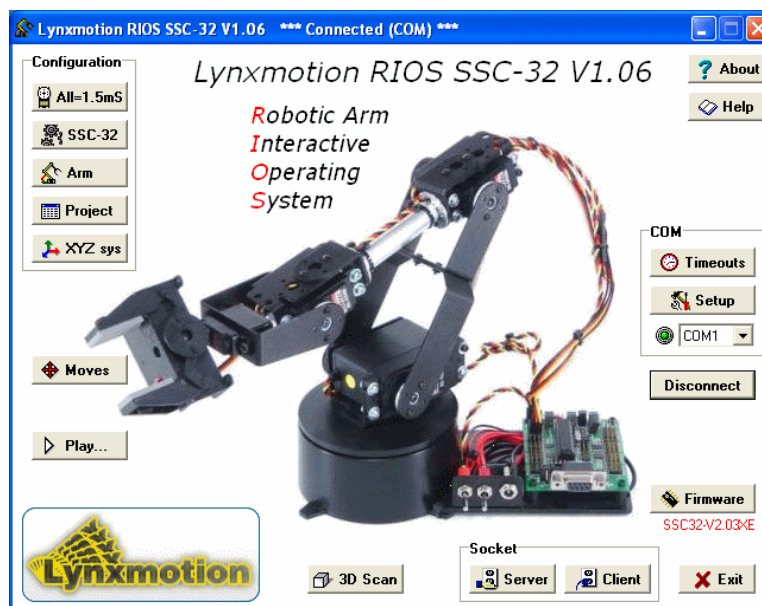# Lynxmotion RIOS SSC-32 V1.06

Robotic Arm
Interactive
Operating
System

Manual

Author and programmer : Laurent Gay - lynxrios@yahoo.fr

Lynxmotion RIOS SSC-32 V1.06 Manual

Safety First!

Read and understand the documentation associated with any of the tools used in the assembly of these kits. Work in a clean, well lit environment. Work slowly taking breaks often. Plan your work with plenty of extra time to avoid cramming to complete the project at the last minute. Lynxmotion, Inc. has taken every step to ensure the products sold are safe when used in a responsible manner. Therefore, Lynxmotion, Inc. can not be held accountable for irresponsible, careless or reckless behavior of the builder.

These kits are purely educational. Items sold by Lynxmotion, Inc. are not authorized for use in human contact, medical, life-saving, life-support, industrial or light industrial applications. Do not under any circumstances use these robots to move, touch, or handle dangerous or hazardous materials. Doing this could result in injury or death to the user, or damage to property.

Robots move without warning, wear eye protection at all times!

Lynxmotion, Inc. is not responsible for any special, incidental, or consequential damages resulting from any breach of warranty, or under any legal theory, including lost profits, downtime, goodwill, damage to or replacement of equipment or property, and any costs of recovering, reprogramming, or reproducing of data associated with the use of the mechanics, hardware or software it sells.

Lynxmotion RIOS SSC-32 V1.06 Manual

Table of contents

Lynxmotion RIOS SSC-32 V1.06 Manual

1 - Configuration of the SSC-32 card

Step 1. Install the RIOS program. Insert the CD into your CD drive and follow the instructions.

The controller card is NOT powered from the Serial or the Serial to USB cable. It will be necessary to apply servo power to complete the setup. Connect the Serial or the Serial to USB cable, wait for the system to recognize the card and run RIOS. If you run RIOS before the system recognizes the card, you will see the message box in Figure 1. Click 'Yes', and the program will enable the grayed buttons when the card is ready.
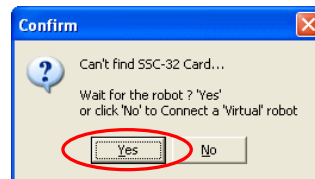
Figure 1.

If the card is not detected, select the right COM port number in the list box.

If you don't know the COM port number you are using :
- Just try from COM1, COM2 etc…
  when you find the right one, the card will auto-connect.

If the COM port you are using is not listed (USB to Serial cable issue) :
- Make sure your USB to Serial cable is connected.
- Wait for Windows to recognize it.
- Click on the 'Setup' button and select your COM port in the Setup form.
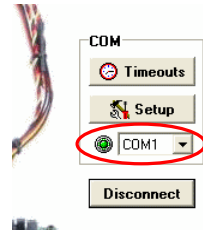
Check SSC-32 firmware is set to SSC32X-ER V1.06 or later.

Step 2. Test your arm.

Check your servos connections, refer to table 1.

SSC-32 pins / servos	
Pin 0	Base
Pin 1	Shoulder
Pin 2	Elbow
Pin 3	Wrist
Pin 4	Grip
Pin 5	Wrist Rotate
Pin 6	Extra Servo 7
Pin 7	Extra Servo 8

(Or Pin #16 and #17…
…see chapter 7.3)

Table 1.

SSC-32 Outputs	
Pin 8 to 15	Output 1 to 8

Table 2.

SSC-32 Inputs (Numeric or analog)	
Pin A to D	Input 1 to 4

Table 3.

Generate a 1.5mS pulse for all servos (neutral) :
- Click on the 'All=1.5mS' button.
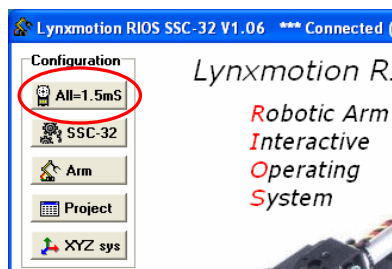
Figure 2.

- Click on the 'Test' button then 'Yes' on the warning message.
- **Check the arm appearance.**
- Click on the 'Stop' button to return to the previous arm state.

Set all servos to 1.5mS

Gripper is half open.

Base rotate is straight ahead.

Wrist rotate (L6) is aligned flat.

Wrist is in-line with forearm.

Elbow is 30° up from horizontal.

30°

Shoulder is vertical.

IMPORTANT!
With the servos set for 1.5mS the arm should look like the diagrams. If your arm looks like this it is ok to continue with the setup.

Figure 3.

Step 3. To enter the SSC-32 configuration screen, click on the 'SSC-32' button in the upper left corner.

Figure 4.

Lynxmotion RIOS SSC-32 V1.06 Manual

1 - Configuration of the SSC-32 card

Step 4. Click 'Yes' on this window.

Figure 5.

Step 5. Click 'No' on this window.

Figure 6.

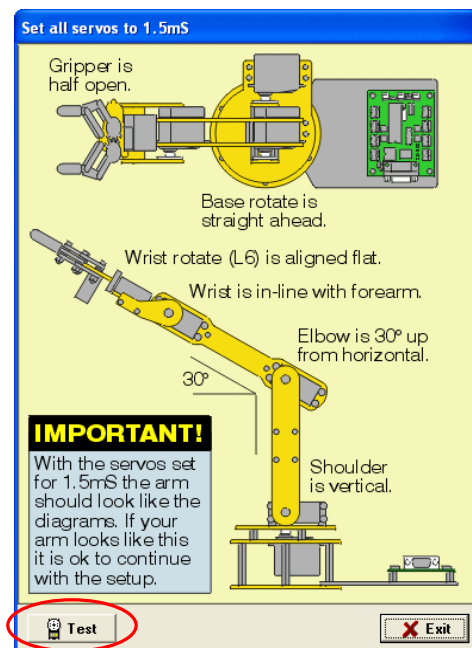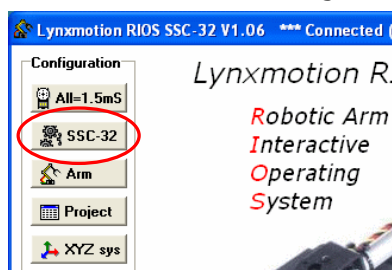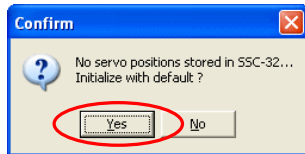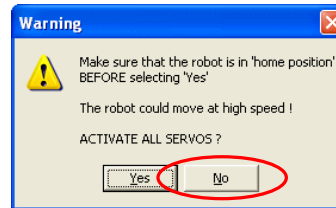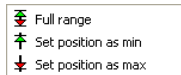**Step 6.** Make sure the boxes in the top row are NOT CHECKED then switch the robot power ON.
Refer to Table 7-1 for what each of the trackbars / sliders controls.

To quick adjust the 'Min Pos' and the 'Max Pos', move the slider to the position and right click to show the popup menu.

↕ Full range
↑ Set position as min
↓ Set position as max

Figure 7.

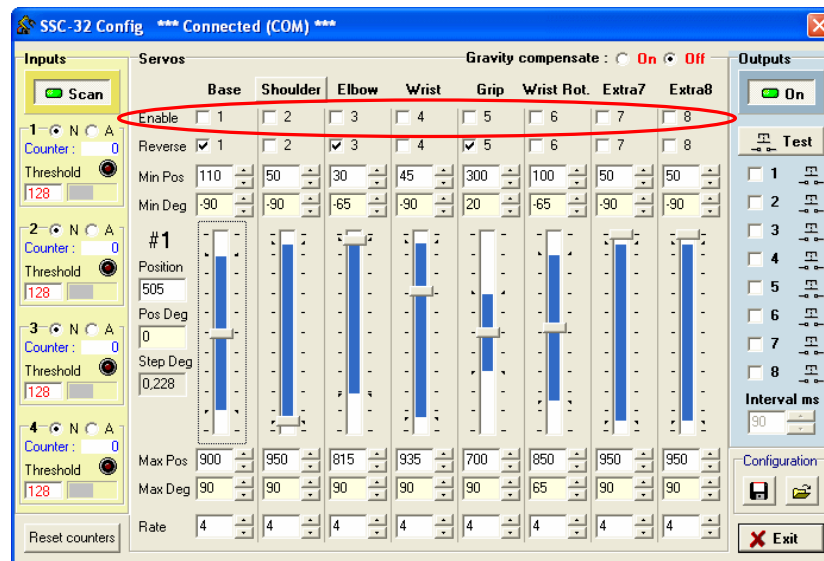| Trackbars / Sliders	
1	Base
2	Shoulder
3	Elbow
4	Wrist (Up/Down)
5	Grip
6	Wrist Rotate

Table 7-1

For now, DO NOT change anything to the 'Min Deg' and 'Max Deg' boxes.

Step 7. Configure the Base

Adjust Slider #1 to the middle.
Move the robot base CAREFULLY to the middle.
Check the 'enable' checkbox #1, and the base will move a bit!
Move slider #1 slowly to the top. The base will turn to the right.
When the slider is all the way at the top, the base must be at 90° to the right.
 - If less, decrease the 'Min Pos' box #1. It will allow you to push the slider a bit more to the top.
 - If more, increase the 'Min pos' box #1. It will push the slider down.
When finished, do the same for the left. It must be full left (90°).
Put the slider #1 at 0° (look at the 'Pos Deg' box when moving).

Step 8. Configure the Shoulder

Adjust Slider #2 to the middle.
Move the robot arm/shoulder vertically CAREFULLY and hold the robot forearm in your hand.
Check the 'enable' checkbox #2, the shoulder will move a bit!
Push slider #2 slowly to the top. The arm must be at the front of the robot and horizontal.
 - Adjust the 'Min Pos' box #2 if needed.
Push slider #2 slowly to the bottom. The arm must be at the rear of the robot and horizontal.
 - Adjust the 'Max Pos' box #2 if needed.
Keep the arm in this position (full rear).

Lynxmotion RIOS SSC-32 V1.06 Manual

1 - Configuration of the SSC-32 card

Step 9. Configure the Elbow

Adjust Slider #3 to the middle.
Move the robot forearm/elbow CAREFULLY vertically.
Check the 'enable' checkbox #3, the elbow will move a bit!
Push slider #3 slowly to the top. The forearm must be slightly touching (NOT crashed into) the arm Hex Spacer.
 - Adjust the 'Min Pos' box #3 if needed.
Push slider #3 slowly to the bottom. The forearm must be at the rear of the robot and horizontal.
 - Adjust the 'Max Pos' box #3 if needed.

Put the slider #3 at 0° (look at the 'Pos Deg' box when moving).

Step 10. Configure the Wrist (Up/Down)

Adjust Slider #4 to the middle.
Move the robot hand/wrist CAREFULLY vertically
Check the 'enable' checkbox #4, the wrist will move a bit!
Push slider #4 slowly to the top. The hand must be at the front of the robot and horizontal.
 - Adjust the 'Min Pos' box #4 if needed.
Push slider #4 slowly to the bottom. The hand must be at the rear of the robot and horizontal.
 - Adjust the 'Max Pos' box #4 if needed.

Put slider #4 at -40° (look at the 'Pos Deg' box when moving).
Put slider #3 at -65° (look at the 'Pos Deg' box when moving).
The forearm must be slightly touching (NOT crashed into) the arm Hex Spacer.

Step 11. Configure the Gripper

Adjust Slider #5 to the middle.
DON'T MOVE THE GRIPPER manually, leave it how it is.
Check the 'enable' checkbox #5, the gripper will move a bit!
Push slider #5 slowly to the top. The gripper must be fully opened.
 - Adjust the 'Min Pos' box #5 if needed.
Push slider #5 slowly to the bottom. The gripper must be fully closed.
 - Adjust the 'Max Pos' box #5

Put slider #5 at 57° (look at the 'Pos Deg' box when moving). The gripper should be half opened.

Step 12. Configure the Wrist (Rotate)

Adjust slider #6 to the middle.
DON'T MOVE THE WRIST ROTATE manually, leave it how it is.
Check the 'enable' checkbox #6, the grip rotate axis will move a bit!
Push slider #6 slowly to the bottom. The Gripper should turn left to 65°.
 - Adjust the 'Max Pos' box #6 if needed.
Push slider #6 slowly to the top. The Gripper should turn right to 65°.
 - Adjust the 'Min Pos' box #6 if needed.
If you want, enter your own 'Min Deg' and 'Max Deg' and adjust the Min Pos and Max Pos to allow the robot to reach your own values. For example, Min Deg = -90° and Max Deg = 90°.
Just be careful to not move the servo past its mechanical limits!

Put the slider #6 at 0° (look at the 'Pos Deg' box when moving).
The gripper rotate axis should be flat (the two fingers at the same height).

Step 13. Save your Progress

The robot should look like Figure 8. If so, click on the
'Save' button. This will be the robot's default position.
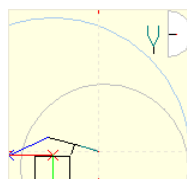

Figure 8.

Lynxmotion RIOS SSC-32 V1.06 Manual

2 - Configuration of the Arm geometry

All angles are in degrees and all distances are in centimeters (1 inch = 2.54 centimeters).

The default values are for an L6 arm.

Switch the robot power OFF before changing geometry values or the robot will move/crash without warning.

If you use a custom arm please measure each section as described, else use the quick arm selector

	From	To
Base		
Height	Tabletop	Shoulder Axis
Radius	Base axis	Base front
Length		
Forearm	Wrist axis	Elbow axis
Arm	Elbow axis	Shoulder axis
Hand/grip		
Max Open (Grip fully opened)	Inside of one finger	Inside of other finger
Max length (Grip fully closed)	Wrist axis	A finger tip
Servo Height	Wrist rotate servo height	

The 'Neutral' values are relative to the part which holds it:
This is not the servo's neutral, but a geometry shift when all joints are at 0° (due to SSC-32 card configuration).

- Shoulder/Arm neutral is relative to horizontal.
- Elbow/Forearm neutral is relative to Arm.
- Wrist/Hand neutral is relative to Forearm.
 You may not have to change these values.

Click on the 'Save' button if you have made changes.

You can use Trackbars to test moves:
 - X, Y and Z perform motions in straight lines and in parallel with X, Y and Z axis.
 - Distance performs a motion in a straight line in a radius away from base axis.

The arm display window auto size, according to arm geometry.

RIOS V1.06 now includes a quick arm selector.

Select an arm, a base, a gripper and a Wrist rotate system then click "OK".
Don't forget to click on the 'Save' button if you have made changes.

Selecting an AL5 or SES arm will auto enable the little gripper support.

Lynxmotion RIOS SSC-32 V1.06 Manual

3 - Configuration of the gravity compensate

Here, you may just have to change the weight of an object that the arm is going to be lifting.

You can modify the geartrain play compensate with the sliders.
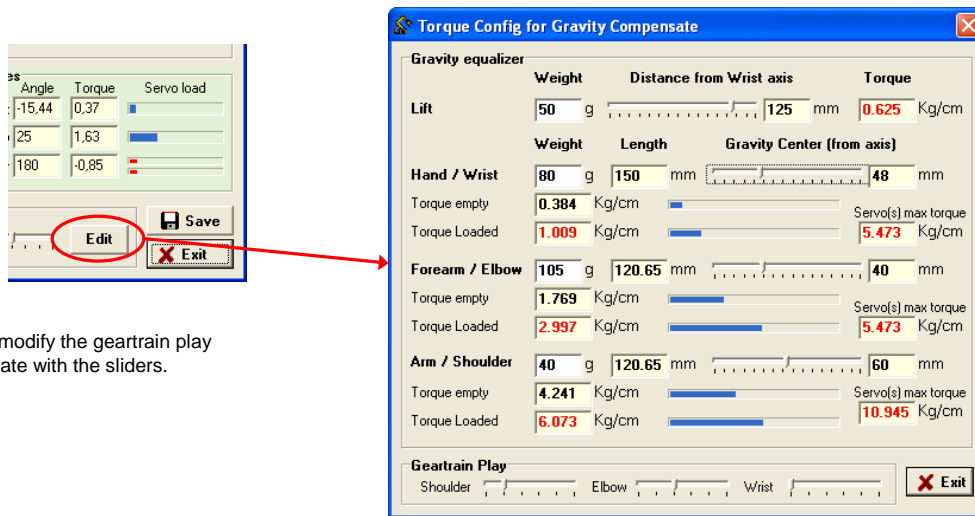
All default values are set according to a L6 arm, we will furnish support for other arm geometries soon.
For now, these values are working fine to generate the gravity compensate for any kind of Lynxmotion's arms, even if the torques are not 100% accurate, the similarity between arms proportions makes this working fine.

Adjusting the gravity compensate effect with the "Empty" and "Load" sliders will be sufficient in all cases.

Click on the 'Save' button on the previous form if you have made changes.

4 - XYZ coordinate system

If you want to use a different XYZ coordinate system than the RIOS default, this is the place.
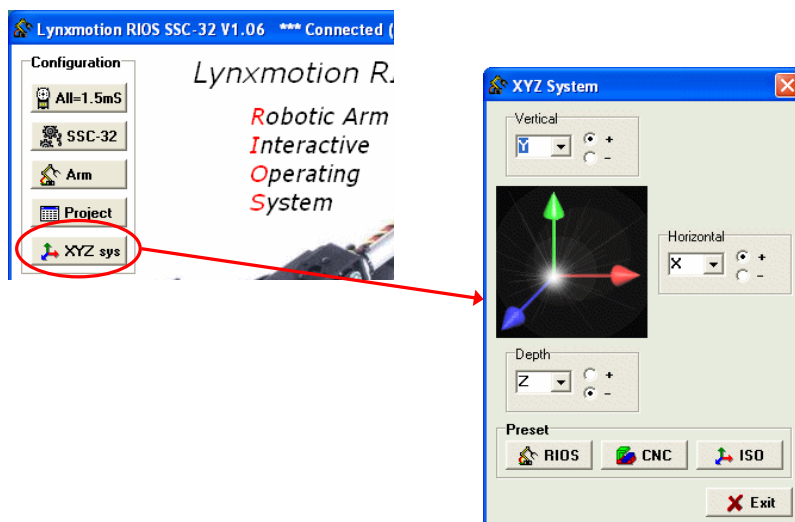
Select X, Y or Z for each axis

Then the axis direction (sign)

For example, as shown:

The Horizontal axis is the **X** here
and values increase (**+**)
in the direction of the **red arrow**

The Depth axis is the **Z** here
and values decrease (**-**)
in the direction of the **blue arrow**

Click on a Preset button to use an existing system, or define your own.

Preset	Axis		
	Horizontal	Vertical	Depth
RIOS	X+	Y+	Z-
CNC	Y+	Z+	X+
ISO	X+	Y+	Z+

All changes here are auto saved and only affect arm coordinates displayed in RIOS forms.
The CSV Export function will use and save RIOS default Coordinate system no matter what, to preserve compatibility.

Lynxmotion RIOS SSC-32 V1.06 Manual

5 - The 'Moves / Motions' module

5.1 Moving robot. Mouse
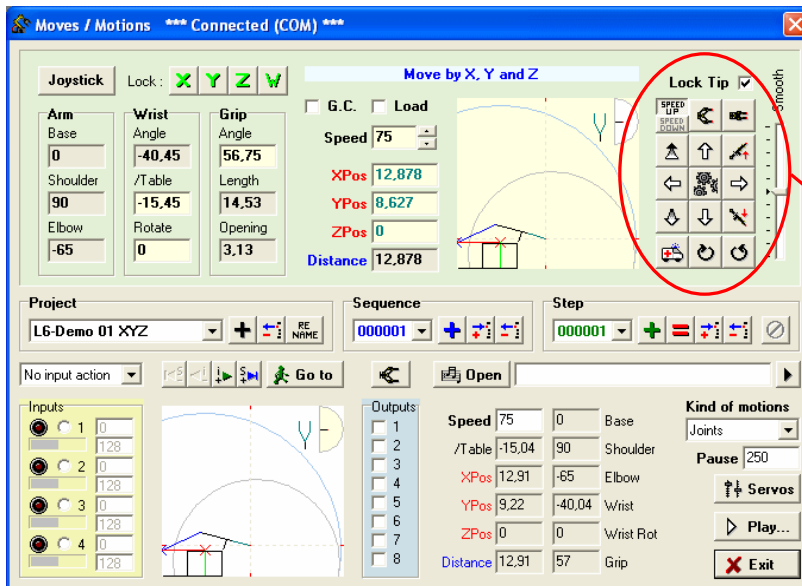
There are 3 kind of moves:
- X, Y and Z.
- Distance, Y, Base angle.
- Joint.

Swap kind of moves by clicking on the gears.

Use 'Smooth' slider to eliminate shakes

This is the X, Y and Z panel.

Distance, Y and Base angle panel.
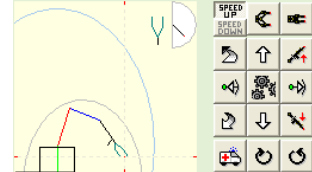
Joint panel.

The 'Ambulance' icon is an emergency stop, it disables/enables torques on all servos.

The 'Lock Tip 'Checkbox' enable special moves with the Tip of the gripper locked on XYZ grid.

The 'Speed up/down' buttons are a shortcut to increase/decrease speed by 50.

Click on the robot picture to swap the robot view, the icons will follow…right click to enable/disable bold view   **Bold view**

Check the 'Load' checkbox to see an object in the grip. It also swaps between the 'empty' and the 'load' gravity compensate.
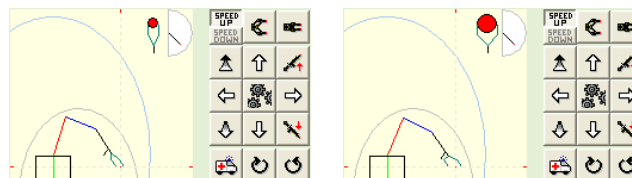
See Chapter 4.3 to learn more about the "Load" button.

Lynxmotion RIOS SSC-32 V1.06 Manual

5 - The 'Moves / Motions' module

5.2 Moving robot. Joystick

Plug a joystick and the grayed button will auto-enable.

Click on the X, Y, Z, W to lock a joystick axis.
Click on the joystick button to show the following window.
You can use a PlayStation 2 controller to move and program the arm (with an USB adaptor).
Test buttons and calibrate the 'dead zone' (red box):
- While the joystick positions (black spot) are in the red boxes, the robot won't move.
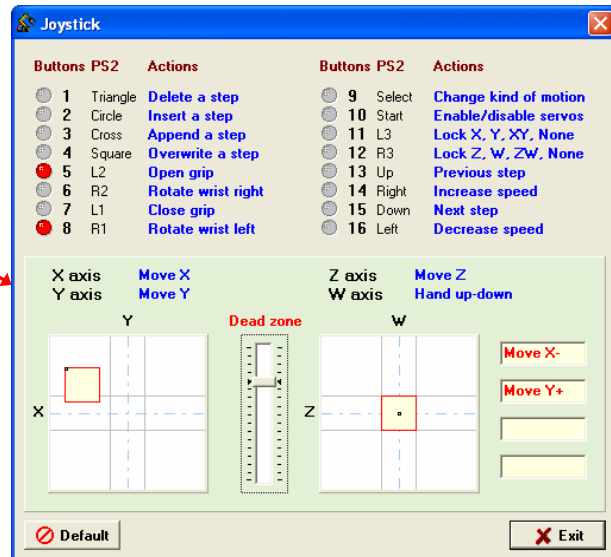
Right-click on an action to open a context menu.

You can change the default button's assignment selecting a new one in the context menu

Sub menu groups "Database" commands and "Move" actions.
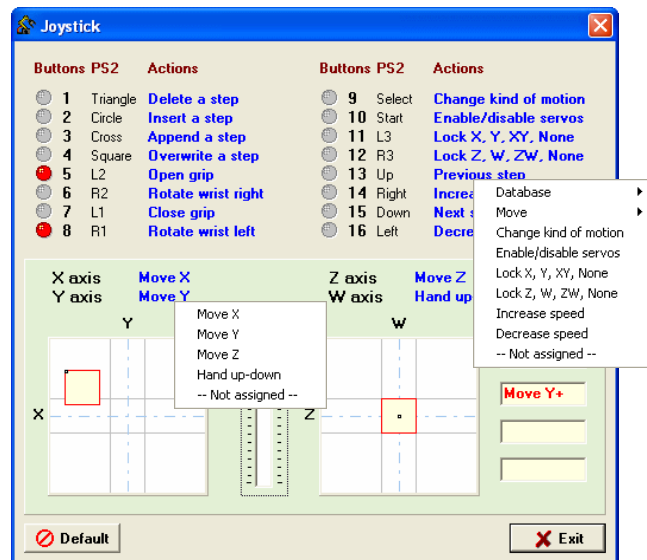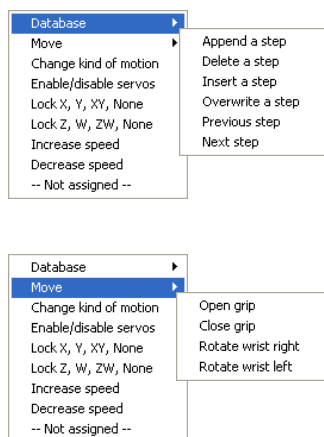
Lynxmotion RIOS SSC-32 V1.06 Manual

5 - The 'Moves / Motions' module

5.3 Storing data. Robot position

Projects contain Sequences, which contain Steps.

You can store the current robot position as a 'Step' using using the 'plus sign' icons :
 - The Project black 'plus sign' icon will append this step in a new Project.
 - The Sequence blue 'plus sign' icon will append this step in a new Sequence of the current Project.
 - The Step green 'plus sign' icon will append this step in the current Sequence of the current Project.

"Undo" button.

Projects :
 - You can 'Append', 'Delete' or 'Rename' Projects.
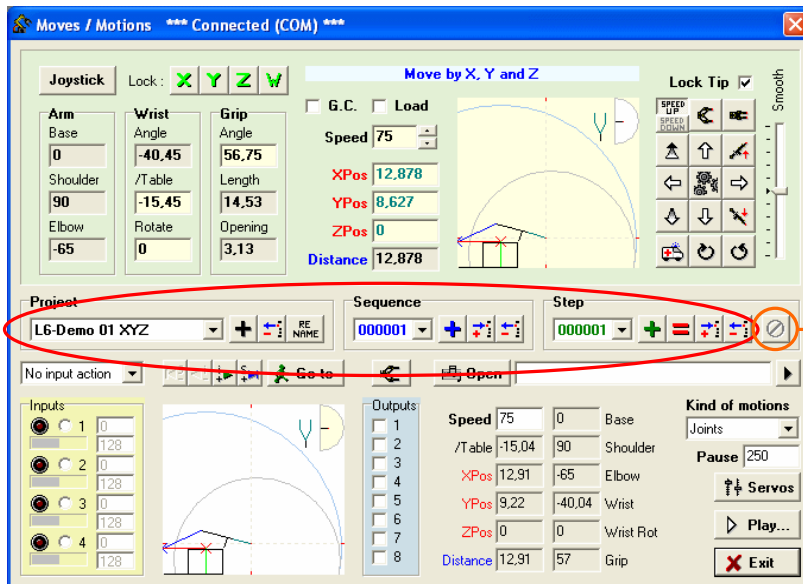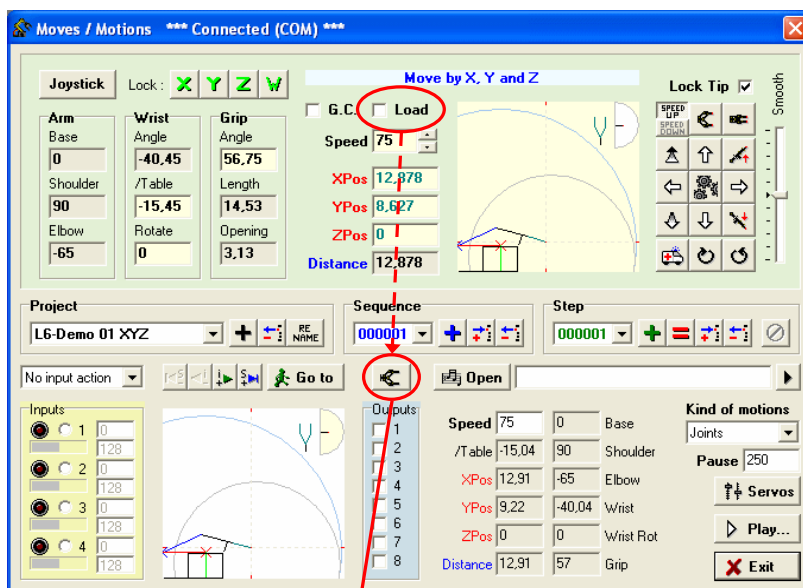For more about Projects, see the 'Project module'.

Sequences :
 - You can 'Append', 'Insert' or 'Delete' Sequences.

Steps :
 - You can 'Append', 'Overwrite', 'Insert' or 'Delete' Steps.

Moves part.
Here you can move the arm.
The graphics shows the current arm position.

Database part.

When steps are stored in the Database :

- You can edit them, browsing with the arrows or list-boxes, then press "goto" to apply to the "Move part" side, then adust the position with buttons and press overwrite (red '=' sign).

- You can change directly in the database : The **speed**, the **kind of motions**, the **pause**, the **input action**, the input **threshold** (if analog), **outputs**, **Grip** state, **media** to play or **comment**. DON'T PRESS OVERWRITE IN THESE CASES. **The graphics shows the stored arm position.**

When you save a step to the database, the "**Load**" check-box state is saved too,
showing a red object in the grip and applying its state to the "**Grip**" button.
You can change this state directly in the database selecting a step and using the "**Grip**" button.

What does it do ?

- It swaps between "Load" and "Empty" gravity compensate effect.

- If state is "Load" (button down), it ignores all grip close/open moves in order to allow a special function used in the "**L6-Demo 09 Grip Close**" :
 - Select this Demo in the Moves / Motions form, then select the step 000003, the arm is ready to grab an object, gripper is full open.
 - Now select the step 000004, this move slowly and fully closes the gripper, an input action is defined, expecting input #1 to occur to "stop this step".
 Warning, it will stop the move to the next interpolated step (see 5.5), if interpolation interval is too big it won't be able to stop quickly.
 - Now browse steps 000005 to 000007, these moves are carrying the object, the "Grip" button is down so the Grip position stored in these 3 steps won't be used => these 3 steps will be performed using the last grip position : the one stopped by the input #1 in the step 000004.

This feature allows to use a pressure sensor or a switch in the gripper to stop the grip while closing, then carry the object
discarding all further grip moves with the "Grip" button pushed (useful when you don't know the object size).

- See now step 000008, this step fully opens the grip to drop the object, but now the "Grip" button is release to allow this grip move.

To try this feature, go to the "Play" form and select the "**L6-Demo 09 Grip Close**" project, press the project's Play button and
wait for the Step 000004 to start, it will say in the white status-bar : "Waiting for input #1 to stop this move", press the input #1 button to simulate
a switch/pressure sensor and the grip close will stop (see the warning above) then, depending on the moment you pressed the button it will carry
a larger or smaller object in the gripper.

Lynxmotion RIOS SSC-32 V1.06 Manual

5 - The 'Moves / Motions' module

5.4 Storing data. Features

Use the combo-boxes to select a Project/Sequence/Step or use the four S-, I-, I+ and S+ icons to navigate through Sequences and Steps.
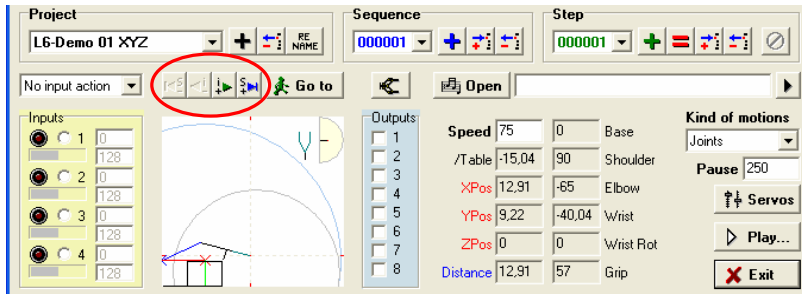
'**Open**' icon allows you to add a media to this Step, music or video, or you can type a little comment.
Test the media with the black triangle icon.

Speed is the one to reach this step.
Pause (ms, 1000ms = 1 second) will be performed at the end of Step, recommended value is 250 or more.

Check the **Outputs** check-boxes to set them On/Off (applied when starting the Step).
(Outputs 1 to 8 are Pin 8 to 15 on the SSC-32)

Kind of motion :

- "**Joint**" will perform curve trajectories and allows to move one joint only for example.
all "Home" positions should use "Joint" moves, it's the only one kind of motion which won't hang in case of a joint
is going to its limits during a travel and locks the move (red cross on joint).

- "**XYZ**" will perform straight lines trajectories on the three axes.

- "**Dist, Y, Base**" will perform straight lines trajectories except the base rotation.
Using this kind of motion, the arm is carrying his "X axis" when the Base is moving,
so the X axis is renamed "Distance" (distance from base axis to gripper tip),
then we are no more talking about "Z" axis but a base rotation angle rather.
To be short, it performs straight lines trajectories in the vertical plan (Distance (from base axis) and Y (height))
and use a Base angle to rotate this plan around the vertical axis.

The Moves panel will let you familiarize with this three kind of moves (use the gears icon to swap kind of moves)

Input actions :

Select an input action if needed:
'Wait for' will wait for an input to occur or a counter to reach a value before starting the step.
'Pause/Play' will Swap between Pause and Play if an input occurs or a counter reaches a value.
'Stop project' will stop the current Project if an input occurs or a counter reaches a value.
'Stop this Step' will stop the current Step if an input occurs or a counter reaches a value and go to the next Step.

Select an input to test and set a counter value
This version of RIOS allows only 'Relative' counter values and not 'Absolute'.
So, if the counter value is set to '3', for example, the input selected must occur 3 times DURING the Step, no matter what the
counter was at before the step.

You can also store the 'Extra servos' #7 and #8 positions in a Step:

Click on the 'Servos' button,
select new Positions and Rates,
store information in the current Step by clicking on the 'Equal' icon (it updates only this new information).
Trackbars show the current servo position, not the stored servo position (if you click on 'Store' they will be
the same), the stored values are in the grey boxes.

These moves are NOT synchronized with the Steps. They will start with a Step, but they can finish the move
before or after the step ends... or never!
You can adjust the rate or add a pause in the Step to ensure the Extra servos' moves will finish during a step.
If the same positions are set during several Steps, the moves will start on the first Step and can end on another.
If an Extra servo's move is not finished when going to a new Step with new values, the Extra servo will abort
the current move to go to the new positions.

You can swap between the 'Move / Motions' window and this window without closing it.
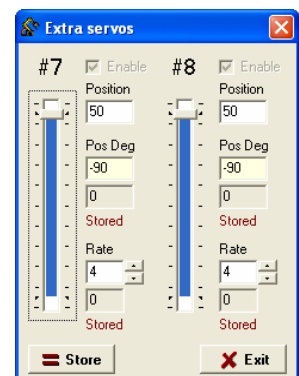
Lynxmotion RIOS SSC-32 V1.06 Manual

6 - The 'Play' module

6.1 Play. Features
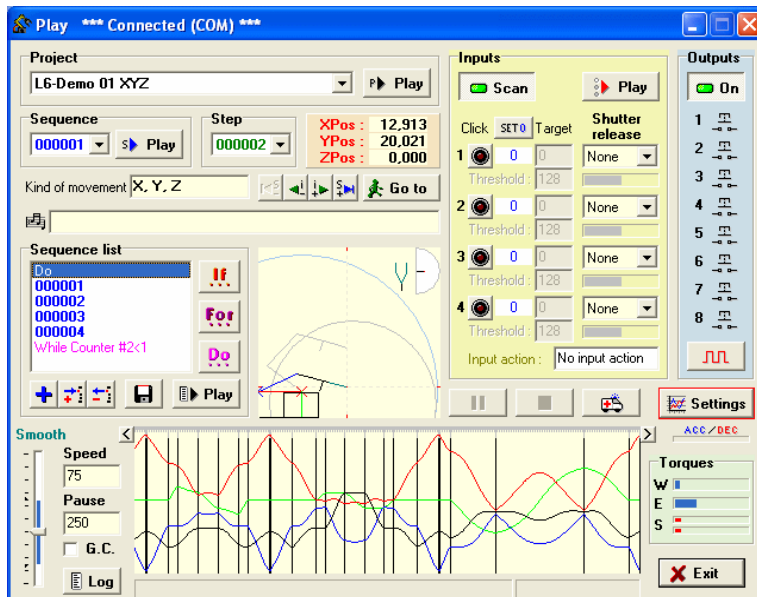
Real time play log.
Use 'Smooth' slider to eliminate shakes

Real time torques display.
Outputs configuration with delays, durations and blink speeds.

Play a Project or a Sequence.

Select a position with the S-, I-, I+ and S+ icons.
'Go to' the position you've selected.

Select 'inputs as starter for sequence' and push the 'Scan' button, if a corresponding input occur, it will Play the Sequence selected (only if there's nothing already playing).

Build a Sequence list and Play it.
Append If-Then-Else structure.
Append For-Next and Do-While loops.
Break a loop with If-Break instruction.
Save a Sequence list by Project.

Click the robot picture to change the view.
The robot picture shows the current position and a ghost of the selected or next Step in Play mode.

See the Base, Shoulder, Elbow and Wrist trajectories... with scrolling!

Trigger inputs and watch outputs turn on and off.

6.2 Outputs options.

Delay :
When a Step sets an output on, this allows a delay before the output is really set on.

Duration :
When an output is really set on, it will auto set off after the Duration period (if duration = 0 it will not auto set off).

Delay and duration are in ms (1000 ms = 1 second) and will run across steps if the next steps also set output on.
Delay and duration do not run after a 'Play' is ended, the outputs will stay in their current states.

Speed :
If you need an output to blink, set a speed > 0. 1 is the slowest and 20 the fastest, 0 is no blink.
Click on the test button to perform the Delay, then the blink/set on, and after the duration, the auto set off.

Click on the 'Signal' button in the Outputs panel to show this window.
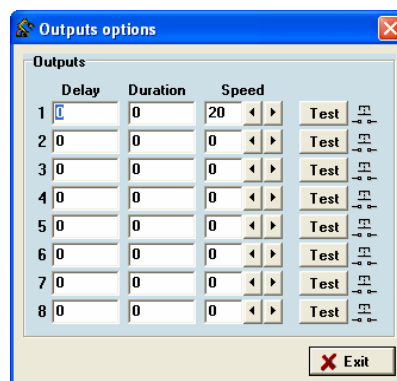
(Outputs 1 to 8 are Pin 8 to 15)

6.3 Play log.

Inputs events :
Push this button if you want to log all the inputs events (set on, set off). If you are using big counters, you may not activate this.

'Thread' means that the message was sent by the separate 'Play thread' (Task).

click on the 'Log' button to show this window.

The playlog.txt file is auto saved in the installation directory.
This window is sizeable.
You can swap between the 'Play' window and this window without closing it.

Lynxmotion RIOS SSC-32 V1.06 Manual

6.4 Sequence list.

Select a Sequence in the Combo-box,
and click on the 'plus sign' to append,
or on the 'right arrow and the little plus sign' to insert.
To delete a sequence (from list only), select a sequence in the list and click on the 'left arrow with a minus sign'.
You can only delete sequence with the delete icon, not the structures words.

The Sequence list contains Sequences from one Project only.

6.4.1 The For-Next loop

There's a better way to repeat a group of lines (Sequences, other loops and structures) than the Sequence list below!

Repeat a group of lines from 1 to 999 times between the For and it's Next.
Use as many For-Next loops as you need.

You can nest up to 10 For-Next loops max.

Abort a loop (the current) prematurely with the 'If-Break' structure.

Example

To repeat the group of lines 'Sequence 000002 and 000003' four times:

Click on the 'For' button

Enter '4' in the 'Loop' box,
select the sequence 000002
(insertion point),
push the 'For' button,
click on 'Insert' icon,
push the 'Next' button,
select the sequence 000004
(insertion point),
click on 'Insert' icon,
click 'Exit'.

Playing this sequence list will perform Sequences 000001, 000002, 000003, 000002, 000003, 000002, 00003, 000002, 000003 and 000004.
Open the Play Log file and look at the details, the log file is a good way to teach yourself...

6.4.2 The Do-While loop

Need to repeat a group of lines while an input is On or while a counter (input counter) is less than a value?
The condition test is at the end of loop, so the content of the loop will be perform one time at least, even if the end test fails.
When entering a 'Do-While' loop the current counter position is stored, so the counter value to test is an increment (relative).

Example

To repeat the Sequence 000004 while the input counter #1 is less than 10:

Click on the 'Do' button

Select the input #1
Enter '10' in the counter box,
select the sequence 000004
(insertion point),
push the 'Do' button,
click on 'Insert' icon,
push the 'While' button,
click on 'Append' icon,
click 'Exit'.

Playing this sequence list will perform Sequences 000001, 000002, 000003 and 000004…(000004) while the input counter #1 is less than 10.
Whatever counter #1 was when entering the 'Do-While' loop it must be increased by 10 to exit the loop.
Abort a loop (the current) prematurely with the 'If-Break' structure.
If you don't have input #1 connected, you can push the input #1 button on the Inputs panel.
Use as many Do-While loops as you need, and you can nest up to 10 Do-While loops.

Lynxmotion RIOS SSC-32 V1.06 Manual

6 - The 'Play' module
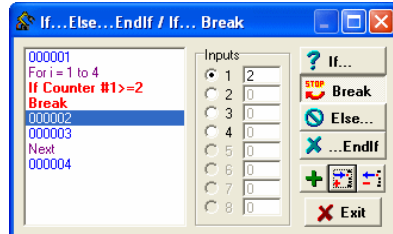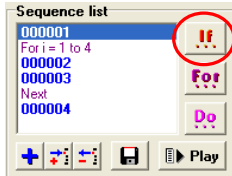
6.4.3 The If-Break Structure

Example

To prematurely Break the 'For-Next' loop example if the input counter #1 reaches 2 or more:

Click on the 'If' button

Select the input #1
Enter '2' in the counter box,
select the sequence 000002
(insertion point),
push the 'If' button,
click on 'Insert' icon,
push the 'Break' button,
click on 'Insert' icon,
click 'Exit'.

Playing this sequence list will perform Sequence 000001, test the input counter #1: if >= 2 it breaks the loop and performs the sequence 000004, else it performs the sequences 000002 and 000003 and loops 3 times again (testing input counter #1 before each Sequence 000002).

So if the input counter #1 is already >= 2 when entering the 'For-Next' loop, the If-Break structure will exit the loop immediately and continue to the 000004.
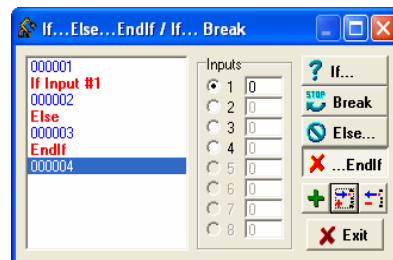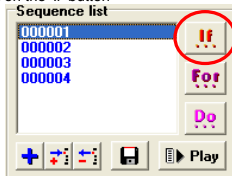The 'If-Break' structure allows a more complex syntax, see the 'All If...Structures'.

6.4.4 The If-Else-Endif Structure

Example

Plays the Sequence 000002 if input #1 is on, else play the 000003.

Click on the 'If' button

Select the input #1
select the sequence 000002
(insertion point),
push the 'If' button,
click on 'Insert' icon,
select the sequence 000003
(insertion point),
push the 'Else' button,
click on 'Insert' icon,
select the sequence 000004
(insertion point),
push the 'Endif' button,
click on 'Insert' icon,
click 'Exit'.

Playing this sequence list will perform Sequence 000001, test the input #1: if On it performs the sequence 000002 else it performs the sequence 000003, and it performs the Sequence 000004 in any case.

Lynxmotion RIOS SSC-32 V1.06 Manual

6 - The 'Play' module

6.4.5 All If… Structures

If-(else)-Endif	**If-(else)-Break**
Perform the line(s) between the 'If' and the 'Endif' if Input #x is On (or the counter #x >= n)	If Input #x is On (or the counter #x >= n), break the current loop (Do-While or For-Next)
…	**…**
If Input #x (or Counter #x >= n)	**If Input #x (or Counter #x >= n)**
…	**Break**
Endif	**…**
…	

Perform the line(s) between the 'If' and the 'Endif'
if Input #x is On (or the counter #x >= n)
…
If Input #x (or Counter #x >= n)
…
Endif
…

If Input #x is On (or the counter #x >= n),
break the current loop (Do-While or For-Next)
…
If Input #x (or Counter #x >= n)
Break
…

Perform the line(s) between the 'Else' and the 'Endif'
if Input #x is Off (or the counter #x < n)
…
If Input #x (or Counter #x >= n)
Else
…
Endif
…

If Input #x is On (or the counter #x >= n),
perform the line(s) between the 'If' and the 'Break',
and break the current loop (Do-While or For-Next)
…
If Input #x (or Counter #x >= n)
…
Break
…

Perform the line(s) between the 'If' and the 'Else'
if Input #x is On (or the counter #x >= n),
else perform the line(s) between the 'Else' and the 'Endif'
…
If Input #x (or Counter #x >= n)
…
Else
…
Endif
…

If Input #x is On (or the counter #x >= n),
perform the line(s) between the 'If' and the 'Else',
else break the current loop (Do-While or For-Next)
…
If Input #x (or Counter #x >= n)
…
Else
Break
…

If Input #x is On (or the counter #x >= n),
perform the line(s) between the 'If' and the 'Else',
else (perform the line(s) between the 'Else' and the 'Break' and
    break the current loop (Do-While or For-Next)).
…
If Input #x (or Counter #x >= n)
…
Else
…
Break
…

If Input #x is Off (or the counter #x < n),
(perform the line(s) between the 'Else' and the 'Break' and
    break the current loop (Do-While or For-Next)).
…
If Input #x (or Counter #x >= n)
Else
…
Break
…

If Input #x is Off (or the counter #x < n),
break the current loop (Do-While or For-Next)).
…
If Input #x (or Counter #x >= n)
Else
Break
…

Lynxmotion RIOS SSC-32 V1.06 Manual

6 - The 'Play' module

6.5 Advanced settings.

See interpolated step marks on the Play graphics.

Set all values to default.

"L6-Demo 03 Joint" example

Checking the "**Show trajectories**" box will display a 100 pixels long trajectory while playing.
It shows one dot per interpolated steps, making difference between joint and XYZ moves visible.

You can notice too the acceleration/deceleration effect
(dots getting closer each others means deceleration, opposite means acceleration).

Interpolation.

RIOS computes interpolated steps (semi key steps) between the ones stored in the database (key steps) in order to fully control trajectories :
 - interpolation allows to perform straight lines trajectories (using the grip tip as a reference).
 - it allows to use a variable (sine) acceleration/deceleration value for each interpolated steps, reducing start and stop shakes.

The less is the interpolation interval, the better is the trajectory control and the better is the acceleration/deceleration effect.
But a small interpolation interval value uses more bandwidth, you can increase the interpolation interval a bit if you experiment some bandwidth issue.
Note the "Max" and "Min" labels refers to interpolation frequency (frequency = 1/interval).

Output/input.

This is the input scan and output update interval, you should increase the interval if you experiment bandwidth issues.
It has effect only if input scan and/or output update are enabled.

Acceleration.

This new feature allows to use a different speed ratio for each interpolated steps following a sine curve acceleration/deceleration function.
0% means the whole step (every semi key steps) will be performed using a constant speed, it could cause shakes during the start and stop of the step.
Using a non zero value will allow acceleration/deceleration feature.

Acceleration/deceleration effect can't work properly if interpolation interval is too high (not enough interpolated steps to apply different speed ratio).
Using acceleration/deceleration feature preserve the global time per step, it starts and ends slowly but it goes faster during the middle travel.

Global speed.

As described in the "Acceleration" feature warnings, the global time per step is preserved, for example : if a step takes 2 seconds to accomplish,
no matter is the acceleration effect value it will always takes 2 seconds to accomplish.
But as the speed is reduced or increased during a step using the Acceleration feature, you should need to change the global speed :
 - if the middle travel is too fast now, you should reduce the speed.
 - if you think it could go faster now there's no more shakes, you should increase the speed.

Global speed has effect on all project and all steps.
It doesn't change anything to the speed stored in the database, it adjust the speed "during" a play.

Thread.

It let you adjust the separate "Play" thread priority, increasing this value could help to reduce shakes during a play on slow (or very busy) computers.
This won't solve "slow" COM port issues !

Interpolate "Joint" moves.

You could ask why joint moves are interpolated as they don't need trajectory control ?
Because they need interpolation for acceleration/deceleration effect to apply.
So if you're not using acc/dec, you can uncheck this box to gain a bit more bandwidth, all joint moves will be performed as SEQ does.

All recommended values are shown as a blue range on sliders.

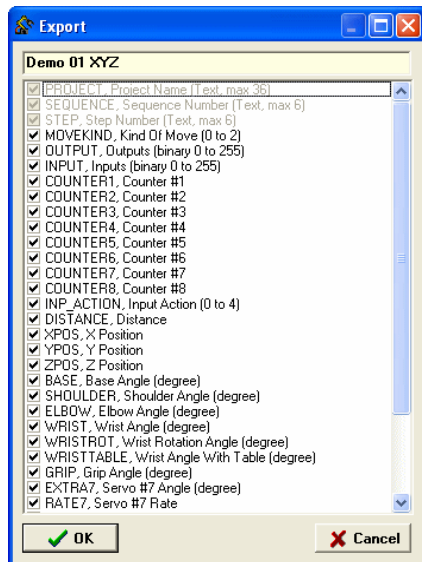# Lynxmotion RIOS SSC-32 V1.06 Manual

7 - The 'Project' module

Here you can export, import, rename or delete a Project and add comments.
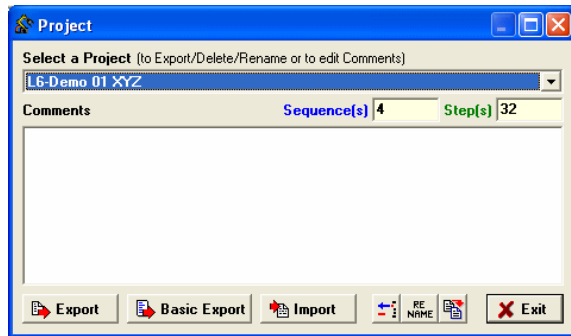
 For 'Basic Export', see chapter 7.1
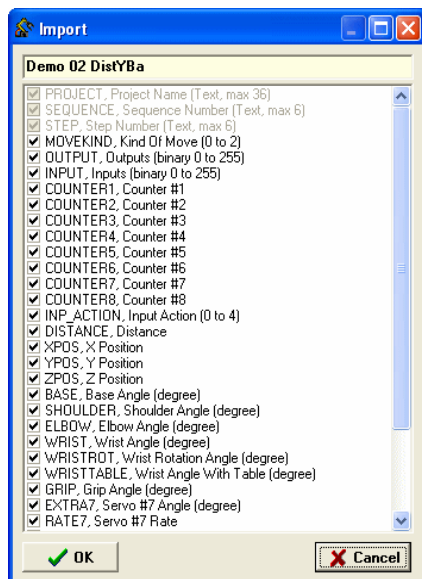
7.1 Export.

Select the Project to export
Select the fields (all selected by default)

Click on the 'Export' button, select a destination directory / file
name and click save.

File format is csv, it's a text file with a semicolon separator for fields.
Notepad, Excel...etc, are able to open and edit this.

Use Export to make a backup of your work or to share projects with friends.

7.2 Import.

Import a previously exported file
Select the csv file to import
Select the fields (all by default)

Click on the 'Import' button, select a project name and click 'Ok'.

Import your home-made csv file
A csv file must contain only ONE project.
The first line is the fields' names separated with semicolons.
All following lines must contain at least:
- the project name
- a Sequence number from 1 to 999999 with no missing number (1 and 3 without 2 is not valid)
- a Step number from 1 to 999999 with no missing number (1 and 3 without 2 is not valid)
and the other fields to import.

If your field names differ from the internal field names, you'll be asked to reassign them.

Importing your home-made csv is a risk to crash the internal database!
To prevent this, close the software, go to installation directory and backup the two files 'Moves.dbf' and 'Moves.mdx' in a secure place. If you crash the database, you'll just have to close the software and restore these two files.

Lynxmotion RIOS SSC-32 V1.06 Manual

8 - New RIOS SSC-32 features

8.1 Export Basic code for Basic Atom or Basic Stamp 2

In the 'Project' module, select 'Export' then 'Export Basic' button, the 'Basic Export parameters' will appear.
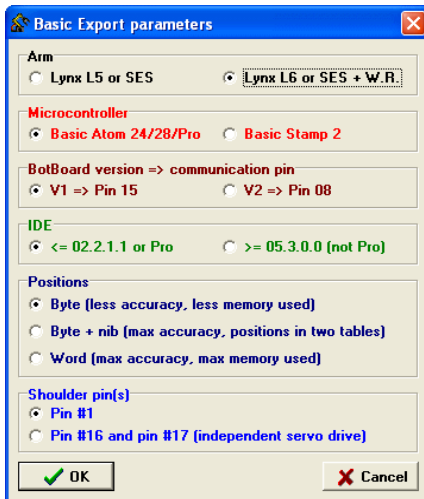
Select the kind of arm (selecting L5 won't export the 'wrist rotate' positions).

Select the kind of microcontroller.

Select your BotBoard Version => communication pin

Select the positions data size :
   - byte, byte + nib or word for a Basic Atom.
   - byte or word for a Basic Stamp 2.

Select the shoulder pin(s). (see chapter 7.3 : Independent shoulder servo drive)

Click 'OK' and select a destination file/folder.

Open the exported file with the 'Basic Micro Atom IDE' (for Basic Atom *.bas files),
*** use the Pro IDE for Pro chips ***
or the 'Basic Stamp Editor v2.2' (for Basic Stamp 2 *.bs2 files)
(don't forget to connect the microcontroller to the PC)
and download the program to the microcontroller.
Then, connect the microcontroller to the SSC-32 (don't forget to remove some jumpers).

The Basic exported file contains position data, durations, pause data,
and the complete program to read data and perform movements.

Basic Stamp 2 program require the version 2.2 of the BASIC Stamp Editor (with Pbasic 2.5 include).

Outputs and inputs settings are not exported.

New Basic Atom IDE V05.3.0.0 :
if you are using this new IDE, select the '>=05.3.0.0' choice, it export the correct 'serout' syntax (new constant values format)
and a config line requested by this new IDE.

The values in this array are NOT GUARANTEED to be exact, it depends on compilers/IDE version and microcontrollers version!

(for example: 'RIOS DEMO 01 XYZ' is only 32 steps)

Maximum export steps				
Chip	**Method**	**L5**	**L6**	
BA	Byte	752	674	BA: Basic Atom 28 - IDE : Basic Micro ATOM IDE 02.2.1.1
	Byte + Nib	552	507	
	Word	485	423	
BA Pro	Byte	3150	2830	BA Pro: Basic Atom Pro 28 - IDE : Basic Micro ATOM Pro IDE 07.2.0.6
	Byte + Nib	2350	2160	
	Word	2030	1770	
BS2	Byte	206	182	BS2: Basic Stamp 2 - IDE : Basic Stamp Editor v2.2
	Word	129	110	

8.2 Analog inputs

To enable analog inputs, go to the SSC-32 config module.
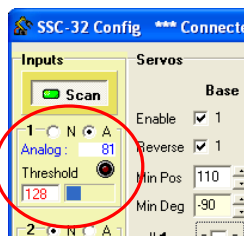
Select 'A' for analog (N = Numeric)
If a sensor is connected, the blue bar and the 'Analog' box will show the value.

The input #1 is the SSC-32 'A' input.
(2 = B, 3 = C, 4 = D)

Enter a threshold value.

When the analog value > threshold value, the Input is 'On'

So, if you use a Sharp GP2D12 IR Sensor for example, the threshold value lets you adjust the 'distance'.

Lynxmotion RIOS SSC-32 V1.06 Manual
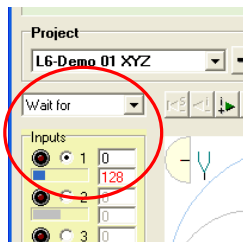
8 - New RIOS SSC-32 features
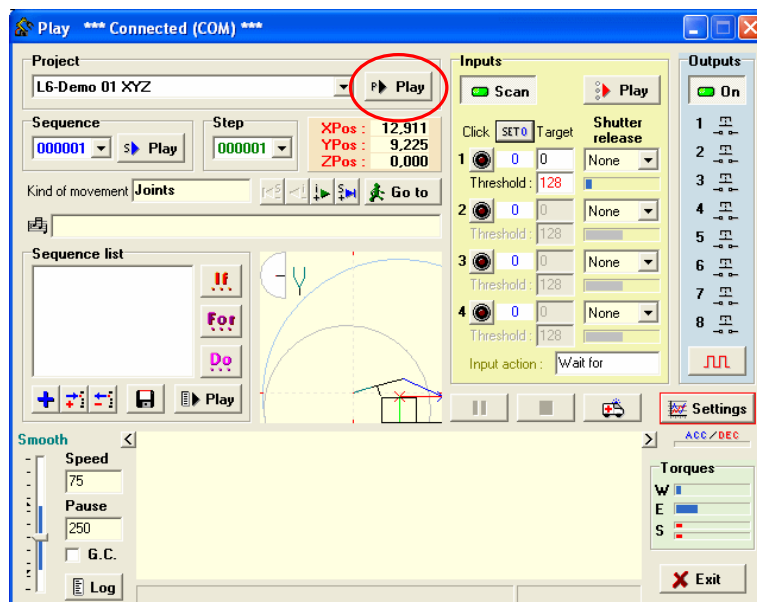
8.2 Analog inputs

Try this :

Go to the 'Moves' module and select 'Demo 01 XYZ'.

Then, select sequence 000001 and step 000001, and select inputs #1.
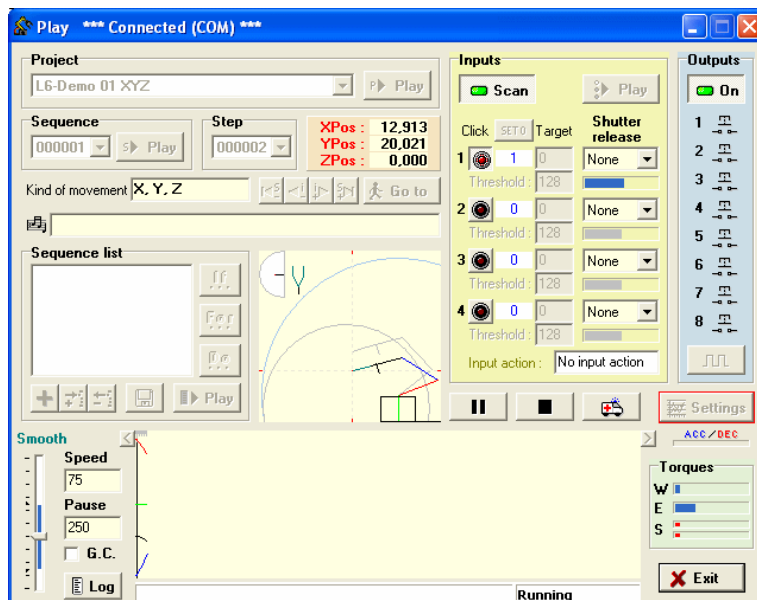It will select 'Wait for' as default 'input action', change the threshold value if needed.

Then, go to the 'Play' module and click on the Project Play button.

The Program will wait until the analog input #1 > threshold value.
If you use a Sharp GP2D12 IR Sensor for example, just approach the sensor with your hand and the program will start!

Lynxmotion RIOS SSC-32 V1.06 Manual

8 - New RIOS SSC-32 features

8.3 Independent shoulder servo drive

Now you can choose to control the shoulder servos independently
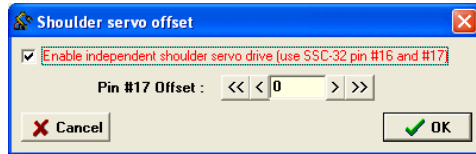
Open the 'SSC-32' window and click on the 'Shoulder' button

When you enable the independent shoulder servo drive, the pin #16 is used to control one shoulder servo, and the pin #17 the other one.
(so you'll have to remove the 'Y' cable and plug each servo on each pin)
Then, you can adjust the pin17 offset to make both servo perfectly lined.
(offset is -100uS to 100uS by 1uS steps)

This is useful when mechanical adjust can't be done.

Note : when independent shoulder servo drive is on, the pin #1 is still driven.
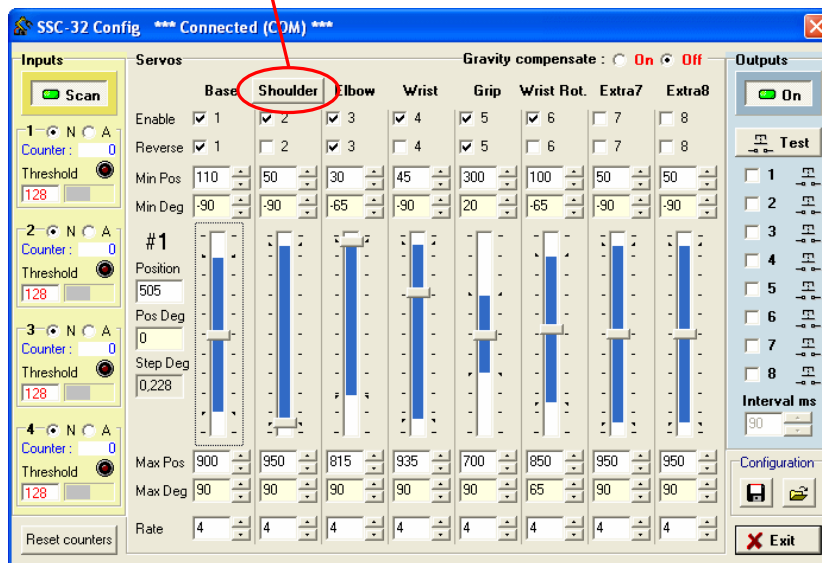
**When you enable the independent shoulder servo drive, the pin #18 is now used too, it's driven as the opposite as the pin #17,
sharing the same offset value and turning opposite side as pin #17, this allows to use a shoulder system with head to head positioned servos.**

*** DON'T USE THE PIN #16 and PIN #18 PAIR WITH LYNXMOTION'S ARMS ***

About the Basic code export function :

Now you can choose to export code to drive either the pin #1 or the pins #16 and #17
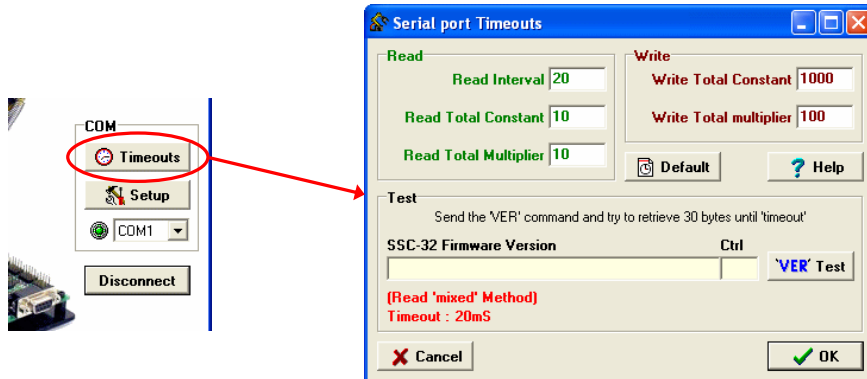if you select 'pin #16 and pin #17', it will add a line with the offset command too.
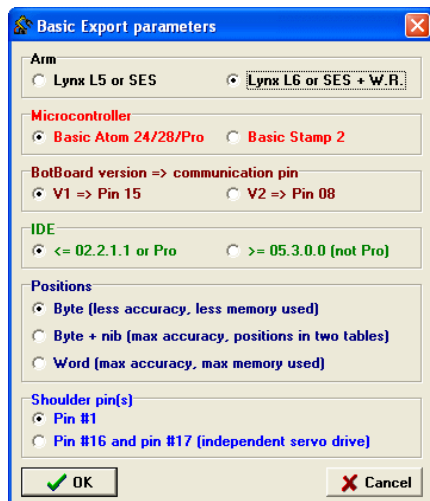
Lynxmotion RIOS SSC-32 V1.06 Manual

9 - The 'Timeouts' module

Here, you can change some advanced COM port communication values.
** Don't change anything if there's no communication problem ! **

If you're using a 'Wiport' and have experienced some communication problems,
try to set the 'Read interval' = 30 (no change on other values)
Some USB to serial cable needs a 'Read interval' values >= 30 too.

The 'Write' values are not critical, so no need to change them.

Before clicking 'Test', check the 'Timeout : xxmS' value. It's the time you will wait in the worst case.
If this value is too big, it could freeze the window for a long time, be careful !

More information with the 'Help' button :

You normally have to specify the number of bytes when reading bytes on the Com port. However, when you don't know
how many bytes to read, the timeout value will let you know when the Com port has finished sending data.

Trying to read 4 bytes

4 bytes read

1. Total method — Multiplier (1) | Multiplier (2) | Multiplier (3) | Multiplier (4) | Constant | Timeout

Byte 1   Byte 2   Byte 3   Byte 4

2. Interval method — Interval | Interval | Interval | Interval

OK   OK   OK   **Timeout! 3 bytes read**

1. Read Total method :
Activated if the "Read Interval" value = 0
- The "Read Total Multiplier" is the value multiplied by the number of bytes requested. The "Read Total Constant" value
  is the maximum time to wait for all the requested bytes before generating a timeout event.
Timeout value depends on the number of bytes requested

2. Read Interval method :
Activated if the "Read Interval" value is > 0 and Both "Read Total Multiplier" and "Read Total Constant" = 0
- The "Read Interval" value is the maximum time to wait between bytes before generating a timeout event.
Timeout value is the "Read Interval"

3. Read 'mixed' method :
Activated if the "Read Interval" value is > 0 and "Read Total Multiplier" and/or "Read Total Constant" > 0
- The first Timeout value elapsed will end the function.
Timeout value depends on the number of bytes requested and the "Read Interval"

Read Interval method ends faster when you don't know how many bytes to ask for.
For example, when asking for the SSC-32's version string, you don't know how many bytes the SSC-32 card will return,
but you know you don't want more than 30 bytes to display.
RIOS asks for 30 bytes, and the timeout occurs after the last byte is sent (it could be 14, 15, etc…)
So 10mS (default) after the last byte is sent by the SSC-32 card, the Read buffer is cleared to remove unrequested bytes.

If there are some missing bytes, try increasing the "Read Interval" value.
**If it doesn't help, try increasing the "Read Total Multiplier" then "Read Total Constant" slowly.**

# Lynxmotion RIOS SSC-32 V1.06 Manual

10 - 3D Scanner

The 3D scanner module needs a Sharp GP2D12 IR sensor, mounted on the Gripper servo and connected to the SSC-32 Analog Input #A.

It browses an area with the arm, storing GP2D12 sensor data into a 256 levels grayscale bitmap, from black (distance >= 80cm ~ 31.5 inches) to white (distance <= 10cm ~ 4 inches). The result is a 3D image.
(For now it needs an external free 'leveller' software to see the grayscale bitmap in 3D.)
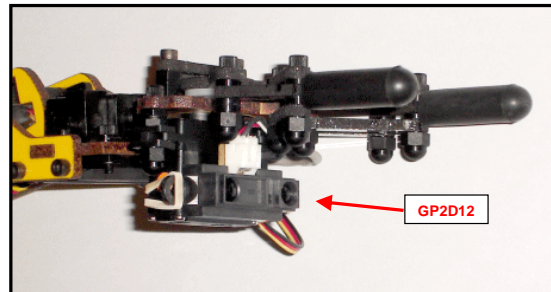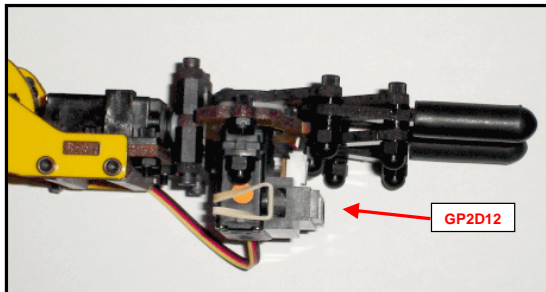
Just use a small rubber band to attach the sensor.
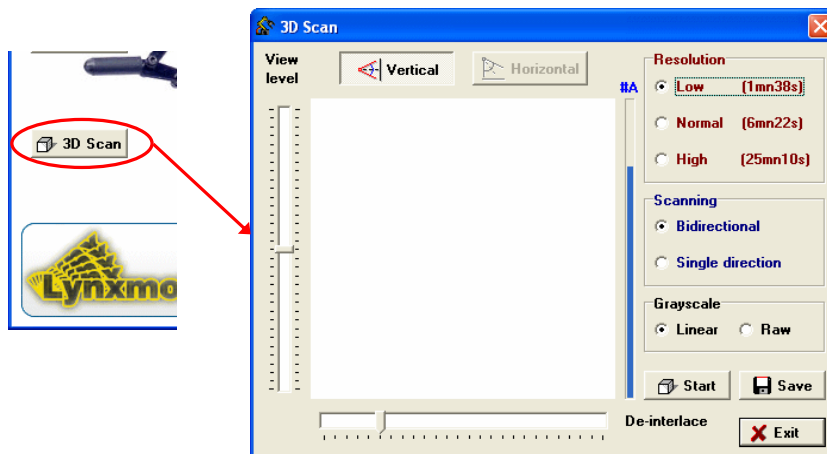
Choose Vertical or Horizontal scan :
- Vertical scan is perfect to scan a face, a bust or any vertical object.
- Horizontal scan is better for small objects, laid on a table.

Select a resolution :
- Low is 64X64, nice for a preview
- Normal is 128X128
- High is 256X256, but slow

Select scanning direction :
- Bidirectional is fast but not the best quality.
- Single direction is a bit slower but pretty good result.

Press 'Start' to begin the Scan

Time with single direction scanning

Grayscale :
- Linear => datas from the GP2D12 are corrected to be proportionally spaced, according to the distance (correcting the non-linearity of the GP2D12).
- Raw => datas are only scaled to suit the 0-255 grayscale range.
You don't have to choose the grayscale option before starting the scan.
You can change this after the scan is complete to decide the one you will save,
or you can select 'linear', save it as bmp image, then select 'raw' and save it too.
You can change this option during the scanning process too, just to check.

De-interlace
This slider allows fine adjustment to the odd and even lines matching.
It's very useful when using the bidirectional scanning option.
For single direction scanning, set the slider in the middle.
You can change this setting during the scanning process or after it's complete.

View level (use this when the scanning process is complete)
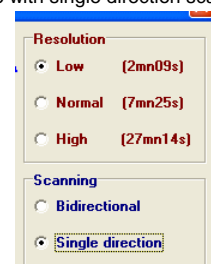You can use this in two different ways :
- Move the slider slowly to highlight in red the corresponding level (distance).
  It's useful to see all the pixels on the 'same level'.
- Click on the bitmap (left click) and keep pressed, then move the mouse cursor on the bitmap,
  all the 'same level' pixels as the one your hovering are highlighted in red, and the view level slider reflects the current level.

Save
Save the current scan to a 256X256 grayscale bitmap (*.bmp) file.
It doesn't save any red pixel from the 'view level' feature.

Lynxmotion RIOS SSC-32 V1.06 Manual

10 - 3D Scanner

Some examples :

Vertical scanning
Low resolution, linear, single direction
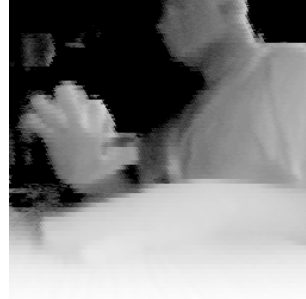

Vertical scanning
Normal resolution, linear, single direction

Vertical scanning
High resolution, linear, single direction

Horizontal scanning
High resolution, linear, single direction

Horizontal scanning
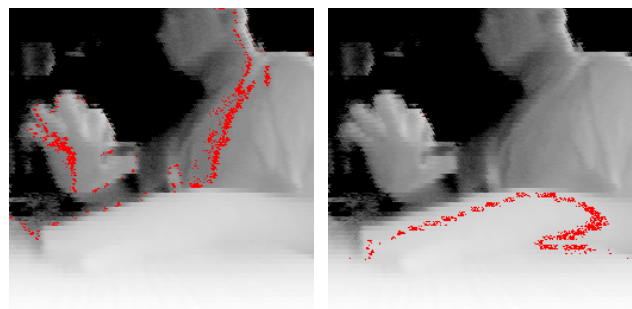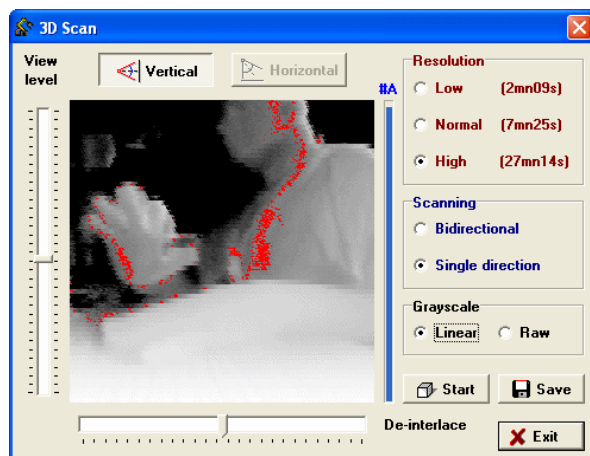**High resolution, raw, single direction**

Vertical scanning
High resolution, raw, single direction

Click on the bitmap or move the 'View level' slider to show all the pixel at the same level.

Lynxmotion RIOS SSC-32 V1.06 Manual

11 - Sockets.

11.1 Socket Server.

**The socket server allows an external application to control some feature in RIOS,
the LAN IP is the address that the external application have to connect, the Port # is the one to use to communicate,
(IP address + Port # is called a socket), click "Activate" to enable the socket server, the log file will help you to see the communications.**
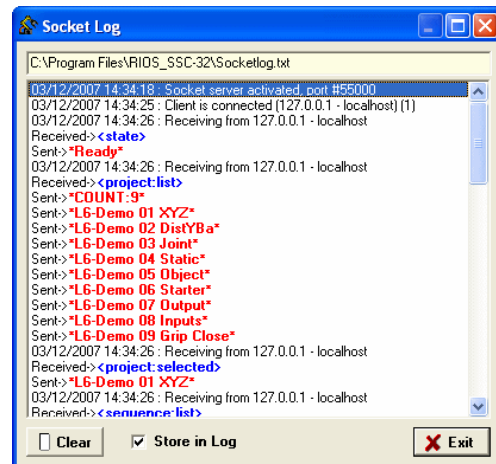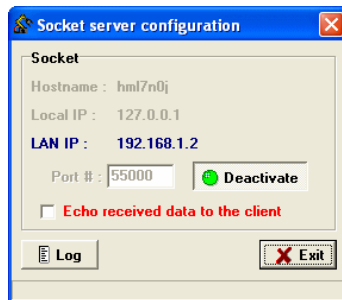
Commands list :

All returned strings from RIOS starts and ends with a "*"
except echo strings (if enabled).

- "**<ssc-32:ver>**" will return the SSC-32 firmware version.
  ("***SSC32-1.06XE***" for example).

- "**<project:list>**", "**<sequence:list>**" and "**<step:list>**" will return the according list formated as follow :
  - a "***COUNT:X***" string where "X" is the number of item(s) in the list.
  - then **X** string(s) "***ITEM***", see the example in the socket log screenshot above.

- "**<project:selected>**", "**<sequence:selected>**" and "**<step:selected>**" will return the according item selected in RIOS.
  ("***L6-Demo 01 XYZ***" for example).

- "**<project_select:projectname>**", "**<sequence_select:sequencename>**" and "**<step_select:stepname>**"
  selects the according item in RIOS, "**<project_select:L6-Demo 02 DistYBa>**" for example.

- "**<project:play>**", "**<sequence:play>**" and "**<step:goto>**"
  clicks on the according "play" button in RIOS.

- "**<sequence_list:play>**" clicks on the Sequence List player "play" button.
  (do nothing if no list exists).

- "**<sequence_input_sr:start>**", "**<sequence_input_sr:stop>**" clicks on the inputs "play" button.
  (do nothing if no shutter release defined).

- "**<input_scan:on>**", "**<input_scan:off>**" sets on or off the scan of inputs.

- "**<output_update:on>**", "**<output_update:off>**" allows or not the outputs update.

- "**<state>**" will return "***Busy***" if a sequence is currently playing or "***Ready***" if none is playing,
  "***Paused***" if paused or "***Error***" if RIOS is not in the "Play" form.

- use "**<stop>**" to stop playing, "**<pause>**" to pause and "**<resume>**" to resume
  and "**<emergency:stop>**" to stop playing and disable torques on all servos.

- "**<input_simu:#input>**". Simulates an input where **#input** is the input number from 1 to 4.
  example : (**<input_simu:1>**) clicks on the input #1 button to simulate an input).

- "**<input_value:#input>**" will return the input value where **#input** is the input number from 1 to 4.
  returns "**0**" or "**1**" if the input is numeric, "**0**" to "**255**" if the input is analog.

Lynxmotion RIOS SSC-32 V1.06 Manual

11 - Sockets.

11.2 Socket Client.

The socket client allows connecting to a robot using TCP/IP instead of COM port, but you need some hardware on the robot side
to receive TCP/IP flow (via WIFI for example) and convert it to the SSC-32 COM port.

You can already use two WIFI COM ports to control a robot, one on the PC side, one on the robot side.
But the Socket system allows the PC to communicate directly through the network, so if you're using an external WIFI modem (www box),
you can use it to communicate with the robot, saving one of the two WIFI COM port adapter,
now you only need the one on the robot side, which must be compatible with your WIFI modem.
**Note : if you have an internal WIFI modem that you can configure to act as a WIFI COM port on the PC side,
don't bother using the socket client method, connect directly to it using its emulated COM port #.**

Enter the "robot side WIFI COM port" IP address   Click "Activate"
Enter the "robot side WIFI COM port" port # (socket)   If the "robot side WIFI COM port" is found it will stay activated.
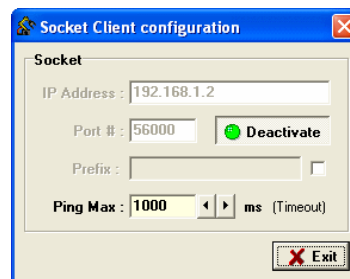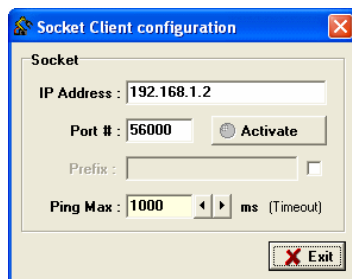
**Ping Max is used as a
Timeout here,
adjust from 100 to 2000.**

**You can only connect to COM port OR Socket client, not both at the same time, "activating" the Socket client doesn't mean "connect".
To finish connecting the software to the robot, exit this window and click "connect" on the Main form.
If the SSC-32 card is found through the network, it will connect and show the firmware version.**

**If a SSC-32 card is found on a COM port, it will auto disable the Socket client and it will connect to COM port.
If the "robot side WIFI COM port" is not found through the network it will auto disable the Socket client.**

"Prefix" allows to add a string before each "normal" data flow to the SSC-32, don't use prefix if you don't know or don't need to.

Lynxmotion RIOS SSC-32 V1.06 Manual

12 - Socket Client Demo.

This little C++ Demo program was made with Borland C++ Builder 6.0.
Source code is given "as is", it shows how to communicate with RIOS V1.06 or higher.

The Socket Client Demo program can connect to RIOS Socket Server
 in order to command the "Play" form through a network.

How it woks :
- At first, run RIOS and open the Socket Server form.
- Note the Local IP if you plan to test the Demo program on the same PC or note the LAN IP if you want to test through a network,
then note the port #.
- Activate the Socket Server, your Firewall may warn you about RIOS trying to communicate, you'll have to allow this.
- Close the Socket Server form and go to the "Play" form.
- Run the Socket Client Demo program, enter the right IP previously noted and so on the Port #.
- Activate the Socket Client.

On the example below, I'm using the local IP (127.0.0.1 for all PC), so it's a test with both programs running on the same PC.

Test 1 :
- Click on the 'Ver' button, it asks RIOS the SSC-32 firmware version.
If all is ok, it will display the SSC-32 firmware version in the white status panel.
("Virtual" appears if you are connected to a virtual robot.)

Test 2 :
- Click on the 'State' button, it asks RIOS the Play state.
If all is ok, it will display "Ready" in the white status panel.

Test 3 :
- Click on "Refresh", it will ask RIOS the project list, the current project selected and its sequences and steps selected.
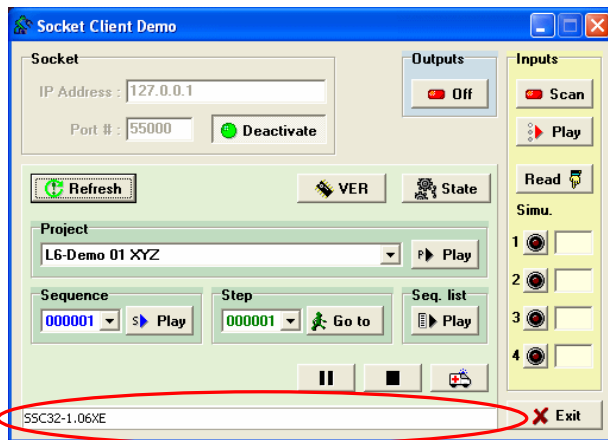
Now you can try to select another project in the Socket client demo program, it will select it too in RIOS automatically.
But if you select another one in RIOS you'll have to click "Refresh" in the Socket client demo program to update.

Now you can try to control a project, click "Play", then "Pause" to pause then click again to resume etc…

Source code files description.

SocketClientDemo.exe **is the compiled program, it can't run if RIOS is not installed (it uses Borland dll).**
SocketClientDemo.bpr **is the Borland C++ Builder project file.**
SocketClientDemo.res **is the resource file (application icon is in here).**
SocketClientDemo.cpp **is the application entry.**
MainForm.dfm **is the main form data (Buttons, ComboBox, CheckBox,Socket Client etc…).**
MainForm.h **is the main form header.**
MainForm.cpp **is the main form code.**

Lynxmotion RIOS SSC-32 V1.06 Manual

13 - Command line.

Using command line, you can start and make RIOS :
- going directly to the 'Moves/Motions' form with the right project selected,
- going directly to 'Play' form, bypassing initialization warning message and play a selected project automatically.

First, you need to create a new RIOS shortcut :

1. Right-click an open area on the desktop, point to New, and then click Shortcut.
2. Click Browse.
3. Locate the **C:\Program Files\RIOS_SSC-32\RIOS_SSC32.exe** file and click on it to select,
click Open or Ok, and then click Next.
4. Type a name for the shortcut and then click Finish.

Now, you need to edit this shortcut to add some command line parameters :

1. Right-click on your new RIOS shortcut, select the 'shortcut' tab.
2. Click on the 'Target' Box and go to the end of the line **"C:\Program Files\RIOS_SSC-32\RIOS_SSC32.exe"**
3. Here you can add 'moves' or 'play' to select the according form to open automatically in RIOS.
4. Then you can add "Your Project name", it will select this project automatically.
5. If you have chosen 'play' form, you can add a third parameter to automatically play the selected project :
 - 'play' to play the complete project
 - 'playseqlist' to launch the sequence list player
 - or 'playscan' to activate the input scan player (the one which play sequences according to inputs, see '5.1 Play')
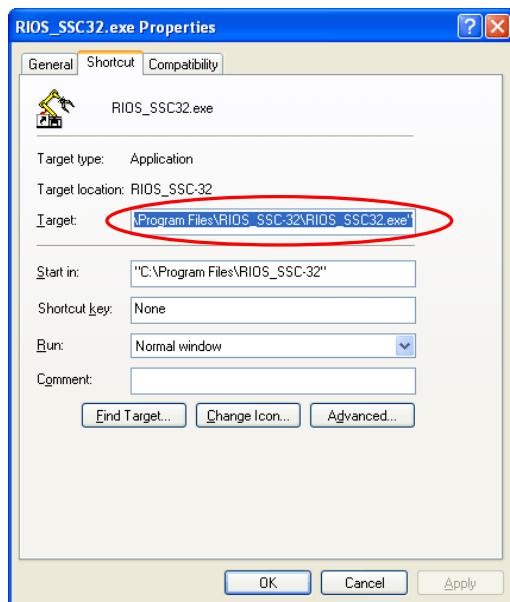
For the (third parameter) 'play' command to work properly, the robot must be 'on' before running RIOS with the shortcut!

Warning, when using the (third parameter) 'play' command, there's no more message asking to set the robot in default position before activating all servos… It will activate them automatically!

Examples :

To Open the 'Moves/Motions' form,
 "C:\Program Files\RIOS_SSC-32\RIOS_SSC32.exe" moves

To Open the 'Moves/Motions' form and select the "Demo 03 Joint" project,
 "C:\Program Files\RIOS_SSC-32\RIOS_SSC32.exe" moves "Demo 03 Joint"

To Open the 'Play' form,
 "C:\Program Files\RIOS_SSC-32\RIOS_SSC32.exe" play

To Open the 'Play' form and select the "Demo 03 Joint" project,
 "C:\Program Files\RIOS_SSC-32\RIOS_SSC32.exe" play "Demo 03 Joint"

To Open the 'Play' form and select the "Demo 03 Joint" project and to play it automatically,
 "C:\Program Files\RIOS_SSC-32\RIOS_SSC32.exe" play "Demo 03 Joint" play

To Open the 'Play' form and select the "Demo 03 Joint" project and launch the sequence list player automatically,
(you must have saved a sequence list for this project, or it won't work)
 "C:\Program Files\RIOS_SSC-32\RIOS_SSC32.exe" play "Demo 03 Joint" playseqlist

To Open the 'Play' form and select the "Demo 03 Joint" project and activate the input scan player automatically,
(you must have defined some sequences to play with the input scanner for this project, or it won't work)
 "C:\Program Files\RIOS_SSC-32\RIOS_SSC32.exe" play "Demo 03 Joint" playscan

Lynxmotion RIOS SSC-32 V1.06 Manual

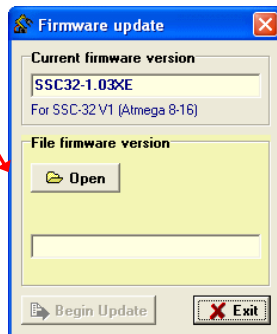## 14 - SSC-32 Firmware update.

**Don't try to update the SSC-32 firmware if all is working properly and if you don't need to.
RIOS works fine with the "SSC-32-1.06XE" (SSC-32 V1) or "SSC-32-2.03XE" (SSC-32 V2) firmware.
Be warned that some "specific" firmware may not work with RIOS.**

Check "SSC-32 Servo Controller" page at www.lynxmotion.com for Firmware update
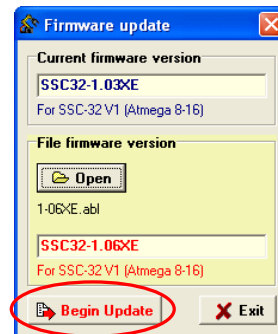
First connect to the SSC-32, then click on the "Firmware" button.

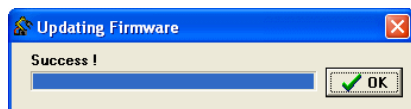**Click on "Open" and browse for the new firmware file (*.abl).**

Check the firmware file version and click "Begin Update".

Don't turn off the SSC-32 card during the firmware update process!

Click "OK" then "Exit".

**If you get errors during the firmware update process,
you must try to update the firmware again or the SSC-32 card won't work correctly.**

**Don't close the "Firmware Update" form and don't disconnect (COM Port) from the SSC-32, or you may not be able
to reconnect RIOS to the SSC-32 as the RIOS program checks to see if the SSC-32 is ok
before allowing connections to it.**

**If you can't connect to the SSC-32 because RIOS is no longer allowing it, go to www.lynxmotion.com
and download the free "SSC-32 Lynx terminal" program to update your firmware. (It will connect no matter what).**
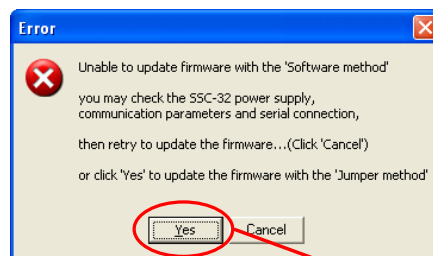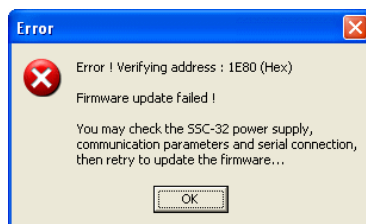
**If the card is no longer able to update its firmware using the "Software method",
then you can use the "Firmware method", which will force the card to accept
the firmware update. Follow the 3 steps described.**
Don't forget Step 2 : Power cycle the board (off then on) or it won't work.