MD_MAX72xx LED Matrix Arduino Library

(https://majicdesigns.github.io/MD_MAX72XX/index.html)

MD_MAX72XX Class Reference

`#include <MD_MAX72xx.h>`

Public Types

enum	**controlRequest_t** { **SHUTDOWN** = 0, **SCANLIMIT** = 1, **INTENSITY** = 2, **TEST** = 3, **DECODE** = 4, **UPDATE** = 10, **WRAPAROUND** = 11 }
enum	**controlValue_t** { **OFF** = 0, **ON** = 1 }
enum	**transformType_t** { **TSL**, **TSR**, **TSU**, **TSD**, **TFLR**, **TFUD**, **TRC**, **TINV** }
typedef const uint8_t	**fontType_t**

Public Member Functions

	MD_MAX72XX (uint8_t dataPin, uint8_t clkPin, uint8_t csPin, uint8_t numDevices=1)
	MD_MAX72XX (uint8_t csPin, uint8_t numDevices=1)
void	**begin** (void)
	~MD_MAX72XX ()

Methods for object and hardware control.

bool	**control** (uint8_t dev, **controlRequest_t** mode, int value)
void	**control** (**controlRequest_t** mode, int value)
bool	**control** (uint8_t startDev, uint8_t endDev, **controlRequest_t** mode, int value)
uint8_t	**getDeviceCount** (void)
uint16_t	**getColumnCount** (void)
void	**setShiftDataInCallback** (uint8_t(*cb)(uint8_t dev, **transformType_t** t))
void	**setShiftDataOutCallback** (void(*cb)(uint8_t dev, **transformType_t** t, uint8_t colData))

Methods for graphics and bitmap related abstraction.

void	**clear** (void)
void	**clear** (uint8_t startDev, uint8_t endDev)
bool	**drawHLine** (uint8_t r, uint16_t c1, uint16_t c2, bool state)
bool	**drawLine** (uint8_t r1, uint16_t c1, uint8_t r2, uint16_t c2, bool state)
bool	**drawVLine** (uint16_t c, uint8_t r1, uint8_t r2, bool state)
bool	**drawRectangle** (uint8_t r1, uint16_t c1, uint8_t r2, uint16_t c2, bool state)
bool	**getBuffer** (uint16_t col, uint8_t size, uint8_t *pd)
uint8_t	**getColumn** (uint8_t c)
bool	**getPoint** (uint8_t r, uint16_t c)
bool	**setBuffer** (uint16_t col, uint8_t size, uint8_t *pd)
bool	**setColumn** (uint16_t c, uint8_t value)
bool	**setPoint** (uint8_t r, uint16_t c, bool state)
bool	**setRow** (uint8_t r, uint8_t value)
bool	**setRow** (uint8_t startDev, uint8_t endDev, uint8_t r, uint8_t value)

bool	**transform** (**transformType_t** ttype)
bool	**transform** (uint8_t startDev, uint8_t endDev, **transformType_t** ttype)
void	**update** (**controlValue_t** mode)
void	**update** (void)
void	**wraparound** (**controlValue_t** mode)

Methods for managing specific devices or display buffers.

bool	**clear** (uint8_t buf)
uint8_t	**getColumn** (uint8_t buf, uint8_t c)
uint8_t	**getRow** (uint8_t buf, uint8_t r)
bool	**setColumn** (uint8_t buf, uint8_t c, uint8_t value)
bool	**setRow** (uint8_t buf, uint8_t r, uint8_t value)
bool	**transform** (uint8_t buf, **transformType_t** ttype)
void	**update** (uint8_t buf)

Methods for font and characters.

uint8_t	**getChar** (uint8_t c, uint8_t size, uint8_t *buf)
uint8_t	**setChar** (uint16_t col, uint8_t c)
bool	**setFont** (**fontType_t** *f)
uint8_t	**getMaxFontWidth** (void)
fontType_t *	**getFont** (void)

Member Enumeration Documentation

◆ controlRequest_t

enum **MD_MAX72XX::controlRequest_t**

Control Request enumerated type.

This enumerated type is used with the **control()** method to identify the control action request.

Enumerator	
SHUTDOWN	Shut down the MAX72XX. Requires ON/OFF value. Library default is OFF.
SCANLIMIT	Set the scan limit for the MAX72XX. Requires numeric value [0..MAX_SCANLIMIT]. Library default is all on.
INTENSITY	Set the LED intensity for the MAX72XX. Requires numeric value [0..MAX_INTENSITY]. LIbrary default is MAX_INTENSITY/2.
TEST	Set the MAX72XX in test mode. Requires ON/OFF value. Library default is OFF.
DECODE	Set the MAX72XX 7 segment decode mode. Requires ON/OFF value. Library default is OFF.
UPDATE	Enable or disable auto updates of the devices from the library. Requires ON/OFF value. Library default is ON.
WRAPAROUND	Enable or disable wraparound when shifting (circular buffer). Requires ON/OFF value. Library default is OFF.

◆ controlValue_t

enum **MD_MAX72XX::controlValue_t**

Control Value enumerated type.

This enumerated type is used with the **control()** method as the ON/OFF value for a control request. Other values may be used if numeric data is required.

Enumerator	
OFF	General OFF status request.
ON	General ON status request.

◆ transformType_t

enum **MD_MAX72XX::transformType_t**

Transformation Types enumerated type.

This enumerated type is used in the **transform()** methods to identify a specific transformation of the display data in the device buffers.

Enumerator	
TSL	Transform Shift Left one pixel element.
TSR	Transform Shift Right one pixel element.
TSU	Transform Shift Up one pixel element.
TSD	Transform Shift Down one pixel element.
TFLR	Transform Flip Left to Right.
TFUD	Transform Flip Up to Down.
TRC	Transform Rotate Clockwise 90 degrees.
TINV	Transform INVert (pixels inverted)

Constructor & Destructor Documentation

◆ MD_MAX72XX() [1/2]

```
MD_MAX72XX::MD_MAX72XX ( uint8_t  dataPin,
                         uint8_t  clkPin,
                         uint8_t  csPin,
                         uint8_t  numDevices = 1
                        )
```

Class Constructor - arbitrary digital interface.

Instantiate a new instance of the class. The parameters passed are used to connect the software to the hardware. Multiple instances may co-exist but they should not share the same hardware CS pin (SPI interface).

Parameters

dataPin	output on the Arduino where data gets shifted out.
clkPin	output for the clock signal.
csPin	output for selecting the device.
numDevices	number of devices connected. Default is 1 if not supplied. Memory for device buffers is dynamically allocated based on this parameter.

◆ MD_MAX72XX() [2/2]

```
MD_MAX72XX::MD_MAX72XX ( uint8_t  csPin,
                         uint8_t  numDevices = 1
                        )
```

Class Constructor - SPI hardware interface.

Instantiate a new instance of the class. The parameters passed are used to connect the software to the hardware. Multiple instances may co-exist but they should not share the same hardware CS pin (SPI interface). The dataPin and the clockPin are defined by the Arduino hardware definition (SPI MOSI and SCK signals).

Parameters

csPin	output for selecting the device.
numDevices	number of devices connected. Default is 1 if not supplied. Memory for device buffers is dynamically allocated based on this parameter.

◆ ~MD_MAX72XX()

MD_MAX72XX::~MD_MAX72XX (void)

Class Destructor.

Released allocated memory and does the necessary to clean up once the object is no longer required.