Mouse.move()

Description

Moves the cursor on a connected computer. The motion onscreen is always relative to the cursor's current location. Before using `Mouse.move()` you must c Mouse.begin()

Syntax

```
Mouse.move(xVal, yVal, wheel)
```

Parameters

`xVal`: amount to move along the x-axis. Allowed data types: `signed char`.
`yVal`: amount to move along the y-axis. Allowed data types: `signed char`.
`wheel`: amount to move scroll wheel. Allowed data types: `signed char`.

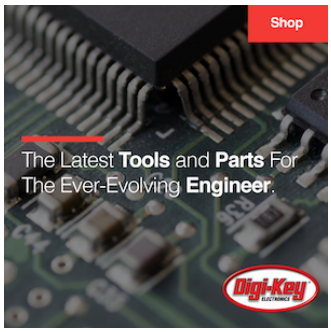
Returns

Nothing

Example Code

```
#include <Mouse.h>

const int xAxis = A1;         //analog sensor for X axis
const int yAxis = A2;         // analog sensor for Y axis

int range = 12;               // output range of X or Y movement
int responseDelay = 2;        // response delay of the mouse, in ms
int threshold = range / 4;    // resting threshold
int center = range / 2;       // resting position value
int minima[] = {1023, 1023};  // actual analogRead minima for {x, y}
int maxima[] = {0, 0};        // actual analogRead maxima for {x, y}
int axis[] = {xAxis, yAxis};  // pin numbers for {x, y}
int mouseReading[2];          // final mouse readings for {x, y}


void setup() {
  Mouse.begin();
}

void loop() {
  // read and scale the two axes:
  int xReading = readAxis(0);
  int yReading = readAxis(1);
```

LANGUAGE
FUNCTIONS
VARIABLES
STRUCTURE
LIBRARIES
IOT CLOUD API
GLOSSARY



Last Revision: 2019/02/21

Last Build: 2021/04/15

EDIT THIS PAGE

```
/*
  reads an axis (0 or 1 for x or y) and scales the
  analog input range to a range from 0 to <range>
*/

int readAxis(int axisNumber) {
  int distance = 0; // distance from center of the output range

  // read the analog input:
  int reading = analogRead(axis[axisNumber]);

  // of the current reading exceeds the max or min for this axis,
  // reset the max or min:
  if (reading < minima[axisNumber]) {
    minima[axisNumber] = reading;
  }
  if (reading > maxima[axisNumber]) {
    maxima[axisNumber] = reading;
  }

  // map the reading from the analog input range to the output range:
  reading = map(reading, minima[axisNumber], maxima[axisNumber], 0, range);

  // if the output reading is outside from the
  // rest position threshold,  use it:
  if (abs(reading - center) > threshold) {
    distance = (reading - center);
  }

  // the Y axis needs to be inverted in order to
  // map the movemment correctly:
  if (axisNumber == 1) {
    distance = -distance;
  }

  // return the distance for this axis:
  return distance;
}
```

Notes and Warnings

When you use the `Mouse.move()` command, the Arduino takes over your mouse
Make sure you have control before you use the command. A pushbutton to to
the mouse control state is effective.

See also

LANGUAGE Mouse.click()

LANGUAGE Mouse.end()

LANGUAGE Mouse.press()

LANGUAGE Mouse.release()

LANGUAGE Mouse.isPressed()