Structure of the SoMachine Basic Project Template

The SoMachine Basic project template is composed of various rungs and POUs as shown in the following graphic:



- Master Task
  - 1 - Init
    - Rung0 - AUTOSET ACTIVATION
    - Rung1 - POWER_STAGE
    - Rung2 - PTO READY
    - Rung3 - START PTO1
    - Rung4 - STOP PTO0
  - 2 - PTO Power-Stop
    - Rung0 - ON PTO0
    - Rung1 - ON PTO1
    - Rung2 - STOP PTO0
    - Rung3 - STOP PTO1
  - 3 - PTO Drill Trans
    - Rung0 - MOTION TABLE PTO1
  - 4 - PTO Drill Rev
    - Rung0 - READ_POS_PTO1
    - Rung1 - SET_FAST_ROTATION
    - Rung2 - SET_SLOW_ROTATION
    - Rung3 - FAST ROTATION
    - Rung4 - SLOW ROTATION
  - 5 - Error Management
    - Rung0 - POS_EXCEED
    - Rung1 - Reset0
    - Rung2 - Reset1
    - Rung3 - PT01_POSITION_SET
    - Rung4 - DRILL_STUCK
  - 6 - Monitoring
    - Rung0 - Read_MotionState_PTO0
    - Rung1 - Read_Status_PTO0
    - Rung2 - Read_MotionState_PTO1
    - Rung3 - Read_Status_PTO1
  - ⚠ 7 - Remote Control
    - Rung0 - TRANSITION 1
    - ⚠ Rung1 - PAUSE STATE
    - Rung2 - TRANSITION 2
    - ⚠ Rung3 - RUN STATE
    - Rung4 - TRANSITION 3
    - ⚠ Rung5 - STOP STATE
- Periodic Task
- Events
- Free POUs

The first four POUs and `POU7` of this template implement the operating principles of the machine and are identical to the Grafcet (SFC) state charts *(see page 28)*. `POU5` and `POU6` are for error management and monitoring, which are implemented to demonstrate the advantages of using the Administrative function blocks.

1 - Init

This POU initializes the machine when the **Start** button is pressed. MC_Power_PTO function blocks are enabled through the variables (%M1 and %M5) of this POU, and this allows the first servo motor to start turning.

2 - PTO Power-Stop

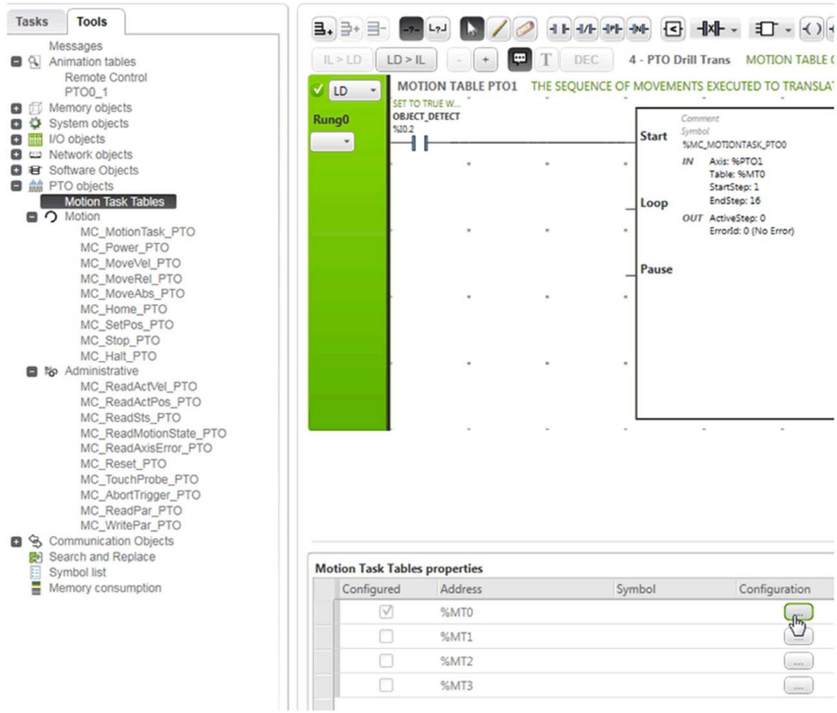
This POU is composed of the MC_Power_PTO and MC_Stop_PTO function blocks of both axes.

When %M5 is set in POU1, MC_Power_PTO0 and MC_Power_PTO1 are enabled, which in turn enables the other Motion function blocks.

If the operator presses the **Stop** button, or if an error is detected in POU5, both axes are disabled.

3 - PTO Drill Trans

This POU is composed of MC_MotionTask_PTO0. This function block needs to be configured in **Tools → PTO objects → Motion Task Tables** before using it.



MOTION TABLE PTO1 THE SEQUENCE OF MOVEMENTS EXECUTED TO TRANSLA

SET TO TRUE W...
OBJECT_DETECT
%I0.2

Start
Loop
Pause

Comment
Symbol
%MC_MOTIONTASK_PTO0

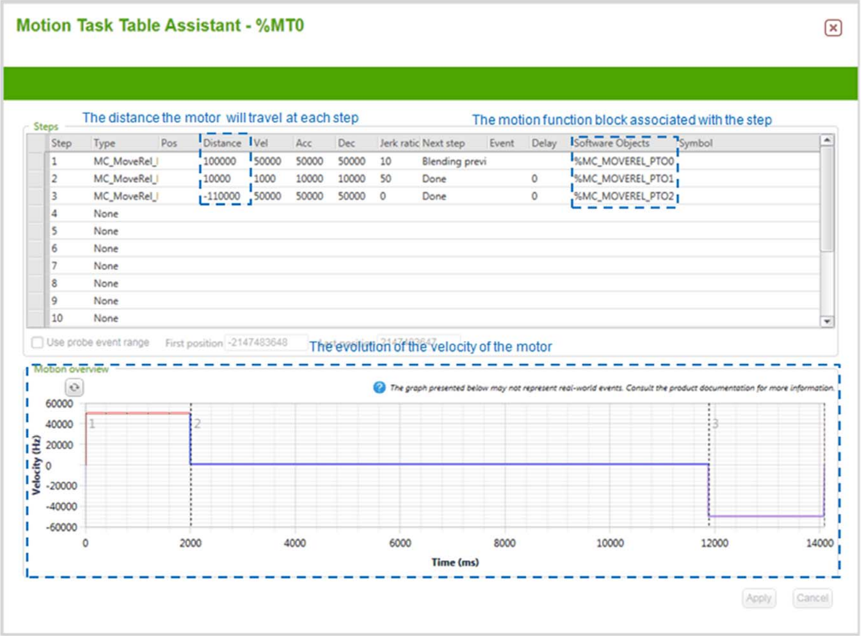
IN Axis: %PTO1
Table: %MT0
StartStep: 1
EndStep: 16

OUT ActiveStep: 0
ErrorId: 0 (No Error)

Motion Task Tables properties

Configured	Address	Symbol	Configuration
☑	%MT0		...
□	%MT1		...
□	%MT2		...
□	%MT3		...

This function block executes the steps **S0.3**, **S0.4**, and **S0.5** of the first Grafcet (SFC) state chart *(see page 28)*. Each step is configured within this function block, which implicitly calls the corresponding Motion function block.



Motion Task Table Assistant - %MT0

The distance the motor will travel at each step

The motion function block associated with the step

Steps

Step	Type	Pos	Distance	Vel	Acc	Dec	Jerk ratic	Next step	Event	Delay	Software Objects	Symbol
1	MC_MoveRel_I		100000	50000	50000	50000	10	Blending previ			%MC_MOVEREL_PTO0	
2	MC_MoveRel_I		10000	1000	10000	10000	50	Done		0	%MC_MOVEREL_PTO1	
3	MC_MoveRel_I		-110000	50000	50000	50000	0	Done		0	%MC_MOVEREL_PTO2	
4	None											
5	None											
6	None											
7	None											
8	None											
9	None											
10	None											

□ Use probe event range   First position -2147483648

The evolution of the velocity of the motor

Motion overview

The graph presented below may not represent real-world events. Consult the product documentation for more information.

When configuring this block, the time required for each step is calculated by SoMachine Basic, allowing you to set the speed and distance independently from each other.

4 - PTO Drill Rev

This POU controls the rotational speed of the servo motor wired to `PTO0`. The position of `PTO1` is read from the `MC_ReadActPos_PTO` function block. `PTO0` adjusts its velocity based on its position. When the drill is in between position 0 and position 100000, it rotates slowly. When the drill is between position 100000 and position 110000, it rotates fast and drills through the metal sheet.

5 - Error Management

This POU manages three error cases:

- `Rung0`: If the position limits are exceeded due to external interruptions, `%M200` is set to TRUE and the motors stop.
- `Rung2` and `Rung3`: These rungs reset the error when the **Stop** button is released.
- `Rung3`: If the position of the drill is not correct, the operator can bring the machine to the pause state and, after adjusting the position of the drill to 0, the **Set_Position** button can be pressed to set the position to 0 in the program.
- `Rung4`: If the drill is not advancing even though it should be drilling, the program waits for five seconds then stops the machine by setting `%M201` to TRUE.

6 - Monitoring

This POU gives information on the state of both PTOs. The output bits of the function blocks found in this POU can be used with an external HMI or can be used while commissioning the machine with SoMachine Basic.

The outputs of `Rung2` are used to determine whether the drill is advancing and for error management.

7 - Remote Control

This POU is an exact translation of the second Grafcet (SFC) state chart *(see page 28)* to the Ladder diagram.