Application Note

FESTO

YJKP - Manual movement

Description how to use manual functions YJKP

100263

Title ..... AppNote YJKP Manual movement
Version 1.10
Document no. 100263
Original ..... en
Author ..... Festo

Last saved 05.09.2019

Copyright Notice

Legal Notice

Table of contents

1 Components/Software used

Type/Name	Version Software/Firmware	Date of manufacture
Servo press kit YJKP	general	--
Application software YJKP (GSAY-A4-F0-Z4-1.3.5)	V1.3.5	--
Firmware controller (CECC-X)	V3.4.6	--
Firmware motor controller (CMMP-AS)	V4.0.1501.2.4	--

Table 1.1: 1 Components/Software used

1.1 Application description

This application note describes how to use manual movement and tare function via WebVisu and Host.

2 Prerequisites

Open a browser and start the WebVisu of the YJKP.

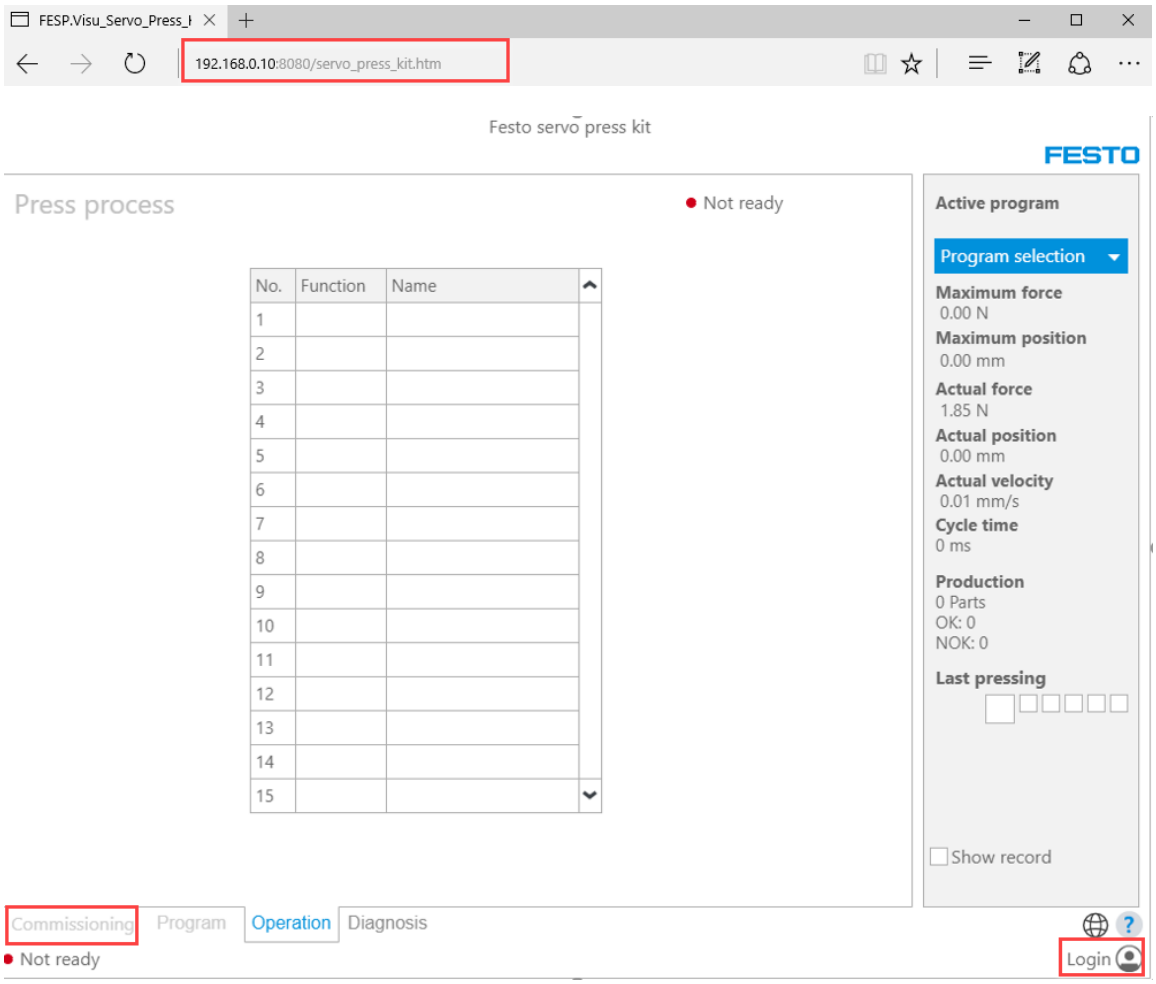
In Browser: ‹IP of the CECC-X›:8080/servo_press_kit.htm

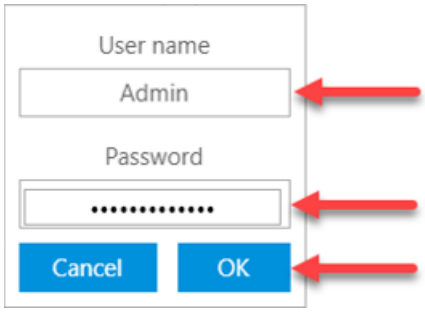
The visualization of the servo-press kit is opened with 4 tabs :

- Commissioning , Program : Not active
- Operation , Diagnosis: Active

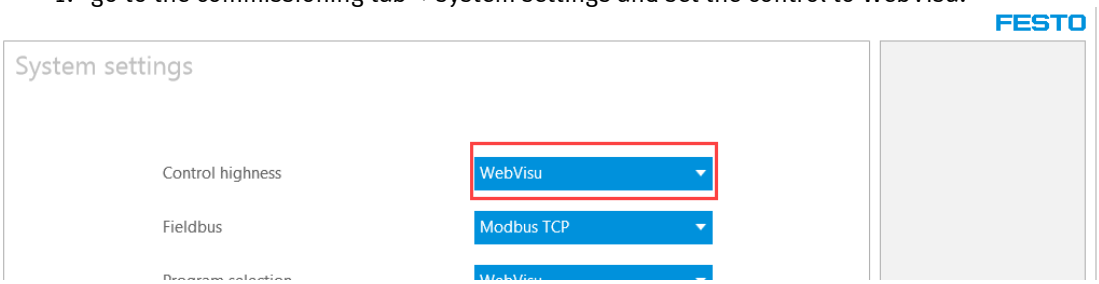
Click on **Login** to active the commissioning and program tabs. A new pop-up window appears and a password must be entered to login.

Default password: User name: Admin
Password: ServoPressKit





1. go to the commissioning tab -› system settings and set the control to WebVisu.

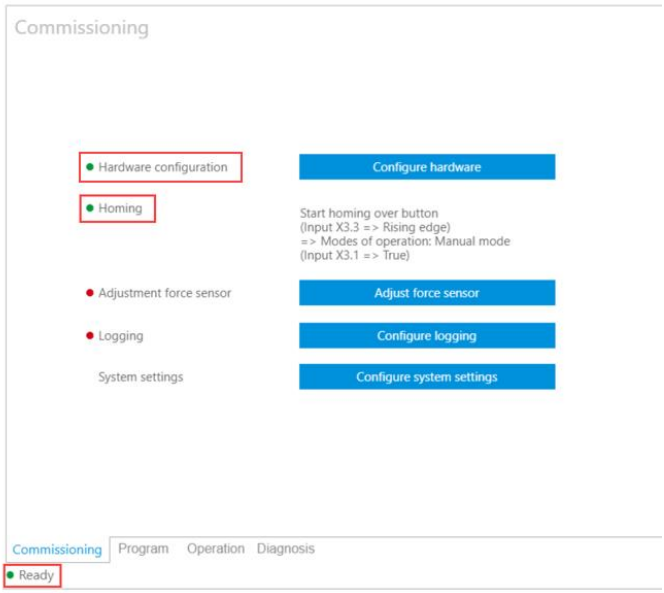


2. In commissioning tab check, if the hardware is configured successfully (= Green) and the system is homed (= Green). You will see a **Ready** in the bottom left corner.

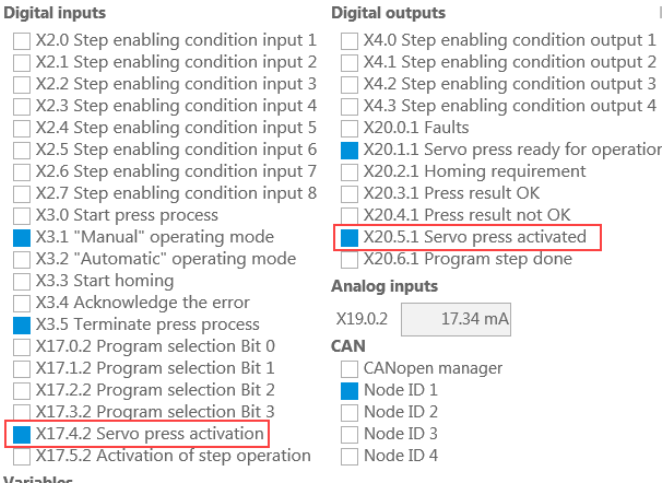
Note :

- For further information about hardware configuration and homing, please read AppNote Servo Press.

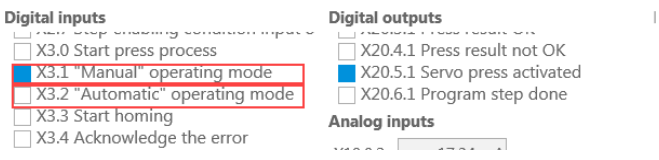
Kit YJKP Hardware Configuration and AppNote Servo Press Kit YJKP Homing.



3. Activate the servo press system with **input X17.4.2** . Output X20.5.1 will turn true.



4. Set the system mode to manual (Input X3.1 high, Input X3.2 low).

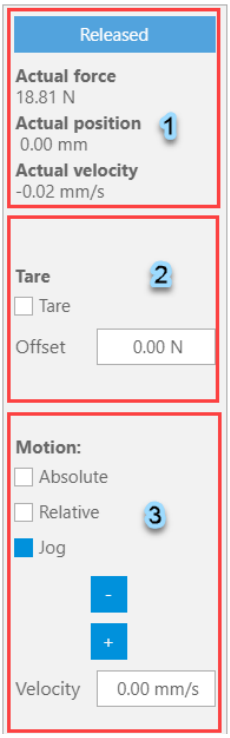


The servo press is now ready for a manual operation.

3 Manual control

3.1 Via WebVisu

The manual control panel consists of the following parts:



1. Cylinder status
   - Release(d): activation status of the system.
   - Actual force: current force value (N).
   - Actual position: current position of the cylinder (mm).
   - Actual velocity: current speed of the cylinder (mm/s).

2. **Tare:** This function is used to manipulate the actual force value.
   While commissioning the YJKP, you will notice that the actual force value in WebVisu might not be equal zero (as shown in the below figure). This value is mainly influenced by the mounting position (horizontal or vertical) and the weight of all mounted parts at the load cell. Thus when you need to press with a certain amount of force, the actual force needs to be reset before starting the pressing process.

   How to use it:

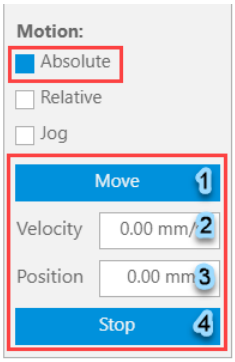
   - Check tare : Activate/Deactivate this function.
   - Offset : This value is added to the actual force value.

Example:
Actual force is 18.38 N, as shown in the previous figure.
After activating the tare function and with an offset of 0 N, the new actual force will be round about 0 N.

3. **Motion :** The control panel of the manual movement.

There are three methods in YJKP system to do a manual movement:

I. **Absolute movement:**
It is a method to move the cylinder to a target position based on the homing position of the cylinder.



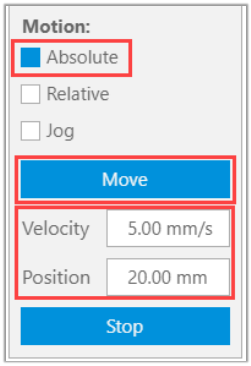
1. *Move* : starts the movement the cylinder.
2. *Velocity :* Cylinder speed when moving to the target position.
3. *Position* : target position(mm)
4. *Stop* : stops the movement of the cylinder.

Example:

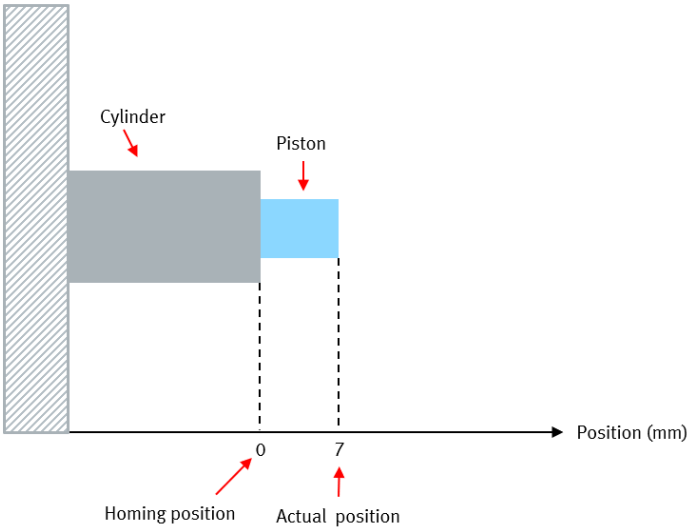
In this example the „Absolute“ method will be used to move the cylinder to the position 20 mm .

Set these values:

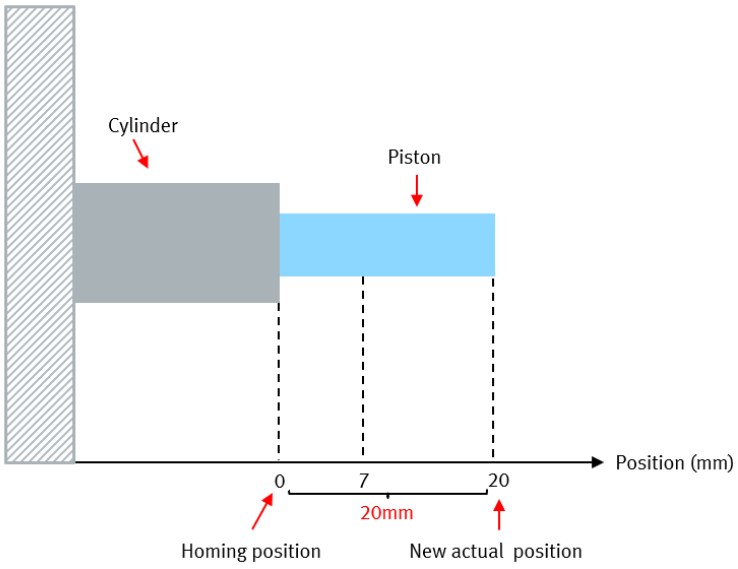
- Motion : Absolute
- Velocity : 5 mm/s
- Position : 20 mm



Startposition: Actual position = 7 mm

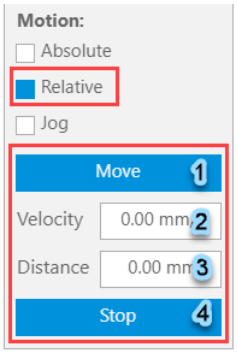


Click on "Move" button: The Cylinder will move from absolute position 7mm to absolute position 20mm (distance of 13mm) at speed of 5 mm/s. New actual position= 20 mm.



I. Relative movement:

It is a method to move the cylinder a certain distance based on the current position of the cylinder.



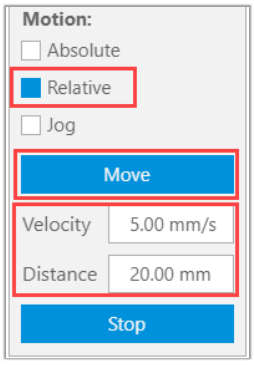
1. *Move*: starts the movement of the cylinder.
2. *Velocity*: Cylinder speed when moving to the target position.
3. *Distance*: target distance (mm)

4. *Stop* : stops the movement of the cylinder.

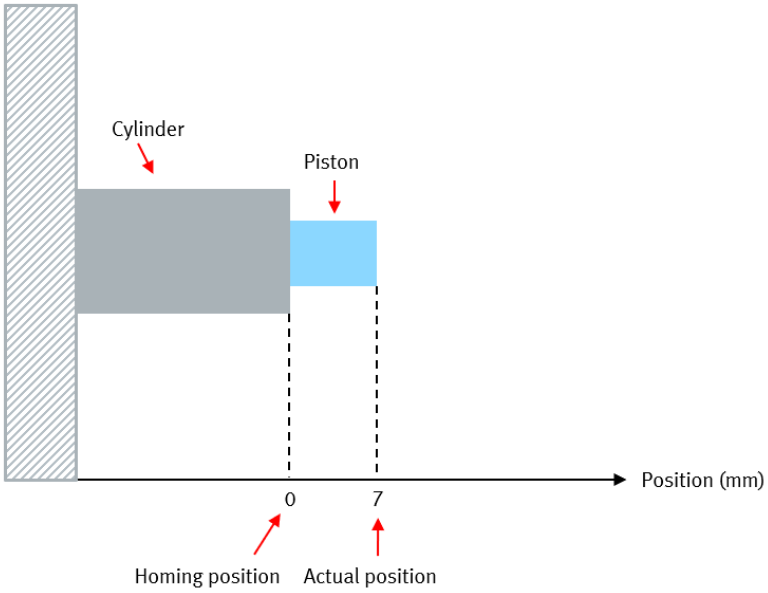
Example:
In this example the „Relative“ method will be used to move the cylinder 20 mm .

Set these values:

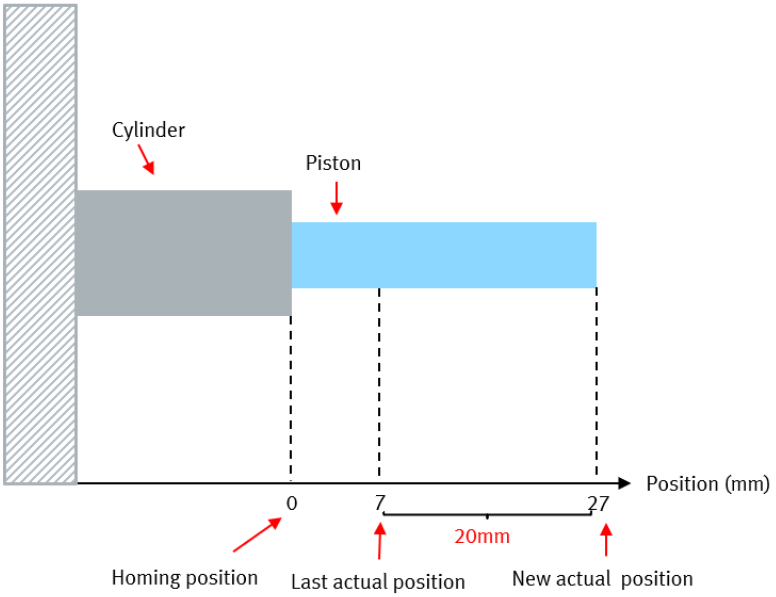
- Motion : Relative
- Velocity : 5 mm/s
- Distance : 20 mm



Startposition: Actual position = 7 mm

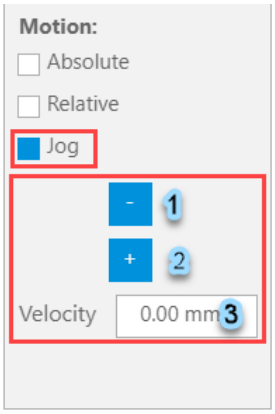


Click on “Move” button: The cylinder will move a distance of 20mm (from absolute position 7mm to absolute position 27mm). New Actual position = 27 mm.


Cylinder
Piston
Position (mm)
0
7
27
20mm
Homing position
Last actual position
New actual position


FESTO
Released
Actual force
358.37 N
Actual position
27.00 mm
Actual velocity
0.02 mm/s

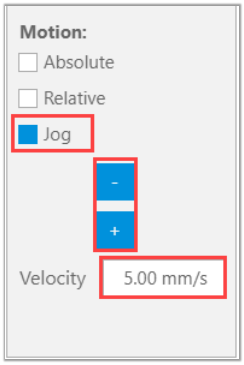
II. Jog movement:
It is a method to move the cylinder as long as you hold the movement buttons.



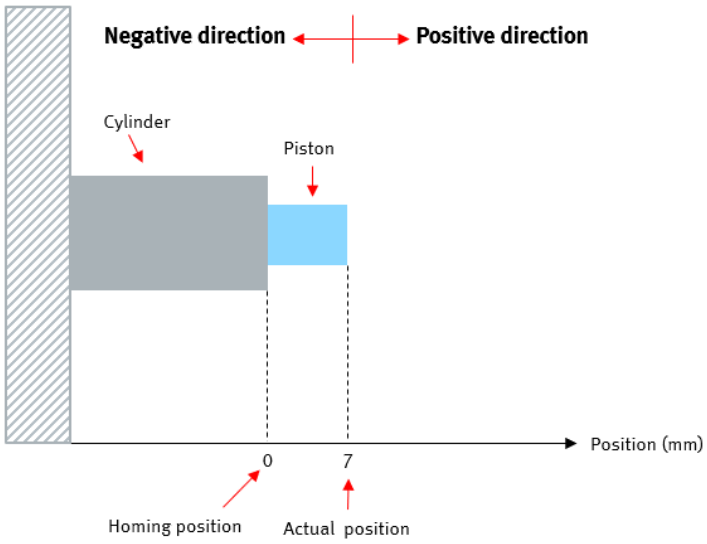
1. + : moves the cylinder in the positive direction.
2. - : moves the cylinder in the negative direction.
3. *Velocity*: Cylinder speed when moving.

Example:
Set this velocity value:

- Velocity : 5 mm/s



Click on the "**+** / **–** buttons" to move the cylinder in the positive/negative directions as shown in the figure :



3.2 Via Host PLC

This chapter will not describe all steps in particular, since they are basically the same like with the WebVisu (please refer to previous chapter).
It shows the needed function blocks and required inputs and outputs.

1. FB_Connect

   Required inputs:

   - xEnable := true;
   - enTargetComMode := 1;

   Required outputs:

   - xActive := true;
   - enActualComMode := 1;
   - xConnected := true;

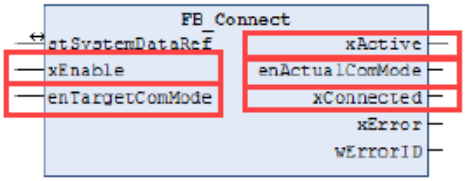


Figure 3-7: FB_Connect

2. FB_Manual

Required inputs:

- xEnable := true;
- xEnableSystem := True;
- xAbort :=True ;
- xTare:= True/False; *(manipulates the force value).*
- rTargetOffsetForceSensor:=; *(Target offset of the force sensor).*
- enTargetPosMode:=; *(Target position mode*
  0x00 = Jog
  0x01 = Move absolute
  0x02 = Move relative).
- rTargetMotionVelocity:= ; *(Cylinder speed when moving to the target position).*
- rTargetMotionPositionDistance:=; *(position/distance when using absolute/relative method)*
- xMove:= True/False; *(move the cylinder when using Absolute/Relative methods).*
- xStopMove:= True/False; *(stops the movement of the cylinder).*
- xJogPos:= True /False; *(moves the cylinder in the positive direction).*
- xJogNeg:= True/ False ; *(moves the cylinder in the negative direction).*

Required outputs:

- xActive := true;
- xSystemEnabled:=True;
- xSystemIsHomed :=True;
-