

(Micro)PDF417 *Library* for the OPL972x



Application Development Manual
for the
(Micro)PDF417 library
version LSAV0102

©2004, Opticon Sensors Europe BV

Introduction

The (Micro)PDF417-Library for the OPL972x provides Opticon's OPL972x hand-held data collection terminals with the possibility to read PDF417 and MicroPDF417 bar codes.

The (Micro)PDF417-Library can be used to read the following (Micro)PDF417 variants:

- **PDF417**
- **MicroPDF417** (all formats supported)
- **EAN•UCC Composite 2D-component:CC-A** (a MicroPDF417 variant,i.e. used on RSS and EAN128 symbols)
- **EAN•UCC Composite 2D-component:CC-B** (a MicroPDF417 variant,i.e. used on RSS and EAN128 symbols)
- **EAN•UCC Composite 2D-component:CC-C** (a PDF417 variant, i.e. used on EAN128 symbols)

The (Micro)PDF417-Library can not be used for the following purposes:

- Read any other type of 2D bar codes besides the ones that are listed above.

Important note:

*The (Micro)PDF417 library for the OPL972x requires **OS version LBxV0225** (or a later version), because earlier versions do not support external decoders.*

How to build an application using the library

To build an application, see the programming manual for the OPL-972x. The functions of the IrDA library can be linked by adding the library to the link-command, e.g.

```
tulink startup.rel LSAV0102.LIB main.rel etc.
```

In all the source files that make use of the functions of the IrDA library, include the file "pdf_lib.h".

To initialize the (Micro)PDF417 decoders in your application add the following function call in the beginning of the 'main'-function of your application:

```
init_PDF417_decoders( );
```

Like all the other decoders in the OS, the decoder options of the MicroPDF and PDF417 decoders can be changed by using so-called menu-options. This can be done using the function:

```
int systemsetting(const char *options)
```

On the next pages the following content will be described:

Chapter 1 describes the use of the menu-options of the MicroPDF and PDF417 decoders.

Chapter 2 gives information on how to read EAN•UCC Composite Symbols (UCC/EAN-128, EAN/UPC or RSS Composite symbols) with adjacent 2D CC-A, CC-B or CC-C components

Appendix A contains the source code of an example program that demonstrates how to use the (Micro)PDF417 library.










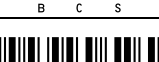

1. (Micro)PDF417-decoder menu options

Since OS version LBx0225 both 2-character menu-options as well as 3-character menu-options can be passed as a string to the 'systemsetting'-function:

```
int systemsetting(const char *options)
```

The 'systemsetting'-function distinguishes 3-character menu-options from 2-character options by checking if a menu-option starts with a '['. Therefor all 3-character menu-options must start with an extra '[' character.

The table below shows the list of all available menu options that can be used to set-up the PDF417 and MicroPDF417 decoders.

	Reset to default (Default: the PDF417 and MicroPDF decoder are turned on)
	Enable all codes (Enables all decoders including the PDF417 and the MicroPDF decoders)
	Disable all codes (Disables all decoders including the PDF417 and MicroPDF417 decoders)
	PDF417 only (Enables only the PDF417 decoder, incl. variant CC-C)
	MicroPDF417 only (Enables only the MicroPDF417 decoder, incl. variants CC-A and CC-B)
	Enable PDF417 (Enables the PDF417 decoder, incl. variant CC-C)
	Enable MicroPDF417 (Enables the MicroPDF417 decoder, incl. variants CC-A and CC-B)
	Disable PDF417 (Disables only the PDF417 decoder, incl. variant CC-C)
	Disable MicroPDF417 (Disables only the MicroPDF417 decoder, incl. variants CC-A and CC-B)
	MicroPDF, Transmit number of data columns (incl. CC-A & CC-B) (Transmit the number the of data columns at the first position of the decoder result of the MicroPDF417 decoder, this option is OFF as default)
	MicroPDF, Not transmit number of data columns (incl. CC-A & CC-B) (The number the of data columns is not transmitted at the first position of the decoder result of the MicroPDF417 decoder, this option is ON as default)

Important note:

Before these menu-options can be used, make sure that the (Micro)PDF417 decoders are initialized in your application by calling the function: 'init PDF417 decoders().

PDF417 Samples



Interdum feror cupidine partium
magnarum Europae vincendarum.



Opticon Sensors Europe B.V.
Opaallaan 35
2132 XV Hoofddorp

MicroPDF417 Samples



0123456789012345
67890123456789



0123456789012345678901234567
8901234567890101234567890123
4567890123456789012345678901

2. Decoding EAN•UCC Composite Symbolologies

2.1. Introduction

EAN•UCC Composite Symbolologies consist of a linear component (encoding the item's primary identification) and an adjacent 2D Composite Component (encoding supplementary data, such as a batch number or expiration date). A composite symbol always includes a linear component so that the primary data can be read by all scanning technologies. The secondary data, which is always encoded in a multi-row 2D composite symbol, can be read by 2D-imagers or by 1D-scanners with 2D-capabilities like the OPL972x with this library.







The complete list of permissible combinations EAN•UCC Composite Symbolologies can be found below.

Linear component	CC-A / CC-B	CC-C
UPC-A and EAN13	Yes (4 data columns)	No
EAN-8	Yes (3 data columns)	No
UPC-E	Yes (2 data columns)	No
UCC/EAN-128	Yes (4 data columns)	Yes (variable width)
RSS-14	Yes (4 data columns)	No
RSS-14 Stacked	Yes (2 data columns)	No
RSS-14 Stacked Omnidirectional	Yes (2 data columns)	No
RSS Limited	Yes (3 data columns)	No
RSS Expanded	Yes (4 data columns)	No
RSS Expanded Stacked	Yes (4 data columns)	No

2.2. Decoding EAN•UCC Composite symbols containing an RSS family linear component

Linear RSS symbols can be decoded since OS version LBx0225. The whole RSS decoder family (RSS-14, RSS-Limited and RSS-Expanded) is **disabled** in the default configuration of the OPL972x.

To enable the 3 main RSS decoders the following menu-options can be used:

	RSS-14 only (Enables only the RSS-14 decoder)
	RSS-Limited only (Enables only the RSS-limited decoder)
	RSS-Expanded only (Enable only the RSS-expanded decoder)
	Enables the RSS-14 decoder
	Enables the RSS-Limited decoder
	Enables the RSS-Expanded decoder

When RSS symbols are used as the linear component of an EAN•UCC Composite symbol, they contain a link-flag to indicate the presence of an adjacent 2D composite component.

This link-flag can be transmitted (or not) by using the following two menu-options:

DS	Transmit RSS Link flag (AI) (Transmit a link-flag character at the first position of the RSS decoder result. '0' indicates no 2D component present, '1' indicates an adjacent 2D composite comp.)
DT	Not transmit RSS Link flag (Do not transmit a link-flag character, default)

Note: the transmission of the link-flag is disabled in the default configuration)

The different members of the RSS family may only be linked to CC-A and CC-B composite component with a certain amount of data-columns.

Therefore only the following combinations are possible/valid:

- RSS-14 symbols may only be linked to CC-A or CC-B symbols with 2 or 4 data columns
- RSS-Limited symbols may only be linked to CC-A or CC-B symbols with 3 data columns
- RSS-Expanded symbols may only be linked to CC-A or CC-B symbols with 4 data columns

So, to successfully combine RSS symbols with a CC-A or CC-B symbol, it's necessary to know the amount data columns of a decoded CC-A, CC-B symbol.
 By default the number of data columns isn't transmitted in the result string of the MicroPDF417 decoder, this can be changed by enabling the 'Transmit number of data columns' menu-option.
 If the 'Transmit number of data columns' menu-option is enabled, then the first character of the MicroPDF decoder result string will indicate the number of data columns. ('1' -> 1 data columns, '2' -> 2 data columns, etc.)

[BI0	MicroPDF, Transmit number of data columns (incl. CC-A & CC-B) (Transmit the number the of data columns at the first position of the decoder result of the MicroPDF417 decoder, this option is OFF as default)
[BI1	MicroPDF, Not transmit number of data columns (incl. CC-A & CC-B) (The number the of data columns is not transmitted at the first position of the decoder result of the MicroPDF417 decoder, this option is ON as default)

Using the 'Transmit number of data columns' options of the MicroPDF417 decoder and the 'Transmit link flag' option of the RSS decoder, the result strings of both decoders can be correctly combined to a fully decoded EAN•UCC Composite symbol.

In Appendix A, the source code of an example program can be found, which demonstrates the correct combining of RSS symbols with CC-A or CC-B composite symbols.

2.2.1. Examples of RSS Composite symbols



RSS-14 (Stacked) with CC-A
 Data: 019876543210987_012345



RSS-14 Stacked with CC-B
 Data: 010123456789012_012345678901234567890123456789012345678901234567890



RSS-14 with CC-A
 Data: 019876543210987_0123456789



RSS-14 with CC-B
 Data: 019876543210987_01234567890123456789012345678901234567890123456789



RSS-Limited with CC-A
 Data: 010123456789012_012345678901234567890123456789



RSS-Limited with CC-B
 Data: 010123456789012_012345678901234567890123456789012345678901234567890123456789



RSS-Expanded with CC-A
 Data: 01937123456789043103001234_911A2B3C4D5E



RSS-Expanded (Stacked) with CC-A
 Data: 010001234567890510ABCDEF_2112345678



RSS-Expanded (Stacked) with CC-B
 Data: 010001234567890510ABCDEF_01234567890123456789012345678901234567890123456789

2.3. Decoding EAN•UCC Composite symbols containing an UCC/EAN-128 linear component

UCC/EAN-128 symbols are a subset of Code 128 symbols commencing with a FNC1 character in the first symbol position after the start character.

To be able to distinguish an UCC/EAN-128 symbol from a normal Code 128 symbol the menu-option 'Enable EAN128' should be used. This will cause the Code 128 decoder to return a different returned code-ID in case an UCC/EAN128 symbol is read (EAN128 instead of CODE128).

OF	Disable EAN128 (This option is ON as default)
OG	Enable EAN128 (This option is OFF as default)

When used as the linear component of an EAN•UCC Composite symbol the UCC/EAN-128 symbol ends with a so-called 'Code Set'-character (encoded in the last symbol character position). This extra 'Code Set'-character serves as a link flag, which indicates the presence of an adjacent 2D Composite Component.

The used type of the extra 'Code Set'-character determines the type of the adjacent 2D Composite Component, which can be a MicroPDF417-based (CC-A or CC-B) or PDF417-based (CC-C) component.

The EAN128-link flag can be transmitted as an extra character at the first position of the decode result string using the menu-option: 'EAN128, Transmit link flag'.

OT	EAN128, Transmit link flag (Transmit link flag at the first position of the decoder result, this option is OFF as default) '0' indicates no 2D component present, '1' indicates an adjacent CC-A or CC-B component and '2' indicates an adjacent CC-C component)
OU	EAN128, Not transmit link flag (Link flag is not transmitted at the first position of the decoder result, this option is ON as default)

In case a CC-A or CC-B Composite Component is adjacent to an EAN-128 symbol, the number of data columns should always be 4. (In case of a CC-C Composite Component any number of columns is valid.)

In paragraph 2.2 'Decoding EAN•UCC Composite symbols containing an RSS family linear component' is explained how the number of data columns of an MicroPDF417 based symbol can be transmitted using menu-options: '[BI0]' and '[BI1]'.

Using the menu-options 'Enable EAN128 Emulation', 'Transmit link flag' and 'Transmit number of data columns' the result strings of an EAN128 and a (Micro)PDF417 symbol can be correctly combined to a fully decoded EAN•UCC Composite symbol.

In Appendix A, the source code of an example program can be found, which demonstrates the correct combining of UCC/EAN-128 symbols with CC-A, CC-B and CC-C Composite components.

2.3.1. Examples of EAN128 Composite symbols



UCC/EAN128 Composite symbol with CC-A
Data: 0103212345678906_21A1B2C3D4
E5F6G7H8



UCC/EAN128 Composite symbol with CC-C
Data: 00030123456789012340_02130123
456789093724<FNC1>101234567ABCDEFG

2.4. Decoding EAN•UCC Composite symbols containing an EAN/UPC linear component

In contrast to RSS and EAN128 Composite symbols, EAN/UPC symbols do not have a link flag to indicate the presence of an associated 2D Composite Component. Therefore it's more difficult to combine an EAN/UPC symbol to its adjacent 2D Composite Component in an application.

The different members of the EAN/UPC family may only be linked to CC-A and CC-B composite component with a certain amount of data-columns. The following combinations are possible:

- UPC-E symbols may only be linked to CC-A or CC-B symbols with 2 data columns
- EAN-8 symbols may only be linked to CC-A or CC-B symbols with 3 data columns
- EAN-13 and UPC-A symbols may only be linked to CC-A or CC-B symbols with 4 data columns

Because the number of data columns of a CC-A or CC-B Composite Component is related to the type of EAN/UPC symbol, it's necessary to know the number of data columns of the 2D Composite Component.

In paragraph 2.2 'Decoding EAN•UCC Composite symbols containing an RSS family linear component' is explained how the number of data columns of an MicroPDF417 based symbol can be transmitted using menu-options: '[BI0]' and '[BI1]'.

In Appendix 3, the source code of an example program can be found, which demonstrates the correct combining of RSS and UCC/EAN-128 symbols with CC-A, CC-B and CC-C Composite components.

The combining of EAN/UPC Composite symbols hasn't been implemented due to the absence of a link-flag in this type of symbols.

2.4.1. Examples of EAN/UPC Composite symbols



EAN-8 Composite symbol with CC-A
Data: 12345670_ABCDEFGHIJKL



EAN-8 Composite symbol with CC-B
Data: 0123456_A1B2C3D4E5F6G7H8I9
JKL MNOPQRSTUVWXYZ



EAN-13 Composite symbol with CC-A
Data: 3312346678903_991234-abcd



EAN-13 Composite symbol with CC-B
Data: 012345678901_ABCDEFGHIJKLMNOP
QRSTUVWXYZ01234567890-ABCDEFGHIJK



UPC-E Composite symbol with CC-A
Data: 012(00000)1239_15021231



UPC-E Composite symbol with CC-B
Data: 012(00000)1239_0123456789
0123456789ABCDEFGH

Appendix A

```
#include <stdio.h>
#include "lib.h"

#include "pdf_lib.h"           // Included header file the (Micro)PDF417 library

//Combines all allowed combinations of 2D-composite bar codes and linear bar codes
int composite_post_processing(struct barcode *code, struct barcode *prev_code,
                             unsigned long *time_out_value, short reset_time);

void main(void)
{
    int i;
    static char options[30 + 1];    //Holds string of menu-options

    static char bcr_buf[2000];      //Main bar code string buffer
    struct barcode code;             //Bar code structure for main bar code

    static char prev_buff[300];     //Storage for first read part of a linear/2D Composite symbol
    struct barcode prev_code;       //Bar code structure for first read part of a linear/2D Composite)

    unsigned long time_out_value;   //Holds system time before which 2nd part should be read

    init_PDF417_decoders();         //Initialize the following decoders to the Operating System
    // -> PDF417 (incl. its composite variant CC-C)
    // -> MICRO_PDF417 (incl. its composite variants CC-A & CC-B)

    strcpy(options, "JX");          //Enable RSS-14
    strcat(options, "JY");          //Enable RSS-Limited
    strcat(options, "DR");          //Enable RSS-Expanded

    strcat(options, "OG");          //Enable EAN128

    strcat(options, "[BCF");        //Enable PDF417 (incl. its composite variant CC-C)
    strcat(options, "[BCG");        //Enable MicroPDF (incl. its composite variants CC-A and CC-B)

    strcat(options, "[BHC");        //Enable transmit RSS link flag
    // '0' = no 2D-composite
    // '1' = CC-A or CC-B composite attached

    strcat(options, "[BI0");        //Transmit MicroPDF/CC-A/CC-B nr of data columns
    // '1' = MicroPDF symbol with 1 data column
    // '2' = MicroPDF or CC-A/B symbol with 2 data columns
    // '3' = MicroPDF or CC-A/B symbol with 3 data columns
    // '4' = MicroPDF or CC-A/B symbol with 4 data columns

    strcat(options, "OT");          //Transmit EAN128 link flag
    // '0' = no 2D-composite attached
    // '1' = CC-A or CC-B composite attached
    // '2' = CC-C composite attached

    systemsetting( options );       //Execute all menu-options shown above

    code.min    = 5;                //Minimum barcode length of 5
    code.max    = 2700;             //Maximum barcode length of 2700 (=max of PDF417)
    code.text   = bcr_buf;          //Pointer to result string buffer

    prev_code.min    = 5;           //Minimum barcode length of temporary stored composite component
    prev_code.max    = 400;         //Maximum barcode length of temporary stored composite component
    prev_code.text   = prev_buff;   //Pointer to result string of temporary stored composite component

    printf("\f\nPress TRIGGER\nto start\nscanning!  ");

    while(!kbhit()) idle();         //Wait untill key-pressed

    for(;;)
    {
        scannerpower( TRIGGER, 10000 ); //Set scanner in TRIGGER mode

        //Call readbarcode function and if try to combine a linear code to a 2D composite symbol
        while( readbarcode( &code ) != OK
            || composite_post_processing( &code, &prev_code, &time_out_value, 150) !=0)
        {
            if(!triggerpressed())    //If trigger is released
            {
                prev_code.id = 0;    //Clear ID of 1st part of linear/2D composite symbol
                scannerpower( TRIGGER, 10000 );
            }
            idle();
        }
    }
}
```



```

//Complete Barcode has been read

goodreadled(GREEN,10); //Beep en show Green LED on
sound(TSTANDARD,VHIGH,SMEDIUM,SHIGH,0);

scannerpower(OFF,0); //Turn of scanner

resetkey();
cursor( NOWRAP );

printf("\f"); //Clear screen
i=0;
while(i<code.length) //Print result
{
    printf("\n%-13.13s",&code.text[i]);
    i += 13;

    if(kbhit()) //Abort printing if key-pressed
        break;
}
}

/*****
Combines all allowed combinations of 2D-composite bar codes and linear bar codes

The following combinations of a linear bar codes with a 2D composite are allowed:
- RSS-14(standard) - CCA-COMPOSITE (4 data columns)
- RSS-14(standard) - CCB-COMPOSITE (4 data columns)

- RSS-14(stacked) - CCA-COMPOSITE (2 data columns)
- RSS-14(stacked) - CCB-COMPOSITE (2 data columns)

- RSS-LIMITED - CCA-COMPOSITE (3 data columns)
- RSS-LIMITED - CCB-COMPOSITE (3 data columns)

- RSS_EXPANDED - CCA-COMPOSITE (4 data columns)
- RSS_EXPANDED - CCB-COMPOSITE (4 data columns)

- EAN128 - CCA_COMPOSITE (4 data columns)
- EAN128 - CCB_COMPOSITE (4 data columns)
- EAN128 - CCC_COMPOSITE

Returns:
0 Normal barcode read OR both parts of a linear/2D composite combination read
1 First part of a linear/2D composite combination read
*****/
int composite_post_processing(struct barcode *code, struct barcode *prev_code,
                           unsigned long *time_out_value, short reset_time)
{
    char format_info,format_info_prev;

    switch(code->id)
    {
        case MICRO_PDF417: //Remove transmitted 'nr of data columns' char at first position
            memmove(code->text,&(code->text)[1],code->length);
            code->text[--code->length] = '\0';
            prev_code->id = 0;
            return 0; //Normal bar code, return OK
        case EAN128:
        case RSS_14:
        case RSS_LIMITED:
        case RSS_EXPANDED:
            if(code->text[0] == '0') //If no link flag set remove transmitted 'link-flag' at first pos.
            {
                memmove(code->text,&(code->text)[1],code->length);
                code->text[--code->length] = '\0';
                return 0; // Normal bar code, return OK
            }
        case CCA_COMPOSITE:
        case CCB_COMPOSITE:
            format_info = code->text[0]; //Store format info = link-flag(RSS) or nr-of-data-colums(CC-A/B)
            break;
        case CCC_COMPOSITE:
            break; //Note: CC-C composite symbols have no format info
        default:
            prev_code->id = 0;
            return 0; //Normal bar codes, return OK
    }

    if(code->length > prev_code->max) //Check maximum length for valid combinations
        return 0;

```

```

//If 1st half of a combination is already known, try to combine
if(prev_code->id != 0)
{
    if(GetSystemTime() < *time_out_value)
    {
        format_info_prev = prev_code->text[0];

        if (code->id == prev_code->id) //if same bar code ID
        {
            if(strcmp((char*)code->text, (char*)prev_code->text)==0) //if same data
            {
                *time_out_value = GetSystemTime() + reset_time; //time out value
                return 1; //First part read again!
            }

            prev_code->id = 0; //Store this part as 1st part
        }
        else if( code->id == CCA_COMPOSITE || code->id == CCB_COMPOSITE )
        {
            switch(prev_code->id) //Check if combination is invalid
            {
                case RSS_14:
                case RSS_LIMITED:
                case RSS_EXPANDED:
                case EAN128:
                    if(prev_code->id == RSS_14 && (format_info == '2' || format_info == '4'))
                        break;
                    if(prev_code->id == RSS_LIMITED && format_info == '3')
                        break;
                    if(prev_code->id == RSS_EXPANDED && format_info == '4')
                        break;
                    if(prev_code->id == EAN128 && format_info_prev == '1' && format_info == '4')
                        break;
                default:
                    prev_code->id = 0; //Store this part as 1st part
                    sound(TCLICK, VSTANDARD, SMEDIUM, 0);
            }
        }
        else if( code->id == CCC_COMPOSITE )
        {
            switch(prev_code->id) //Check if combination is invalid
            {
                case EAN128:
                    if(format_info_prev == '2') //Check link flag value
                        break;
                default:
                    prev_code->id = 0; //Store this part as 1st part
            }
        }
        else if (code->id >= RSS_14 && code->id <= RSS_EXPANDED)
        {
            switch(prev_code->id)
            {
                case CCA_COMPOSITE: //Check if combination is invalid
                case CCB_COMPOSITE:
                    if(code->id == RSS_14 && (format_info_prev == '2' || format_info_prev == '4'))
                        break;
                    if(code->id == RSS_LIMITED && format_info_prev == '3')
                        break;
                    if(code->id == RSS_EXPANDED && format_info_prev == '4')
                        break;
                default:
                    prev_code->id = 0; //Store this part as 1st part
                    sound(TCLICK, VSTANDARD, SMEDIUM, 0);
            }
        }
        else if( code->id == EAN128 )
        {
            switch(prev_code->id) //Check if combination is invalid
            {
                case CCA_COMPOSITE:
                case CCB_COMPOSITE:
                case CCC_COMPOSITE:
                    if((prev_code->id==CCA_COMPOSITE || prev_code->id==CCB_COMPOSITE)
                        && format_info == '1' && format_info_prev == '4')
                        break;
                    if(prev_code->id == CCC_COMPOSITE && format_info == '2')
                        break;
                default:
                    prev_code->id = 0; //Store this part as 1st part
                    sound(TCLICK, VSTANDARD, SMEDIUM, 0);
            }
        }
        else return 0;
    }
}

```

```

    else    // if Time out store this part as 1st part
    {
        prev_code->id = 0;
        sound(TCLICK,VSTANDARD,SMEDIUM,0);
    }
}
else
    sound(TCLICK,VSTANDARD,SMEDIUM,0);

if(prev_code->id != 0)    // if 2 parts read, try to combine
{
    if(code->length + prev_code->length - 2 > code->max)        // too large combination
        return 0;

    if(code->id != CCC_COMPOSITE)    //If not CC-C remove link flag/nr of columns of 2nd part
    {
        memmove(code->text,&(code->text)[1],code->length);
        code->text[--code->length] = '\0';
    }

    if(prev_code->id != CCC_COMPOSITE) //If not CC-C remove link flag/nr of columns of 1st part
    {
        memmove(prev_code->text,&(prev_code->text)[1],prev_code->length);
        prev_code->text[--prev_code->length] = '\0';
    }

    //Put RSS and EAN128 Codes in front and 2D (Micro)PDF417 variants behind it
    if((code->id >= RSS_14 && code->id <= RSS_EXPANDED) || code->id == EAN128)
    {
        strcpy(&(code->text)[code->length+1],prev_code->text);
        (code->text)[code->length] = GS;    //Add Primary/Secondary data seperation char.
    }
    else
    {
        memmove(&(code->text)[prev_code->length+1],code->text,code->length+1);
        memcpy(code->text,prev_code->text,prev_code->length);

        (code->text)[prev_code->length] = GS;    //Add Primary/Secondary data seperation char.

        code->id = prev_code->id;    //Change ID to ID of linear part
    }

    code->length += prev_code->length;    //Set combined length

    prev_code->id = 0;    //Clear stored ID

    return 0;    //Combining of a linear code with a 2D composite code -> succesfull
}

prev_code->id = code->id;    //Store ID of 1st part
prev_code->length = code->length;    //Store length of 1st part
strcpy(prev_code->text,code->text);    //Store 1st part in prev_code
*time_out_value = GetSystemTime() + reset_time; //Set time out value

return 1;    // Return 'Fist part of EAN*UCC Composite Symbol stored'
}

```