Steppermotor-Chess-Figure-Movement-Protocol (SChFiM-Protocol)

Authors: Ovidiu Victor Tatar, Alessio Mossudu

Version: 1.2

Description:

An implementation of the ChFiM-Protocol (see specification for further information) using stepper motors to move a magnet which moves the figures. The interface has been extended to allow communication with the concrete implementation.

The checkerboard is additionally square-shaped consisting of 8 x 8 fields with an added border (at least as wide as a single field). Internally each field (including the border) is split into four squares resulting in 9 corners per field (overlapping with other fields). These are the internal coordinates. This results in 21 coordinates for each axis. Figures to be removed will be moved to the checkerboard's border, where they do not interfere with the other figures.

Interface:

The SChFiM-Protocol inherits its specification from the ChFiM-Protocol. Functions and commands that remain unchanged will not be mentioned again below.

public functions:

Syntax	Description
void **swapWithStorage**(byte, byte, byte)	Swap a figure with another figure on the border. If the index given is greater than the total number of figures on the border, no action will be taken. *parameters:* 1. x-coordinate of the figure's current field 2. y-coordinate of the figure's current field 3. index of the figure in storage/on the border *return values: -*
void **resetMotors**()	Reset the Steppers to their initial position. *parameters: -* *return values: -*
void **setBaseSpeed**(long)	Set the base-speed to use on the steppers in rpm. *parameters: base-speed in revolutions per minute* *return values: -*
long **getBaseSpeed**()	Return the base-speed to use on the steppers in rpm. *parameters: -* *return values: base-speed in revolutions per minute*
byte **getCurrentX**()	Return the current x-coordinate of the magnet's corner-position. *parameters: -* *return values: the x-coordinate*
byte **getCurrentY**()	Return the current y-coordinate of the magnet's corner-position. *parameters: -* *return values: the y-coordinate*

Syntax	Description
`int getDistanceToLastX()`	Return the distance on the x-axis of the magnet's last corner-position. *parameters: -* *return values: the distance on the x-axis*
`int getDistanceToLastY()`	Return the distance on the y-axis of the magnet's last corner-position. *parameters: -* *return values: the distance on the y-axis*
`bool isMagnetEnabeled()`	Return whether the magnet is enabeled. *parameters: -* *return values: -*
`byte getFiguresInStorage()`	Return the amount of figures stored on the border. *parameters: -* *return values: -*

commands:

Syntax (<…>: argument)	Description
`/w <x1> <y1> <i> \n`	Swap a figure with another figure on the border. If the index given is greater than the total number of figures on the border, no action will be taken. *arguments: -* x1 (int): x-coordinate of the figure's current field y1 (int): y-coordinate of the figure's current field i (int): index of the figure in storage/on the border *sends back: -*
`/x \n`	Resets the motors. *arguments: -* *sends back: -*
`/s s <rpm> \n`	Set base-speed to use on the steppers in rpm. *arguments: rpm (long): the base-speed in rpm* *sends back: -*
`/g s \n`	Sends back base-speed to use on the steppers in rpm. *arguments: -* *sends back: s<long>\r\n*
`/g cx \n`	Sends back the current x-coordinate of the magnet's corner-position. *sends back: cx<int>\r\n*
`/g cy \n`	Sends back the current y-coordinate of the magnet's corner-position. *arguments: -* *sends back: cy<int>\r\n*
`/g dx \n`	Sends back the distance on the x-axis of the magnet's last corner-position. *arguments: -* *sends back: dx<int>\r\n*
`/g dy \n`	Sends back the distance on the y-axis of the magnet's last corner-position. *arguments: -* *sends back: dy<int>\r\n*
`/g m \n`	Sends back whether the magnet is enabeled. *arguments: -* *sends back: m<bool>\r\n*
`/g sf \n`	Sends back the amount of figures stored on the border. *arguments: -* *sends back: sf<int>\r\n*