

microui

*User Manual*



**MICROEJ<sup>®</sup>**

Reference:	TLT-XXX-MAN-microui-microui
Version:	1.0.3
Revision:	XXX

---

## Confidentiality & Intellectual Property

All rights reserved. Information, technical data and tutorials contained in this document are confidential and proprietary under copyright Law of Industrial Smart Software Technology (IS2T S.A.) operating under the brand name MicroEJ®. Without written permission from IS2T S.A., *copying or sending parts of the document or the entire document by any means to third parties is not permitted*. Granted authorizations for using parts of the document or the entire document do not mean IS2T S.A. gives public full access rights.

The information contained herein is not warranted to be error-free. IS2T® and MicroEJ® and all relative logos are trademarks or registered trademarks of IS2T S.A. in France and other Countries.

Java™ is Sun Microsystems' trademark for a technology for developing application software and deploying it in cross-platform, networked environments. When it is used in this documentation without adding the ™ symbol, it includes implementations of the technology by companies other than Sun.

Java™, all Java-based marks and all related logos are trademarks or registered trademarks of Sun Microsystems Inc, in the United States and other Countries.

Other trademarks are proprietary of their authors.

---

---

# Table of Contents

1. File Documentation .....	1
1.1. LLDW_PAINTER_impl.c File Reference .....	1
1.1.1. Macros .....	1
1.1.2. Functions .....	2
1.2. LLUI_PAINTER_impl.c File Reference .....	3
1.2.1. Macros .....	3
1.2.2. Functions .....	4

---

# Chapter 1. File Documentation

## 1.1. LLDW\_PAINTER\_impl.c File Reference

```
#include "LLDW_PAINTER_impl.h"
```

```
#include "dw_drawing.h"
```

```
#include "LLUI_DISPLAY.h"
```

### 1.1.1. Macros

- #define LOG\_DRAW\_START LLUI\_DISPLAY\_logDrawingStart(CONCAT\_DEFINES(LOG\_DRAW\_, fn))
- #define LOG\_DRAW\_END LLUI\_DISPLAY\_logDrawingEnd(CONCAT\_DEFINES(LOG\_DRAW\_, fn))
- #define DRAWING\_PAINTER\_NATIVE\_NAME (CONCAT\_DEFINES(DRAWING\_PAINTER\_NATIVE\_PREFIX, fn))
- #define LOG\_DRAW\_drawThickFadedPoint 100
- #define LOG\_DRAW\_drawThickFadedLine 101
- #define LOG\_DRAW\_drawThickFadedCircle 102
- #define LOG\_DRAW\_drawThickFadedCircleArc 103
- #define LOG\_DRAW\_drawThickFadedEllipse 104
- #define LOG\_DRAW\_drawThickLine 105
- #define LOG\_DRAW\_drawThickCircle 106
- #define LOG\_DRAW\_drawThickEllipse 107
- #define LOG\_DRAW\_drawThickCircleArc 108
- #define LOG\_DRAW\_drawFlippedImage 200
- #define LOG\_DRAW\_drawRotatedImageNearestNeighbor 201
- #define LOG\_DRAW\_drawRotatedImageBilinear 202
- #define LOG\_DRAW\_drawScaledImageNearestNeighbor 203

- #define LOG\_DRAW\_drawScaledImageBilinear 204

## 1.1.2. Functions

- void DRAWING\_PAINTER\_NATIVE ( drawThickFadedPoint , MICROUI\_GraphicsContext \* gc, jint x, jint y, jint thickness, jint fade)
- void DRAWING\_PAINTER\_NATIVE ( drawThickFadedLine , MICROUI\_GraphicsContext \* gc, jint startX, jint startY, jint endX, jint endY, jint thickness, jint fade, DRAWING\_Cap startCap, DRAWING\_Cap endCap)
- void DRAWING\_PAINTER\_NATIVE ( drawThickFadedCircle , MICROUI\_GraphicsContext \* gc, jint x, jint y, jint diameter, jint thickness, jint fade)
- void DRAWING\_PAINTER\_NATIVE ( drawThickFadedCircleArc , MICROUI\_GraphicsContext \* gc, jint x, jint y, jint diameter, jfloat startAngle, jfloat arcAngle, jint thickness, jint fade, DRAWING\_Cap start, DRAWING\_Cap end)
- void DRAWING\_PAINTER\_NATIVE ( drawThickFadedEllipse , MICROUI\_GraphicsContext \* gc, jint x, jint y, jint width, jint height, jint thickness, jint fade)
- void DRAWING\_PAINTER\_NATIVE ( drawThickLine , MICROUI\_GraphicsContext \* gc, jint startX, jint startY, jint endX, jint endY, jint thickness)
- void DRAWING\_PAINTER\_NATIVE ( drawThickCircle , MICROUI\_GraphicsContext \* gc, jint x, jint y, jint diameter, jint thickness)
- void DRAWING\_PAINTER\_NATIVE ( drawThickEllipse , MICROUI\_GraphicsContext \* gc, jint x, jint y, jint width, jint height, jint thickness)
- void DRAWING\_PAINTER\_NATIVE ( drawThickCircleArc , MICROUI\_GraphicsContext \* gc, jint x, jint y, jint diameter, jfloat startAngle, jfloat arcAngle, jint thickness)
- void DRAWING\_PAINTER\_NATIVE ( drawFlippedImage , MICROUI\_GraphicsContext \* gc, MICROUI\_Image \* img, jint regionX, jint regionY, jint width, jint height, jint x, jint y, DRAWING\_Flip transformation, jint alpha)
- void DRAWING\_PAINTER\_NATIVE ( drawRotatedImageNearestNeighbor , MICROUI\_GraphicsContext \* gc, MICROUI\_Image \* img, jint x, jint y, jint rotationX, jint rotationY, jfloat angle, jint alpha)
- void DRAWING\_PAINTER\_NATIVE ( drawRotatedImageBilinear , MICROUI\_GraphicsContext \* gc, MICROUI\_Image \* img, jint x, jint y, jint rotationX, jint rotationY, jfloat angle, jint alpha)
- void DRAWING\_PAINTER\_NATIVE ( drawScaledImageNearestNeighbor , MICROUI\_GraphicsContext \* gc, MICROUI\_Image \* img, jint x, jint y, jfloat factorX, jfloat factorY, jint alpha)
- void DRAWING\_PAINTER\_NATIVE ( drawScaledImageBilinear , MICROUI\_GraphicsContext \* gc, MICROUI\_Image \* img, jint x, jint y, jfloat factorX, jfloat factorY, jint alpha)

## Detailed Description

This file implements all "Drawing" (MicroUI extended library) drawing native functions.

See also: . LLGRAPHICS\_EXT\_impl.h file comment

Author: . MicroEJ Developer Team

Version: . 1.0.3

Date: . 8 December 2020

Definition in file C:/jenkins/workspace/M0092\_UI-CCO-LLMicroUI/bsp-llmicroui/target~/ccomponent-Working/bsp/ui/src/LLDW\_PAINTER\_impl.c

## 1.2. LLUI\_PAINTER\_impl.c File Reference

```
#include "LLUI_PAINTER_impl.h"
```

```
#include "ui_drawing.h"
```

```
#include "LLUI_DISPLAY.h"
```

### 1.2.1. Macros

- #define LOG\_DRAW\_START LLUI\_DISPLAY\_logDrawingStart(CONCAT\_DEFINES(LOG\_DRAW\_, fn))
- #define LOG\_DRAW\_END LLUI\_DISPLAY\_logDrawingEnd(CONCAT\_DEFINES(LOG\_DRAW\_, fn))
- #define MICROUI\_PAINTER\_NATIVE\_NAME (CONCAT\_DEFINES(MICROUI\_PAINTER\_NATIVE\_PREFIX, fn))
- #define LOG\_DRAW\_writePixel 1
- #define LOG\_DRAW\_drawLine 2
- #define LOG\_DRAW\_drawHorizontalLine 3
- #define LOG\_DRAW\_drawVerticalLine 4
- #define LOG\_DRAW\_drawRectangle 5
- #define LOG\_DRAW\_fillRectangle 6
- #define LOG\_DRAW\_drawRoundedRectangle 8
- #define LOG\_DRAW\_fillRoundedRectangle 9

- `#define LOG_DRAW_drawCircleArc 10`
- `#define LOG_DRAW_fillCircleArc 11`
- `#define LOG_DRAW_drawEllipseArc 12`
- `#define LOG_DRAW_fillEllipseArc 13`
- `#define LOG_DRAW_drawEllipse 14`
- `#define LOG_DRAW_fillEllipse 15`
- `#define LOG_DRAW_drawCircle 16`
- `#define LOG_DRAW_fillCircle 17`
- `#define LOG_DRAW_drawARGB 18`
- `#define LOG_DRAW_drawImage 19`

## 1.2.2. Functions

- `static void _check_bound ( jint max, jint * bound, jint * size, jint * origin)`
- `void MICROUI_PAINTER_NATIVE ( writePixel , MICROUI_GraphicsContext * gc, jint x, jint y)`
- `void MICROUI_PAINTER_NATIVE ( drawLine , MICROUI_GraphicsContext * gc, jint startX, jint startY, jint endX, jint endY)`
- `void MICROUI_PAINTER_NATIVE ( drawHorizontalLine , MICROUI_GraphicsContext * gc, jint x, jint y, jint length)`
- `void MICROUI_PAINTER_NATIVE ( drawVerticalLine , MICROUI_GraphicsContext * gc, jint x, jint y, jint length)`
- `void MICROUI_PAINTER_NATIVE ( drawRectangle , MICROUI_GraphicsContext * gc, jint x, jint y, jint width, jint height)`
- `void MICROUI_PAINTER_NATIVE ( fillRectangle , MICROUI_GraphicsContext * gc, jint x, jint y, jint width, jint height)`
- `void MICROUI_PAINTER_NATIVE ( drawRoundedRectangle , MICROUI_GraphicsContext * gc, jint x, jint y, jint width, jint height, jint cornerEllipseWidth, jint cornerEllipseHeight)`
- `void MICROUI_PAINTER_NATIVE ( fillRoundedRectangle , MICROUI_GraphicsContext * gc, jint x, jint y, jint width, jint height, jint cornerEllipseWidth, jint cornerEllipseHeight)`
- `void MICROUI_PAINTER_NATIVE ( drawCircleArc , MICROUI_GraphicsContext * gc, jint x, jint y, jint diameter, jfloat startAngle, jfloat arcAngle)`
- `void MICROUI_PAINTER_NATIVE ( drawEllipseArc , MICROUI_GraphicsContext * gc, jint x, jint y, jint width, jint height, jfloat startAngle, jfloat arcAngle)`

- void MICROUI\_PAINTER\_NATIVE ( fillCircleArc , MICROUI\_GraphicsContext \* gc, jint x, jint y, jint diameter, jfloat startAngle, jfloat arcAngle)
- void MICROUI\_PAINTER\_NATIVE ( fillEllipseArc , MICROUI\_GraphicsContext \* gc, jint x, jint y, jint width, jint height, jfloat startAngle, jfloat arcAngle)
- void MICROUI\_PAINTER\_NATIVE ( drawEllipse , MICROUI\_GraphicsContext \* gc, jint x, jint y, jint width, jint height)
- void MICROUI\_PAINTER\_NATIVE ( fillEllipse , MICROUI\_GraphicsContext \* gc, jint x, jint y, jint width, jint height)
- void MICROUI\_PAINTER\_NATIVE ( drawCircle , MICROUI\_GraphicsContext \* gc, jint x, jint y, jint diameter)
- void MICROUI\_PAINTER\_NATIVE ( fillCircle , MICROUI\_GraphicsContext \* gc, jint x, jint y, jint diameter)
- void MICROUI\_PAINTER\_NATIVE ( drawImage , MICROUI\_GraphicsContext \* gc, MICROUI\_Image \* img, jint regionX, jint regionY, jint width, jint height, jint x, jint y, jint alpha)

## Detailed Description

This file implements all MicroUI drawing native functions.

See also: . LLGRAPHICS\_impl.h file comment

Author: . MicroEJ Developer Team

Version: . 1.0.3

Date: . 8 December 2020

Definition in file C:/jenkins/workspace/M0092\_UI-CCO-LLMicroUI/bsp-llmicroui/target~/ccomponent-Working/bsp/ui/src/LLUI\_PAINTER\_impl.c