TOSHIBA

Produce name:

Model

WD-111 Control Library Specification (Based on Windows)

First Edition: September 18, 2009

TOSHIBA TEC CORPORATION Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

Revision Record

No. EAA-02965

WD-111 Control Library Specifications (Based on Windows)

Server Infrastructure Software & Key Components Group

Rev. No	Date	Pages	Description and reasons for change	Person in charge	Inspected by	Approved by
---	Sep 18, 2009	---	Newly published.			

Distributed to:

TOSHIBA TEC CORPORATION Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

TABLE OF CONTENTS

1 INTRODUCTION 1-1

2 OVERVIEW 2-1

2.1 REQUIREMENTS 2-1
2.2 SYSTEM REQUIREMENTS 2-1
2.2.1 Hardware Environment 2-1
2.2.2 Software Environment 2-1
2.3 DEVELOPMENT ENVIRONMENT 2-1

3 SOFTWARE CONFIGURATION 3-1

3.1 OVERVIEW 3-1
3.2 INTERNAL STRUCTURE 3-2
3.3 CONTROL DATA 3-3

4 FUNCTION SPECIFICATIONS 4-1

4.1 FUNCTION LIST 4-1
4.2 DETAILS ON FUNCTIONS 4-2

5 CONFIGURATION 5-1

6 ERROR MESSAGES 6-1

7 DISPLAY CODE TABLE 7-1

7.1 DISPLAYING CHARACTERS 7-2

8 STATE TRANSITION 8-3

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

1 Introduction

This is a specification for a library designed to control the line display specifically for ST-B10. Note that the ST-B10 runs on Windows XP.

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

2 Overview

2.1 Requirements

1. The library must run on the WD-111 line display.
2. The library must run as a dynamic link library (DLL) on Windows.
3. The parameters must be configurable in the registry.
4. Multithreads must be allowed within a single process.

2.2 System Requirements

2.2.1 Hardware Environment

WD-111

2.2.2 Software Environment

Windows XP Pro SP3 / WEPOS1.1 / Windows Embedded POS Ready 2009

2.3 Development Environment

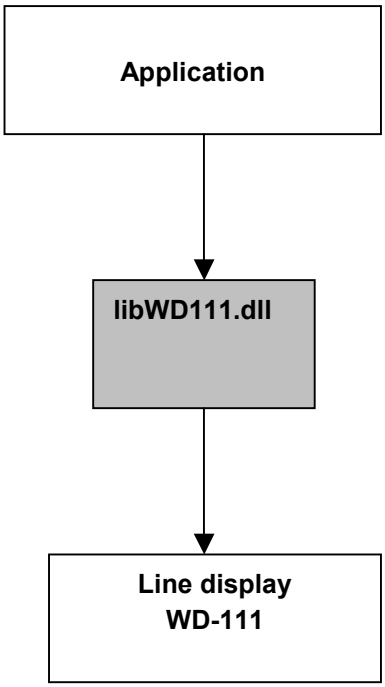
The following programming tool and language are used for development.

OS	:	Windows XP Pro SP3
Programming tool	:	VisualC++ 6 SP5
Programming language	:	C++
Module name	:	libWD111.dll

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

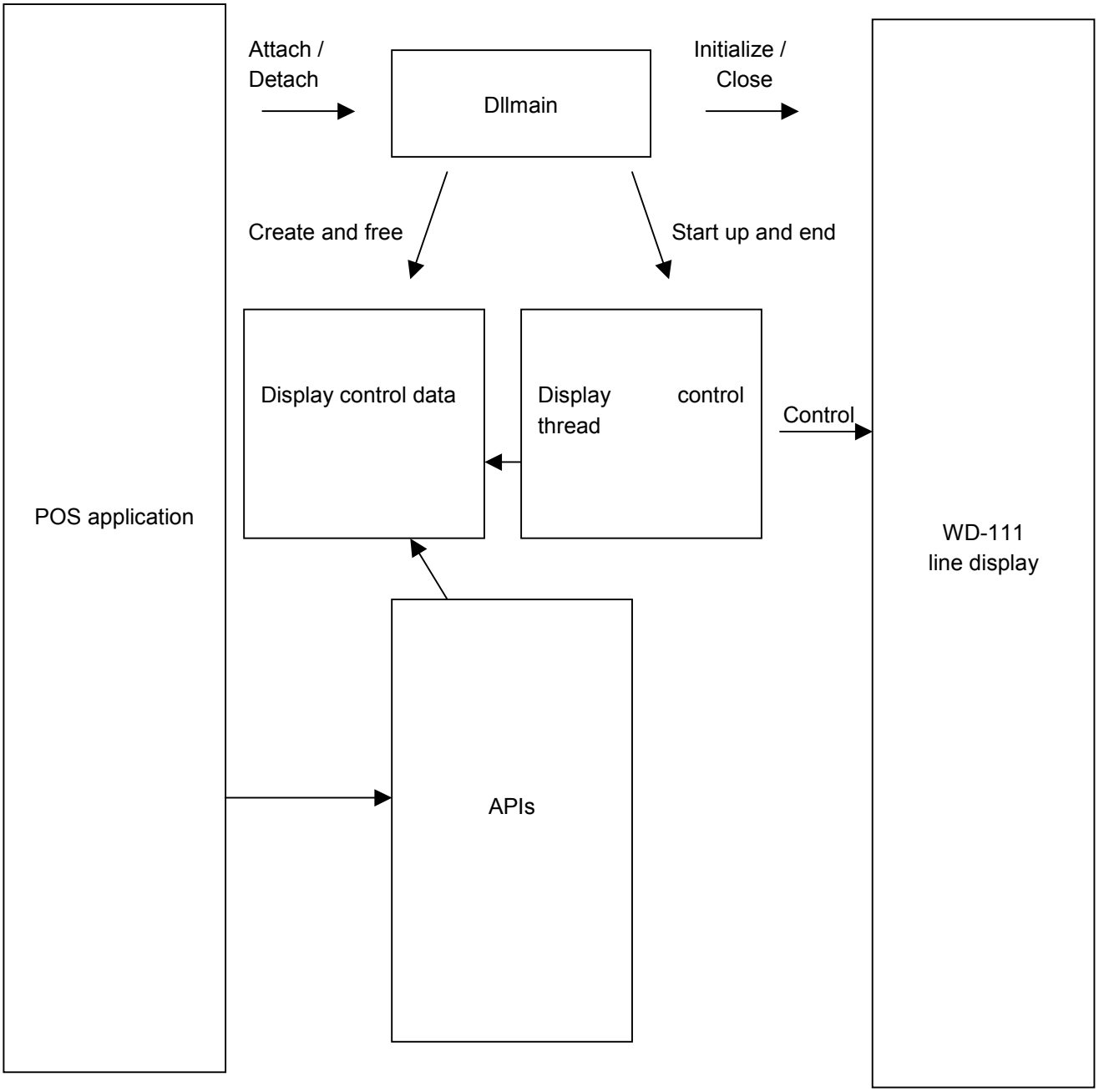
3 Software Configuration

3.1 Overview



 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

3.2 Internal Structure



 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

3.3 Control Data

```
typedef struct      __liuwd111info__      {
        HANDLE    hMutex;                 // Exclusive handle
        HANDLE    hMutexAPI;              // Exclusive handle (for multithreads)
        HANDLE    hThread;                // Thread handle
        HANDLE    hStartEvent;            // Event handle (start)
        HANDLE    hEndEvent;              // Event handle (end)
        short     sResult;                // Processing result
        char      szDev[256+1];           // Device
        HANDLE    hDevCom;                // Device handle
        char      szBuff[128];            // Data buffer (128 bytes)
        BOOL      bBrinkSW;               // Blink status FALSE: OFF TRUE: ON
        BOOL      bBrinkSts;              // Current blink status FALSE: Not displayed TRUE: Displayed
        BOOL      bBrink[128];            // Blink attribute  FALSE: OFF TRUE: ON
        WORD      wBInterval;             // Blink interval (msec)
        BOOL      bScroll;                // Scroll attribute FALSE: OFF TRUE: ON
        WORD      wSSpeed;                // Scroll speed (msec)
        WORD      wSPosition;             // Scroll position (beginning of display)
        unsigned short usNextBrinkTime;   // Remaining wait time for the subsequent blink process (msec)
        unsigned short usNextScrollTime;  // Remaining wait time for the subsequent scroll process (msec)
        unsigned short usWaitTime;        // Sleep time (msec)
} LIUWD111INFO;
```

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

4 Function Specifications

4.1 Function List

This is a list of the functions in this library.
For further information regarding each function, please refer to a relevant subsection.

Function name	Description
liuwd_init	Initializing the display
liuwd_exc	Displaying characters
liuwd_exc2	Displaying characters (specifying a specific display position)
liuwd_scr	Controlling scrolling of the displayed text in a horizontal direction
liuwd_ctl	Controlling the display
str_blink	Decorating a character string in order to make it blink
liuwd_blinkinteval	Specifying a blink interval
liuwd_close	Closing the library

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

4.2 Details on Functions

This subsection gives the details on each function.

#include "liuwd111.h"

int liuwd_init();

Description: This function is intended to initialize the line display.

Argument: None

Return value: 0: Normal end Value other than 0: Error

Details:
If liuwd_init is called, the connected WD-111 line display will be initialized.
The WD-111 line display will be initialized again even if it has been already initialized. The displayed text is cleared when the WD-111 line display is initialized.

liuwd_init must be called before LIU is used. Calling another LIU function before calling liuwd_init results in an error

Example:

```
// To perform initialization
nRet = liuwd_init();
```

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

#include "liuwd111.h"

int liuwd_exc(char *buffer);

Description: This function is intended to display text on the line display.

Argument: char *buffer : Pointer to the buffer where the display data is stored
The data must be null-terminated.

Return value: 0: Normal end Value other than 0: Error

Details:
The data in the buffer must comply with the codes applicable to WD-111 (*1).

*1
For further information regarding the displayed text, see Section 7.

Note:
If only a dot (".") is specified or if a dot is specified at the beginning of the argument, the dot will always appear at the right end of the display.
In this case, blinking may not be performed properly.

Example:

// Displaying 3 characters
nRet = liuwd_exc("123");
Display sample:

1	2	3								

// Displaying 11 characters
nRet = liuwd_exc("1234567890a");
Display sample:

1	2	3	4	5	6	7	8	9	0	A

// Displaying 22 characters with "."
nRet = liuwd_exc("1.2.3.4.5.6.7.8.9.0.a.");
Display sample:

1.	2.	3.	4.	5.	6.	7.	8.	9.	0.	A.

// Displaying 22 characters with "." (Only "." is displayed.)
nRet = liuwd_exc("Δ.Δ.Δ.Δ.Δ.Δ.Δ.Δ.Δ.Δ.Δ.");
"Δ" represents a single-width space.
Display sample:

.	.	.	.	.	.	.	.	.	.	.

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

#include "liuwd111.h"

int liuwd_exc2(int column, char *buffer);

Description: This function is intended to display text in specified columns of the line display.

Argument: int column : Specifying the column from which characters are displayed. (1 to 11)
Determined by calculation based on the number of single-width characters (C varies depending on screen mode.)
char *buffer : Pointer to the buffer where the display data is stored
The data must be null-terminated.

Return value: 0: Normal end Value other than 0: Error

Details:
The data in the buffer must comply with the codes applicable to WD-111 (*1).

“column” is intended to specify a column from which characters are displayed.

*1
For further information regarding the displayed text, see Section 7.

Note:
If only a dot (“.”) is specified or if a dot is specified at the beginning of the argument, the dot will always appear at the right end of the display.
In this case, blinking may not be performed properly.

Example:

// Displaying 3 characters starting column 5
nRet = liuwd_exc2(5,“123”);
Display sample:

				1	2	3				

// Displaying 11 characters starting column 6
nRet = liuwd_exc2(6,“1234567890a”);
Display sample:

					1	2	3	4	5	6

// Displaying 22 characters with “.” starting column 6
nRet = liuwd_exc2(6,“1.2.3.4.5.6.7.8.9.0.a.”);
Display sample:

					1.	2.	3.	4.	5.	6.

// Displaying 6 characters with “.” starting column 2 (Only “.” is displayed.)
nRet = liuwd_exc2(2,“Δ.Δ.Δ.”);
“Δ” represents a single-width space.
Display sample:

	.	.	.							

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

#include "liuwd111.h"

int liuwd_scr(int func, WORD time, char *buffer);

Description: This function is intended to control the scrolling of the displayed text in a horizontal direction.

Argument: int func : = 0: Start scrolling =1: Stop scrolling
int time : Scroll speed (msec)
char *buffer: Pointer to the buffer where the scroll display data is stored

Return value: 0: Normal end Value other than 0: Error

Details:
General motion

- Characters are scrolled column by column from right to left. No blank must be present between characters.
- A character string to be scrolled can be replaced by reconfiguring the start scroll command.
- If liuwd_exc() and liuwd_exc2() are used to display text in the line being scrolled, the displayed text will be updated, but scrolling will be stopped.

Stopping of scrolling

- The value of the parameter “time,buffer” is not used.
- Once scrolling has been stopped, the data displayed until just before scrolling was stopped will not be displayed.
- (All columns of the display are filled with a space. The attribute is “normal.”)

Details:
The data in the buffer must comply with the codes applicable to WD-111 (*1).
A maximum of 128 characters (including “.”) is acceptable.

*1
For further information regarding the displayed text, see Section 7.

Note:
If only a dot (“.”) is specified or if a dot is specified at the beginning of the argument, the dot will always appear at the right end of the display.
In this case, blinking may not be performed properly.

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

Example:

// Starting to scroll 3 characters (The scroll speed is set at 500 msec.)
nRet = liuwd_scr(1,500,“123”);
Display sample in an initial state:

1	2	3	1	2	3	1	2	3	1	2

// Starting to scroll 4 characters (The scroll speed is set at 500 msec.)
nRet = liuwd_scr(1,500,“123Δ”);
“Δ” represents a single-width space.
Display sample in an initial state:

1	2	3		1	2	3		1	2	3

// Stopping scrolling
nRet = liuwd_scr(0,0,NULL);
Display sample in an initial state:

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

#include "liuwd111.h"

int liuwd_ctl(int func, int para);

Description: This function is intended to control the line display.

Argument: int func : Specifying a display operation

Return value: 0: Normal end Value other than 0: Error

Details:
func
func=0: The entire displayed text is cleared.

Example:

```
// Clearing the entire displayed text
nRet = liuwd_ctl(0,0);
```

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

#include "liuwd111.h"

int str_blink(unsigned char *inpbuffer, int inpsize, unsigned char *outbuffer,
int *outsize, int *indisize);

Description: This function is intended to decorate a displayed character string in order to make it blink.

Argument:			
	unsigned char *inpbuffer	:	Pointer to the buffer where a character string to be decorated is stored
	int inpsize	:	Length of a character string to be decorated
	unsigned char *outbuffer	:	Pointer to the buffer where the decorated character string is outputted
	int *outsize	:	(Input) Size of outbuffer
		:	(Output) Length of the decorated character string
	int *indisize	:	Number of columns actually displayed on the LIU

Return value: 0: Normal end Value other than 0: Error

Details:
outbuffer (outsize)
outbuffer (outsize) must reserve space of at least 8 more characters than inpbuffer (inpsize).

This function is just intended to decorate selected characters in a character string. Therefore, further processing of the decorated characters allows them to be blinked at 2 different locations at the same time.

For instance, if normal character strings A and B are joined to character strings decorated for blinking α and β (e.g. A + α + B + β) and displayed using the functions including liu_exc, liu_exc2, and liu_src, character strings α and β will blink at the same time. In addition, character strings A and B will stay steady.

Also, the blink attribute defined for the character outputted most recently applies.
For instance, under the condition in which the blink attribute is ON for all 11 characters, if liu_exc2 is used to display 3 characters starting column 2 and the blink attribute is turned OFF (no decoration for blinking), the characters in column 1 and column 5 through 11 will continue to blink, but the ones in columns 2 through 4 will go steady.

On the other hand, under the condition in which the blink attribute is OFF for all 11 characters (no decoration for blinking), if liu_exc2 is used to display 3 characters starting column 2 and the blink attribute is turned ON, the characters in column 1 and columns 5 through 11 will remain steady, but the ones in columns 2 through 4 will start blinking.

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

Example:

```
// Decorating all 20 characters in order to make them blink
char   szOutbuff[28];
int   nOutLen;
int   nDispLen;

nOutLen = sizeof szOutbuff;
nRet = str_blink(“1234567890ABCDEFGHIJ”,20, szOutbuff,&nOutLen,&nDispLen);


// Decorating characters in columns 3 through 10 out of 20 characters
char   szInbuff[21];
char   szOutbuff[28];
int   nOutLen;
int   nDispLen;

memset(szInbuff, 0x00 ,sizeof szInbuff);
strcpy(szInbuff);
nOutLen = sizeof szOutbuff;
nRet = str_blink(&szInbuff[2],10, szOutbuff,&nOutLen,&nDispLen);
```

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

#include "liuwd111.h"

int liuwd_blinkinterval (WORD Interval);

Description: This function is intended to specify an interval between blinks.

Argument: WORD Interval : Interval between blinks (msec)

Return value: 0: Normal end Value other than 0: Error

Details:
This function is used to specify an interval between blinks when the displayed text is blinked.
An interval between blinks is defined in the unit of msec. An acceptable value range is between 1 and 65535.
If a value is out of the above range or if no value is specified, an interval between blinks will default to 1000 msec.
This value is specified tentatively. Therefore, it does not affect the registry.

Example:

```
// Setting the interval between blinks to 500 msec
nRet = liuwd_blinkinterval(500);

// Setting the interval between blinks to 1500 msec
nRet = liuwd_blinkinterval(1500);
```

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

#include "liuwd111.h"

int liuwd_close ();

Description: This function is intended to terminate access to the device or threads and close the process.

Argument: None

Return value: 0: Normal end Value other than 0: Error

Details:
This function is intended to terminate the threads used in the library and close the device in order to end the process, because when the process is ended on Windows, the thread in action may be forcibly terminated before DLL detach. (For further information, see the specifications for Windows.)
This function is called in order to avoid this problem, terminate the thread in action, and get ready for the end of the process.

After this function is called, all functions including liuwd_close result in an error.

Example:

```
nRet = liuwd_close ();
```

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

5 Configuration

The default values stored in the registry "¥¥HKEY_LOCAL_MACHINE¥SOFTWARE¥TEC¥WD111" must be used.

- **device (REG_SZ)**
  A device to use is specified.
  If no device is specified, "COM1" will be used by default.

- **blinkinterval (REG_DWORD)**
  An interval between blinks is specified when the displayed text blinks.
  It must be defined in the unit msec. An acceptable value range is between 1 and 65535.
  If a value is out of the above range or if no value is specified, an interval between blinks will default to 1000 msec.

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

6 Error Messages

Definition of an error	Value	Description
WD111_ERR_OPEN	-1000	Open error
WD111_ERR_CLOSE	-1001	Close error
WD111_ERR_PRM	-1002	Parameter error
WD111_ERR_SEND	-1003	Command transmission error
WD111_ERR_ILL	-1004	Other errors (internal error)
WD111_ERR_INIT	-1005	Non-initialization error

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

7 Display Code Table

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
20H																
30H																
40H																
50H																
60H																
70H																

* The codes other than the above may be used to display characters depending on the hardware.

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

7.1 Displaying Characters

In general, characters are displayed by sending an appropriate code to the line display. For instance, if "0123456789A" is sent to the line display, the characters will be displayed as below.

0	1	2	3	4	5	6	7	8	9	A

Normally, only 11 characters can be displayed as illustrated above. However, if "." is included, a maximum of 22 characters can be displayed. (For instance, "0.1.2.3.4.5.6.7.8.9.A." is sent to the line display.)

0.	1.	2.	3.	4.	5.	6.	7.	8.	9.	A.

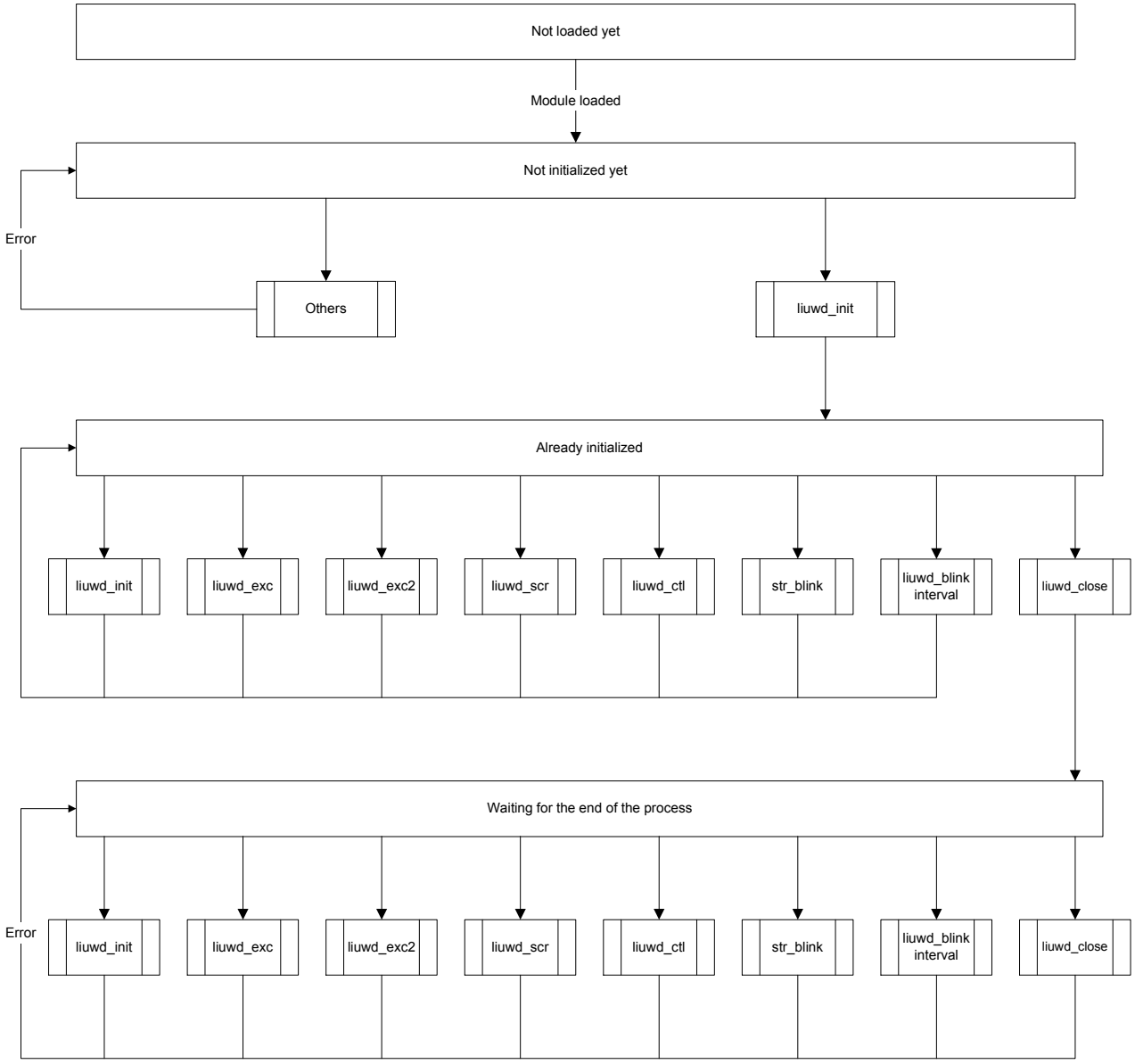
It is only when both a character and a dot "." are displayed in combination that more than 11 characters can be displayed on the line display. If the display shows only ".", it will not be able to display more than 11 characters. For instance, if ". . . . . " is sent, the characters will be displayed as below.

.	.	.	.	.	.	.	.	.	.	.

In addition, when ". . . . . " is sent to the line display, the dots will be displayed on the line display in the same manner as in when "Δ. Δ. Δ. Δ. Δ. Δ. Δ. Δ. Δ. Δ. Δ." is sent. (*Δ represents a space.)

 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.

8 State Transition



 Copyright©2009 TOSHIBA TEC Corporation All Rights Reserved.