

Instructions of Use

Technical specifications

- Platform: Android
- Operating system: At least Android 3.x Honeycomb, Android 4.0.x or higher recommended
- Processor frequency: 1 GHz
- RAM memory: 512MB
- Minimum hard disk capacity: 1 GB
- Minimum display resolution: 10"

ABC Communicator Interface

It is shown the main page of the ABC application for Android in the Figure 1. In this page, it is possible to view the Network and IP Address and it also allows importing news libraries created with the ABC library manager and open these library files imported.

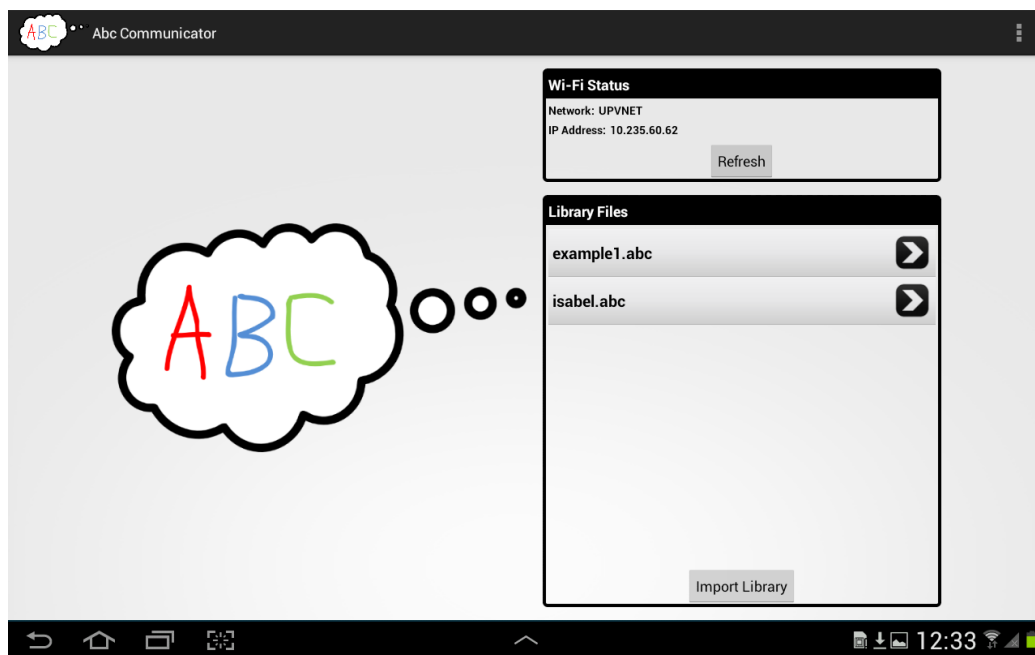


Figure 1: ABC Communicator

In Figure 2 is shown an example of a library page created.

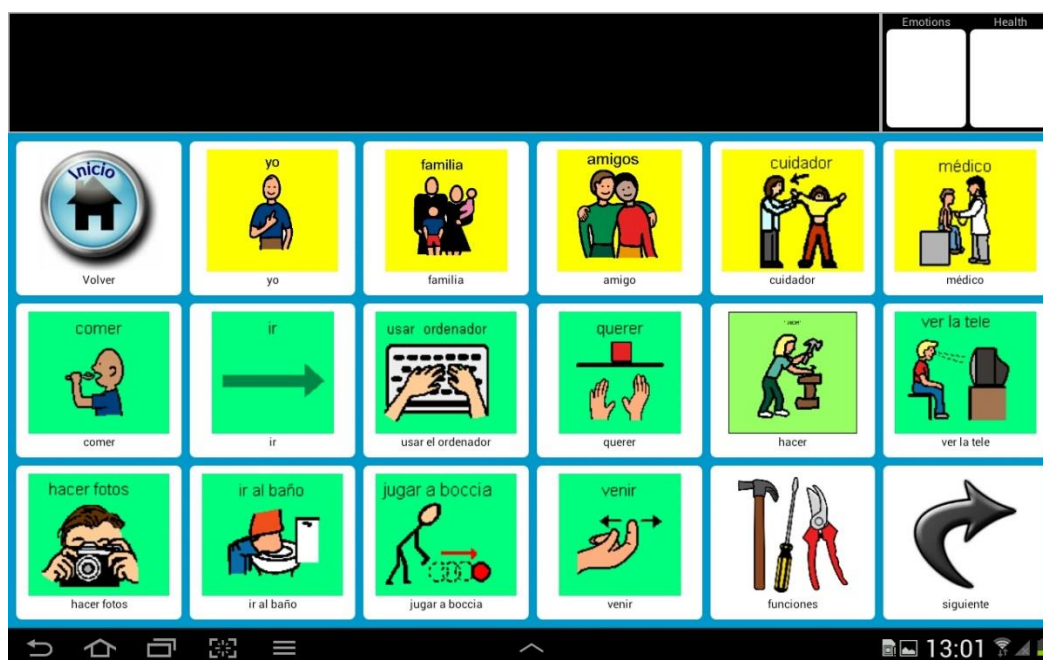


Figure 2: Library opened with ABC Communicator

For each library created, it is possible to change several properties and settings: In the Figure 3 is shown the available settings for each library.

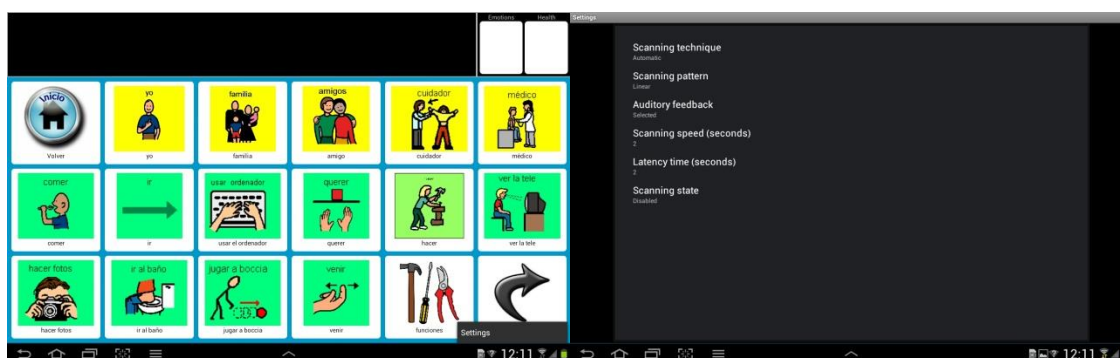


Figure 3: Settings

There are 4 scanning techniques (Figure 4):

- Automatic; when scan on is enabled, the communicator starts scanning and when a click is done, the pictogram is selected and scanning continues.
- Step; the scanning is produced when a click is done in each pictogram. When the click is not produced the pictogram is selected by a dwelling time (Figure 8)
- Inverse; when scan on is enabled, the communicator starts scanning and when a click is done, the pictogram is selected and the scanning stops.
- Directed; the IMU sends a message to indicate a change in the sensor position in two axis and when the click is done, the pictogram in this position is selected.

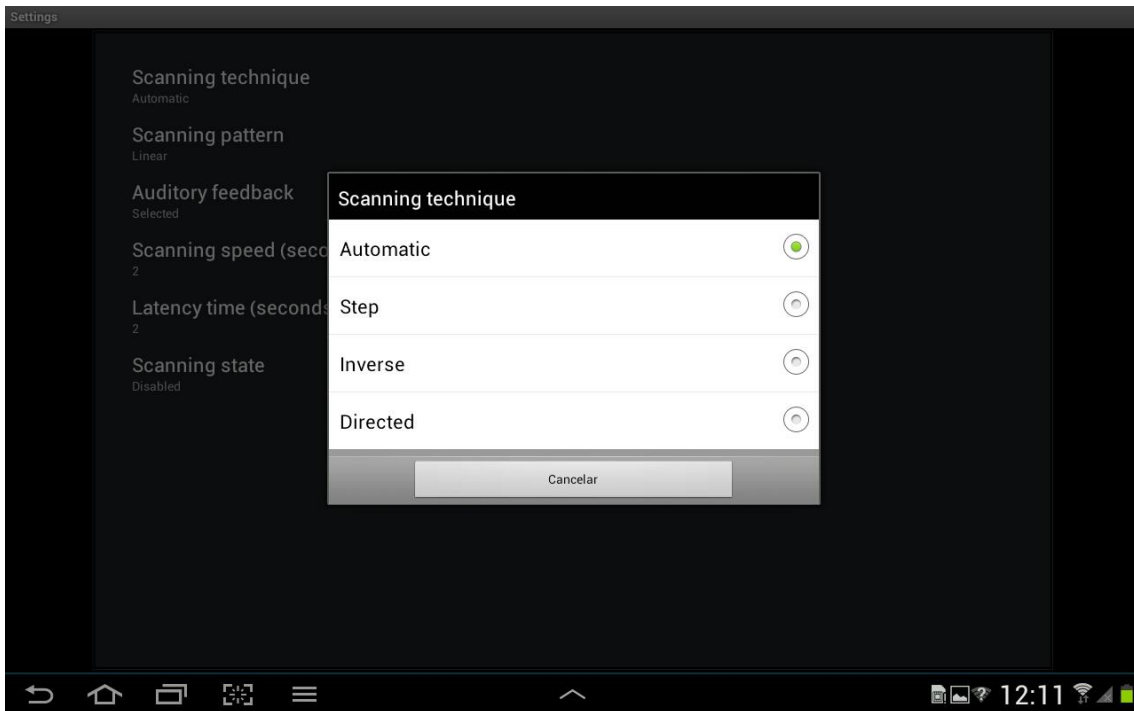


Figure 4: Scanning technique

In Figure 5 is shown the two scanning patterns; linear and row/columns.

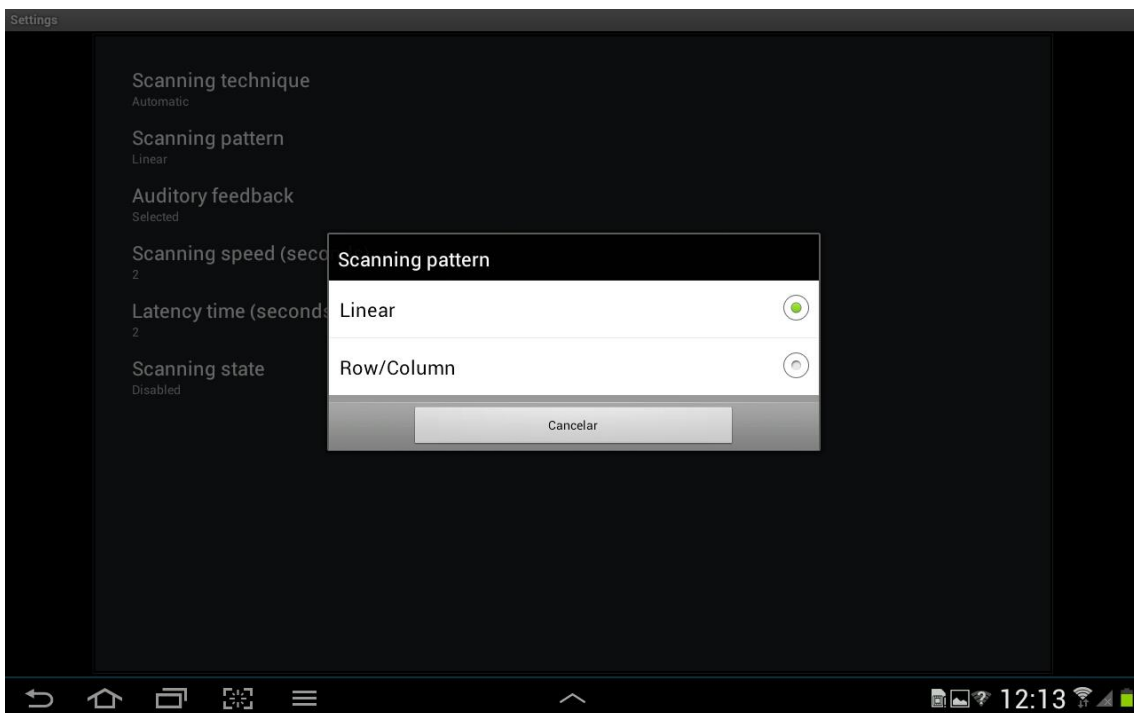


Figure 5: Scanning pattern

It is shown the two possibilities of the auditory feedback (Figure 6); selected: the auditory feedback is produced when the click is done. Scanning: the auditory feedback is produced when every pictogram is scanned.

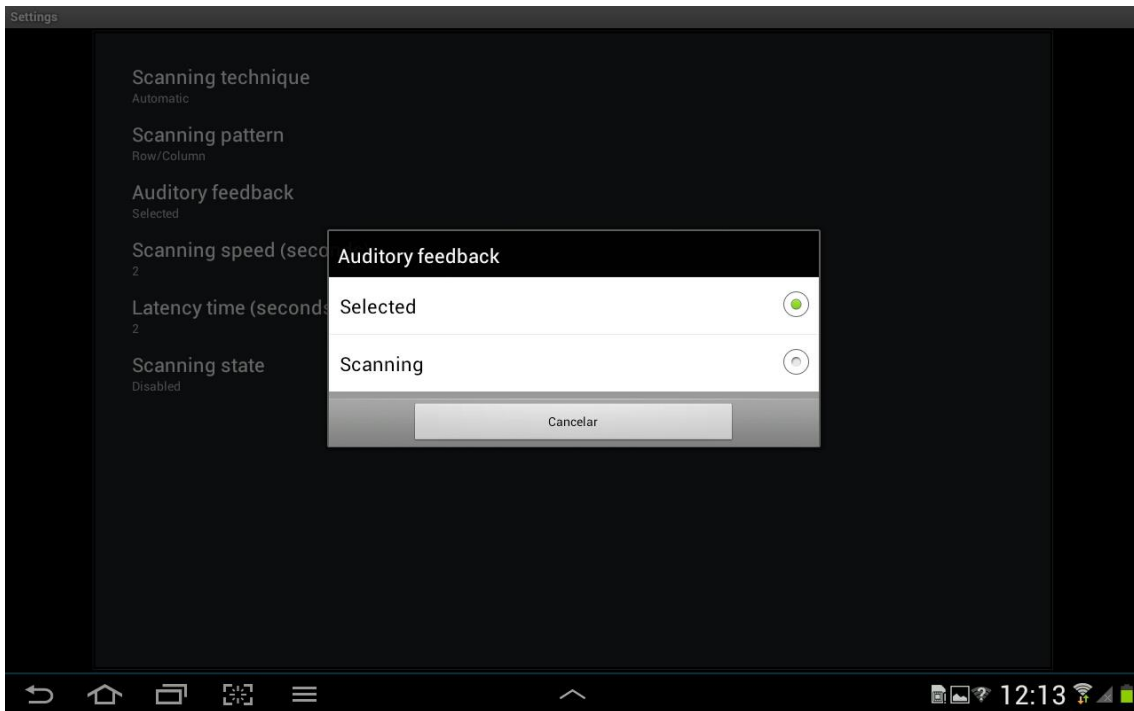


Figure 6: Auditory feedback

In addition, the scanning speed could be modified between 1 and 10 seconds for the automatic and inverse scanning technique.

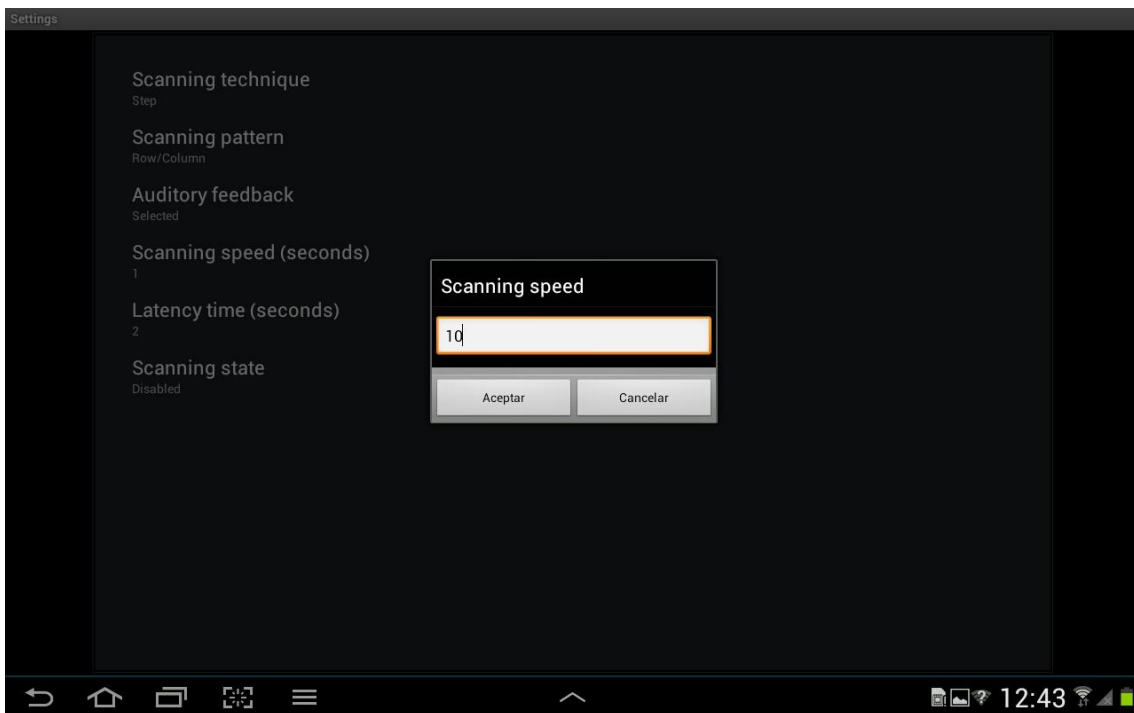


Figure 7: Scanning speed

The dwelling time could be also modified between 1 and 10 for the step scanning technique.

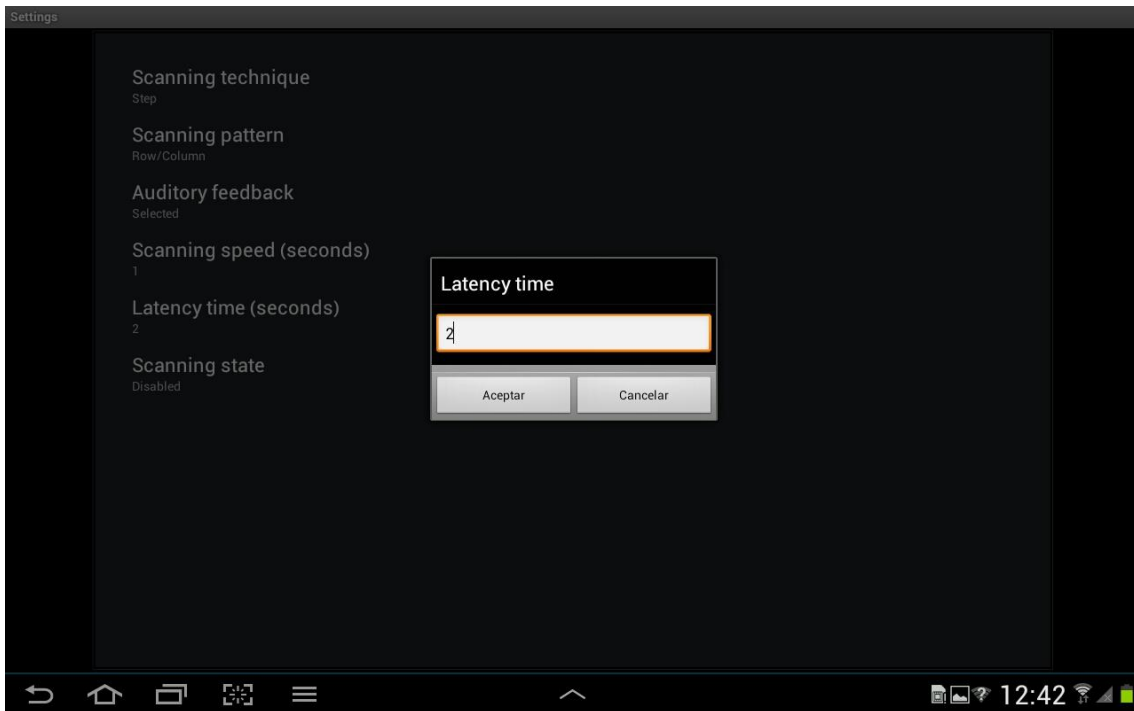


Figure 8: Dwelling time

Finally, the scanning state could be enabled or disabled.

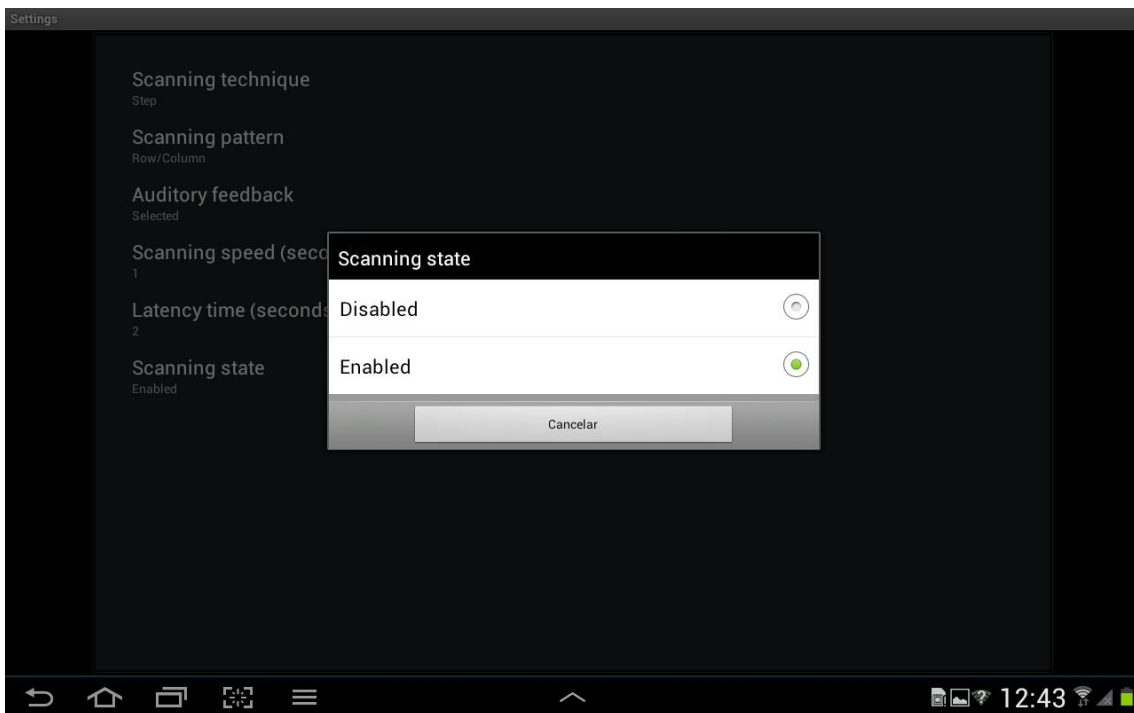


Figure 9: Scanning state

1.1. UDP configuration

The UDP communication protocol is based on a client-server model, where the ABC Communicator for android acts like a server, and the devices are the clients that send requests in order to control the communicator. UDP server is listening to port 7777.

Mini API. Technical specifications

The UDP messages implemented to control the communicator have this general format:

DEVICE ID	COMMAND CODE	COMMAND PARAMS
3 bytes	3 bytes	Variable length

- Device ID. This is a 3 character field that identifies the device that sends the message to the communicator (EMG, EEG, IMU, EMO).
- Command Code. 3 character code that identifies the action requested by the device (position change, button pressed, button released)
- Command Params. This field has a variable format and length, according to the associated command code.

The following list describes the UDP commands for communicate the communicator and the Tobi framework:

Check for server

- Command code: **RDY**
- Description: Message for checking the availability of the communicator.
- Command params: No params
- Command reply: Communicator replies with a 3 character message RDY

Start scanning

- Command code: **SON**
- Description: Starts the scanning process in the communicator. This is the first command that a device must send before sending control commands.
- Command params: No params
- Command reply: Communicator replies with a 3 character message SON

Stop scanning

- Command code: **SOF**
- Description: Stop the scanning process in the communicator.
- Command params: No params
- Command reply: No reply

Click on

- Command code: **CON**
- Description: Indicates that a button has been pressed.
- Command params: No params
- Command reply: No reply

Click off

- Command code: **COF**
- Description: Message to indicate that a button has been released.
- Command params: No params
- Command reply: No reply

Click

- Command code: **CLK**
- Description: Notifies that a button has been clicked (pressed + released).
- Command params: No params
- Command reply: No reply

Position

- Command code: **POS**
- Description: Notifies a change in a sensor position
- Command params: 3 characters with the sensor position (000...999)
- Command reply: No reply

XY Position

- -Command code: **PXY**
- Description: Message to indicate a change in the sensor position in two axis
- Command params:
 - Param1: 3 characters with the X position value (000 ... 999)
 - Param2: 3 characters with the Y position value (000 ... 999)
- Command reply: No reply

Move LEFT active item

- Command code: **POL**
- Description: Updates the active item in the communicator, making active the item located on the left of the currently active item.
- Command params: No params
- Command reply: No reply

Move RIGHT active item

- Command code: **POR**
- Description: Updates the active item in the communicator, making active the item located on the right of the currently active item.
- Command params: No params
- Command reply: No reply

Move UP active item

- Command code: **POU**
- Description: Updates the active item in the communicator, making active the item located above the currently active item.
- Command params: No params
- Command reply: No reply

Move DOWN active item

- Command code: **POD**
- Description: Updates the active item in the communicator, making active the item located below the currently active item.
- Command params: No params
- Command reply: No reply

Get active items

- Command code: **AIT**
- Description: Ask the communicator for the currently active items
- Command params: No params
- Command reply: The communicator replies with a message with this format:

<i>Header</i>	<i>Item index1</i>	<i>Item index 2</i>	<i>...</i>	<i>Item index n</i>
---------------	--------------------	---------------------	------------	---------------------

- Header: 3 character field that contains AIT
- Item indexes: a list of indexes of the activated items, 2 characters by item