

COLOR CUTOUCH

CT2400 시리즈

사용자 설명서

Version : 1.0

COMFILE
TECHNOLOGY

컴파일 테크놀로지 주식회사
www.comfile.co.kr

등록상표

WINDOWS 는 Microsoft Corporation 의 등록상표입니다.
CUBLOC, CUTOUCH 는 Comfile Technology 의 등록상표입니다.
기타 다른 상표는 해당회사의 등록상표입니다.

알림

본 설명서의 내용은 사전 통보 없이 변경될 수 있습니다. 본 제품의 기능은 성능 개선을 위하여 사전 통보 없이 변경될 수 있습니다. 본 제품을 이 자료에서 설명한 용도 외에서 사용할 경우, 폐사에서는 어떠한 책임도 지지 않으므로 주의하시기 바랍니다. 본 제품은 컴파일 테크놀로지의 고유 기술을 사용하여 개발된 제품으로 저작권법에 의한 보호를 받고 있습니다. 따라서 본 제품 (제품에 대한 아이디어 및 설명서 및 기타 포함)의 어떠한 부분도 사전에 폐사와의 문서 동의 없이 복사되거나 변경, 재 생산할 수 없으며 또한 다른 언어로도 번역될 수 없습니다.

주의사항

인쇄된 설명서는 인쇄된 시점에서는 최신 버전이지만, 인쇄된 후 시간이 경과된 뒤에 새로운 내용이 추가되거나, 기존 내용이 바뀔 가능성이 있습니다. 최신 버전의 설명서는 항상 인터넷 홈페이지 (www.comfile.co.kr)에서 확인하시기 바랍니다.

본 제품을 사용하시다가 생긴 손해 및 손실에 대하여 저희 컴파일 테크놀로지 주식회사는 어떠한 책임도 없음을 명시하는 바입니다. 본 사용설명서 및 제품은 저작권법에 의해 보호받고 있으므로 무단복제시 법에 의한 처벌을 받게 됩니다.

인쇄 내역

2008 년 09 월 배포함
Copyright 1996,2008 컴파일 테크놀로지(주)

OVERVIEW

CUTOUCH 17XX 시리즈에 이어서, 칼라 TFT 스크린과 다양한 기능을 갖춘 CT2400 시리즈를 출시하게 되었습니다.

기존 CT17XX 시리즈에서 미흡했던 부분을 대폭 보강하고, 최근 추세에 맞게 COLOR 스크린을 갖춘 터치콘트롤러로 제작되었습니다. 뿐만 아니라 기본 CORE 모듈을 CB405 급으로 채택함으로써, 기존 제품보다 많은 메모리 용량을 지원하게 되었습니다.

다음은 두 제품군간의 차이점을 표로 정리해본 것입니다.

	CT1721	CT2400
프로그램 메모리	80K BYTE	200K BYTE
데이터 메모리	24K BYTE	110K BYTE
화면크기	5.7 인치	7 인치 WIDE
해상도	320 X 240	800 X 480
디스플레이 모듈	GHLCD	INTELLILCD
CORE 모듈	CB290	CB405
I/O 수	80 여개	40 여개
RS232	1 채널	2 채널
사운드지원	안함	함

본 사용설명서에는 “BASIC 문법” 과 “레더로직”에 관련된 부분이 포함되어 있지 않습니다. BASIC 문법 및 명령어는 “큐블록 BASIC 중심 사용설명서 ”를 참조하시고, LADDER LOGIC 은 “큐블록 레더중심 사용설명서”를 참조하시기 바랍니다.

컴파일 테크놀로지㈜

차 례

제 1 장 CT2400의 기초.....	5
CUBLOC 소개.....	6
INTELLLCD 소개	7
개발환경	7
CUBLOC STUDIO.....	8
CUWIN 과 CUTOUCH	9
CT2400의 외관.....	10
CT2400의 외형치수.....	11
제 2 장 입출력 회로연결	13
CT2400 기본 I/O 보드.....	14
기본 I/O 보드 입출력사양.....	15
RS232/485 포트	18
DIP 스위치 설정.....	19
기타연결	20
TIPS : CUWIN 모드.....	21
터치보정.....	22
제 2 장 디스플레이와 터치입력.....	25
디스플레이 명령.....	26
샘플 프로그램 1: 화면표시	27
샘플 프로그램 2: 터치입력.....	31
샘플 프로그램 3: 콘트롤 표시와 입력.....	35
샘플 프로그램 4: 스크린 기능 활용.....	40
제 4 장 I/O 콘트롤.....	55
써모커플 입력.....	56
제 5 장 트러블슈팅.....	59
다운로드 케이블 연결.....	60
다운로드 시 발생하는 에러메시지.....	64
부 록.....	65
ACTIVESYNC 설치하기.....	66
ACTIVESYNC 실행.....	68

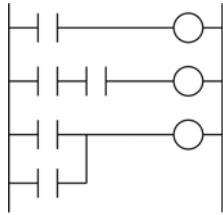
제 1 장 CT2400 의 기초

CUBLOC 소개

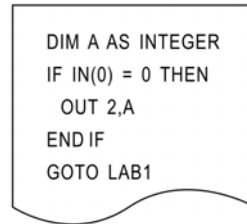
CT2400 시리즈는 CUBLOC 을 기반으로 하고 있는 비주얼 콘트롤러입니다. CUBLOC 은 저회 컴파일 테크놀로지에서 개발한 “반도체형 임베디드 컴퓨터”입니다. 여러가지 모델이 있지만 이중 CB405 라는 제품을 CT2400 에 내장하였습니다.



CUBLOC 은 BASIC 언어와 LADDER LOGIC 을 동시에 사용할 수 있습니다. BASIC 언어로는 화면표시 유저인터페이스 처리등을 담당하고, LADDER LOGIC 으로는 I/O 제어 및 시퀀스제어를 담당합니다.



LADDER LOGIC

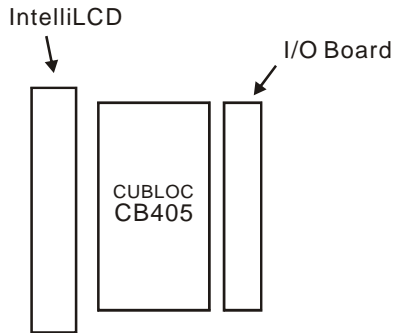


BASIC

IntelliLCD 소개

CT2400 시리즈는 CUBLOC 코어모듈과 IntelliLCD 를 합친 제품이라고 할 수 있습니다. IntelliLCD 는 컴파일테 크놀로지에서 개발한 칼라 디스플레이 엔진입니다. 간단한 명령어로 칼라 LCD 상에 선,원,문자등을 표시할 수 있고, 그외에도 다양한 기능을 갖춘 제품입니다.

CB405 와 IntelliLCD, 이 두개의 제품을 하나의 케이스안에 합침으로써, 유저여러분께서 현장에 바로 응용할 수 있도록 만든 제품입니다.



개발환경

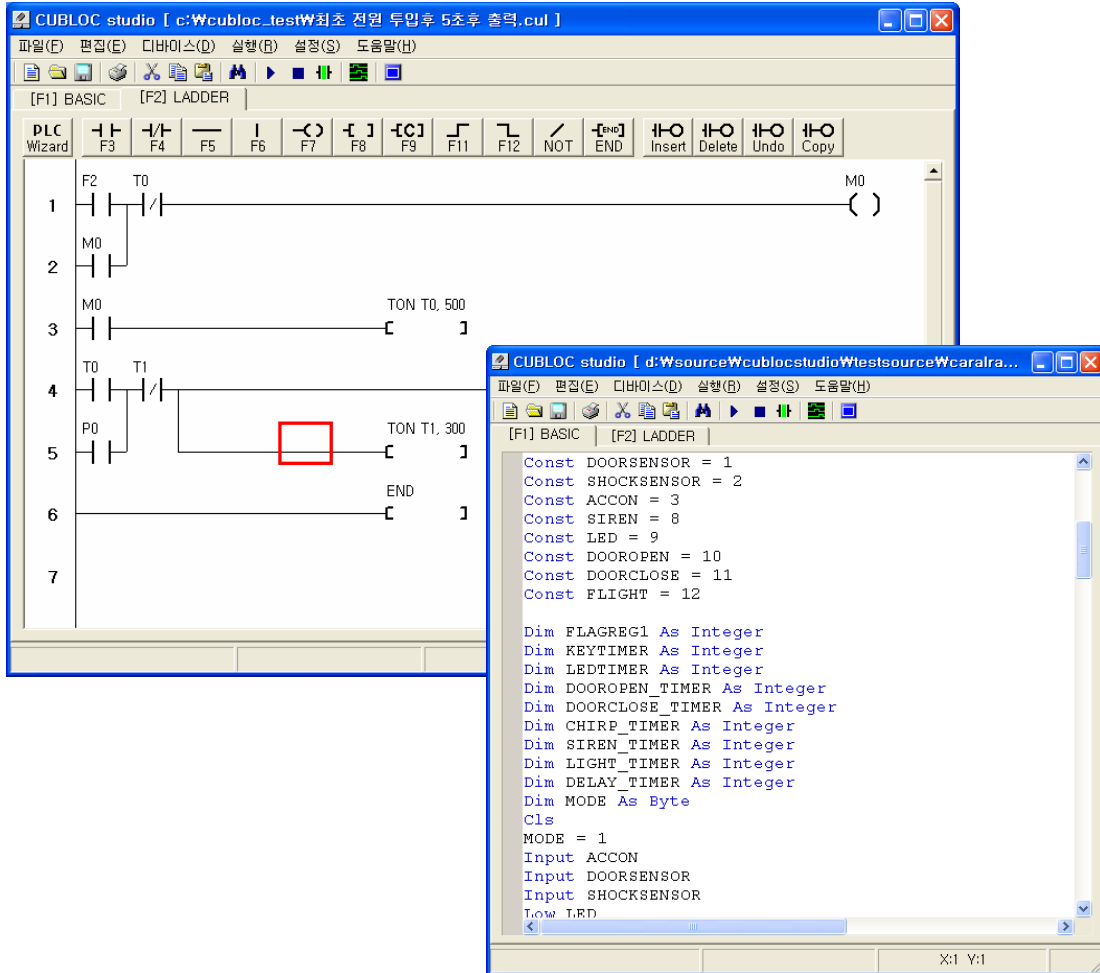
CT2400 을 사용하기 위해서는 Windows XP(2000, VISTA 이상) 운영체제와 RS232 포트를 갖춘 PC 가 필요하고, PC 와 RS232 로 접속합니다. RS232C 포트가 없는 PC 라면, USB to RS232C 변환 컨버터를 사용합니다. PC 와 연결된 상태에서 다운로드 및 모니터링이 가능합니다.

PC 와 연결을 해제하면 혼자서 동작하는 STAND ALONE 상태가 됩니다. CT2400 에는 플래쉬 메모리가 내장되어 있어, 한번 기억된 프로그램은 전원이 없어도 계속 기억하고 있습니다. 나중에 다시 새로운 프로그램을 다운로드 할 수 있으며, 프로그램을 지울 수도 있습니다.

* 노트북 사용자에게는 USB-TO-RS232 보다 PCMCIA-TO-RS232 사용을 권장합니다. PCMCIA 카드로 되어 있어 RS232 포트가 CPU 와 직접 접속되는 방식입니다. 따라서 에러율이 적고, 속도가 빠릅니다. (권장 제품 : 강원전자 의 NETMATE C-191)

CUBLOC STUDIO

CT2400 을 사용하기 위해서는 통합개발환경 소프트웨어인 CUBLOC-STUDIO 가 필요합니다 .이 프로그램은 은 인터넷 홈페이지 www.comfile.co.kr을 통하여 무상 제공됩니다. 유저는 이 프로그램을 사용해서 BASIC 과 LADDER 프로그램을 작성하게 됩니다.



CUBLOC STUDIO 사용법은 별도 매뉴얼인 “CUBLOC BASIC 중심 사용설명서”를 참조하시기 바랍니다.

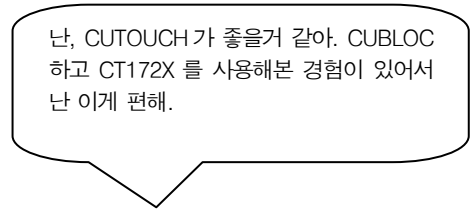
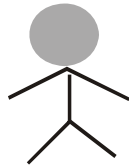
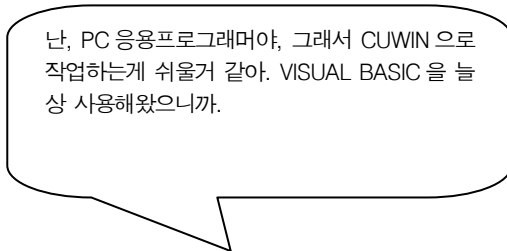
CUWIN 과 CUTOUCH

저희 회사에서 출시된 또하나의 터치컨트롤러 제품인 CUWIN 시리즈가 있습니다. CUTOUCH 제품군과 외관과 기능이 비슷해 보이지만 차이가 있는 제품입니다. 아래 표로 요약해 보았습니다.

	CUWIN	CUTOUCH
개발환경	마이크로 소프트사 개발환경 VISUAL STUDIO (유상)	컴파일사의 개발환경 CUBLOC STUDIO (무상)
사용언어	VISUAL BASIC, 또는 C#	CUBLOC BASIC 과 LADDER
OS	WINDOWS CE	CUBLOC OS
다운로드포트	USB	RS232

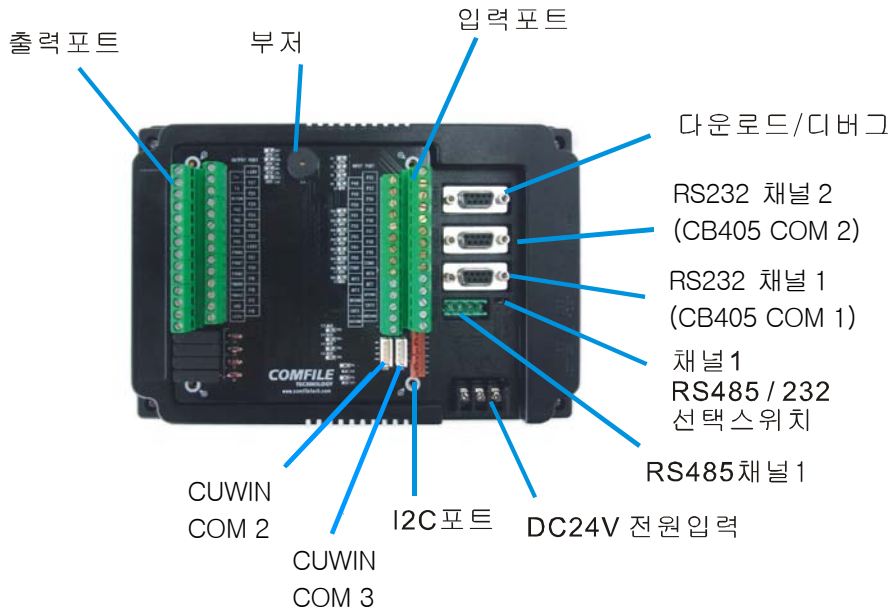
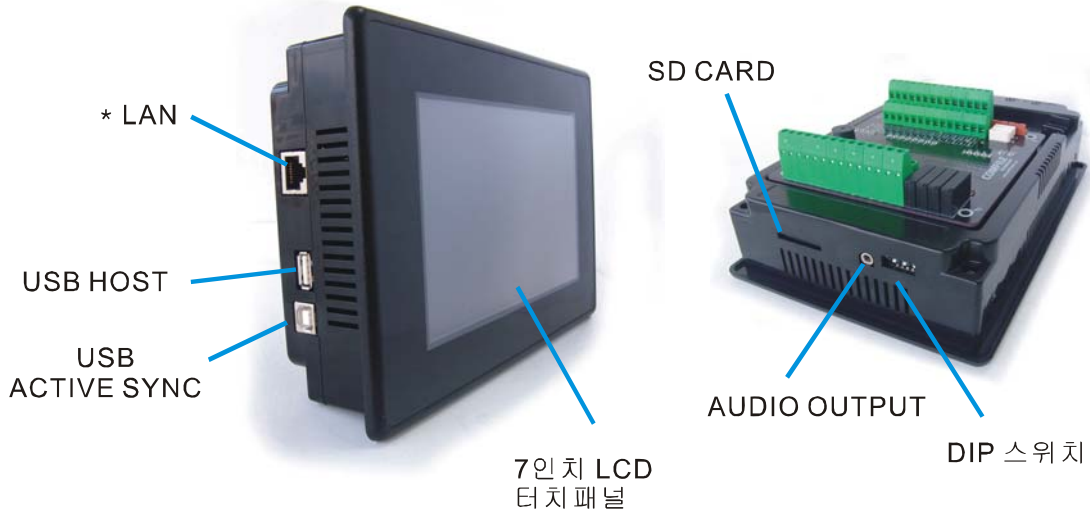
표로 알 수 있듯이 CUWIN 은 마이크로 소프트사의 OS 와 개발환경을 사용하고, CUTOUCH 는 컴파일테크놀로지사에서 만든 CUBLOC OS 와 개발환경을 사용합니다. 두 제품은 같은 터치컨트롤러 이지만 개발환경이 확연하게 차이므로, 선택시 이점을 고려해야합니다.

만약 당신이 PC 환경에서 VISUAL BASIC, C#등에 익숙한 유저라면 CUWIN 을 선택하시는 것이 좋습니다. 그렇지 않고, CUBLOC, CT1721 등을 사용해본 유저라면 CT2400 시리즈가 더 좋을 것입니다.

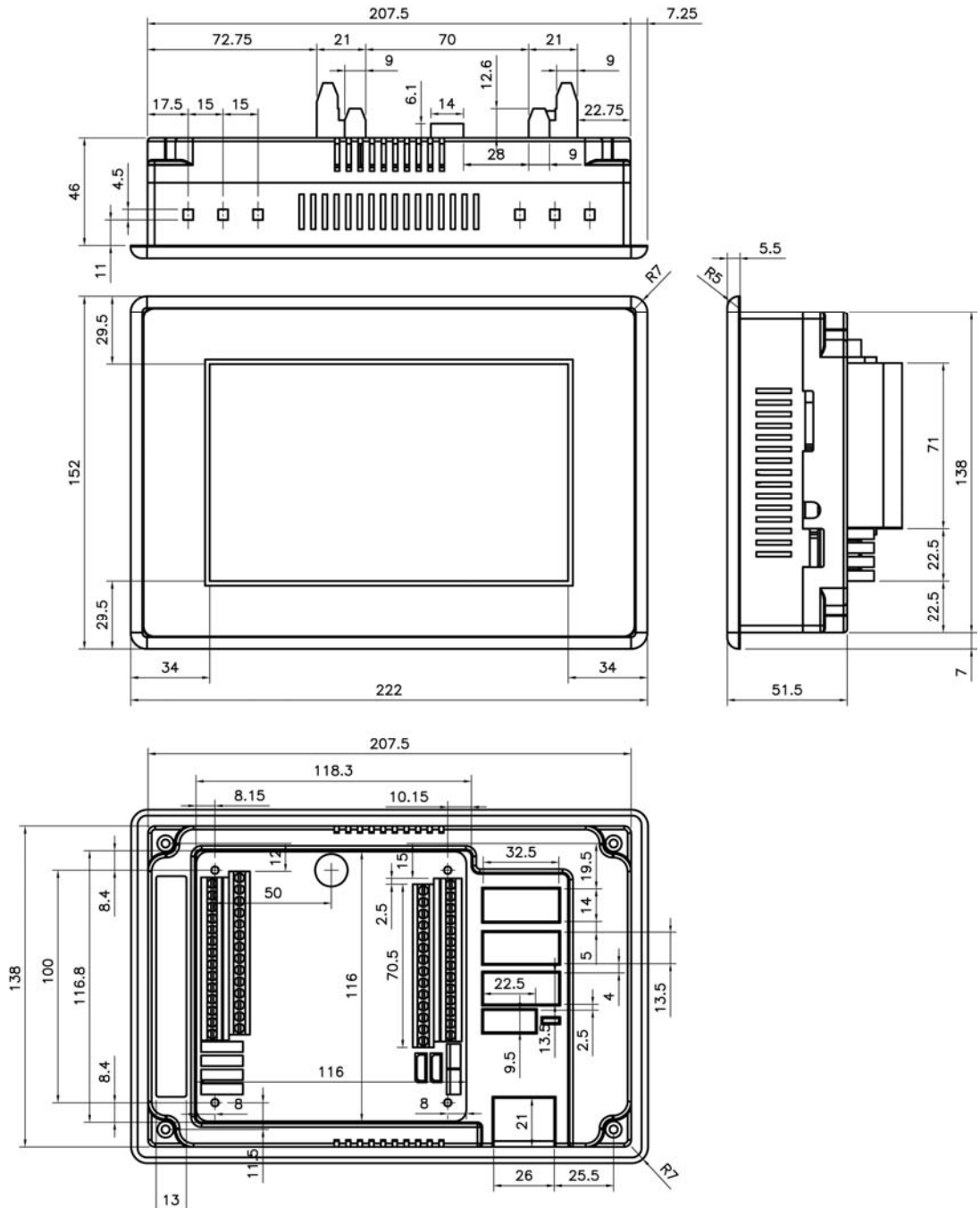


CT2400 의 외관

CT2400 에는 기본 I/O 보드가 장착되어 있습니다. (입력 20, 출력 16 개)



CT2400 의 외형치수

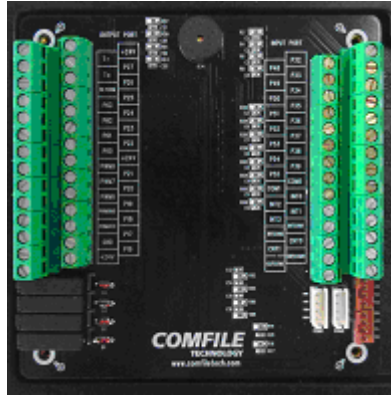


제 2 장

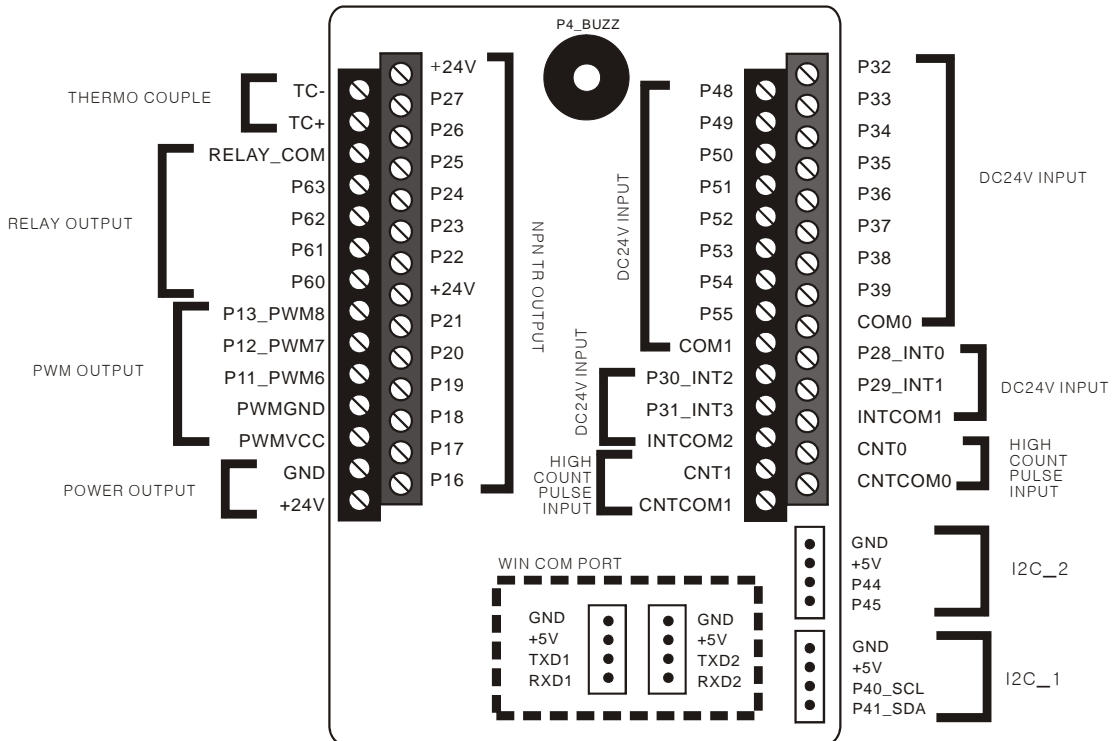
입출력 회로연결

CT2400 기본 I/O 보드

CT2400 후면의 기본 I/O 보드에 대한 상세설명입니다.



이 보드의 구성은 다음과 같습니다.



기본 I/O 보드 입출력사양

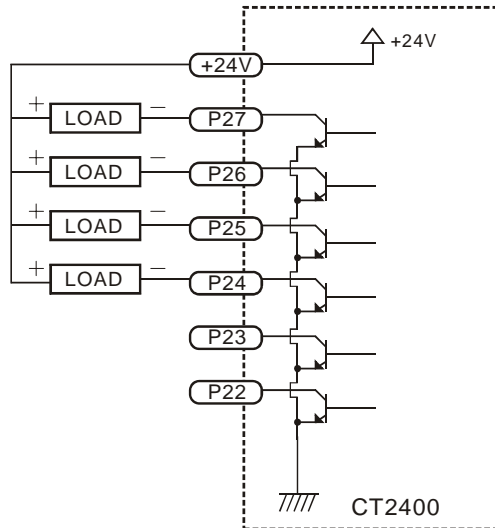
CT2400 내부의 CB405 포트중 일부는 입력, 일부는 출력으로 연결해 놓았습니다. 구체적인 입출력 사양은 다음과 같습니다.

	포트할당	부연설명
DC24V 입력	P32 부터 P39 P48 부터 P55	총 16 개
DC24V 입력 (INT 입력)	P28, P29, P30, P31	총 4 개
NPN TR 출력	P16 부터 P27	총 12 개
RELAY 출력	P60, P61, P62, P63	총 4 개
PWM 출력 (NPN TR 출력)	P11, P12, P13 (PWM 6, 7, 8)	총 3 개
고속카운터 입력 (DC24V)	P46, P47	총 2 개

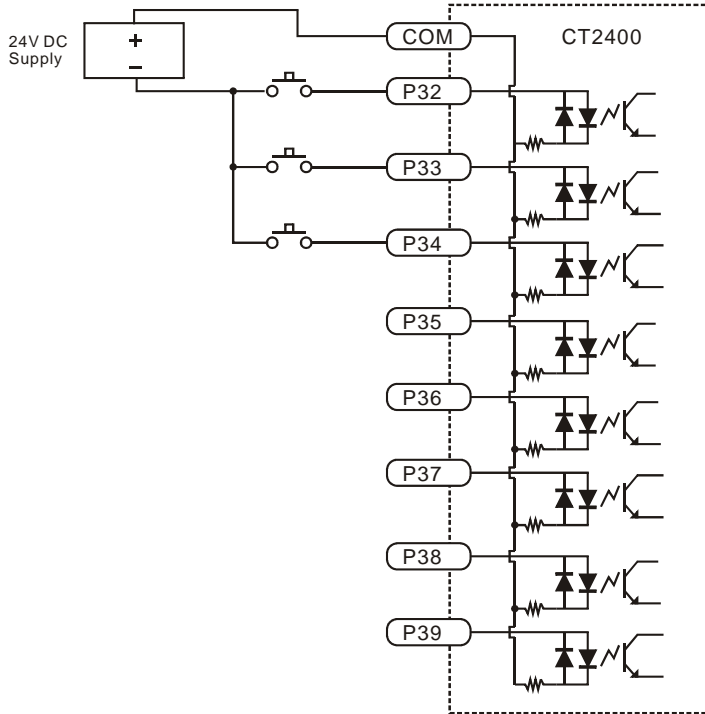
정리하면 입력 20 개, 출력 16 개, PWM 3 개, HIGH COUNTER 2 개입니다. 이외에도 I2C 연결을 위한 포트 2 개가 있습니다. 향후 주변장치 연결용으로 사용될 예정입니다.

WIN COM PORT 는 CT2400 에서는 사용하지 않는 포트입니다. 이 포트는 INTELLILCD 의 COM PORT 중 일부입니다.

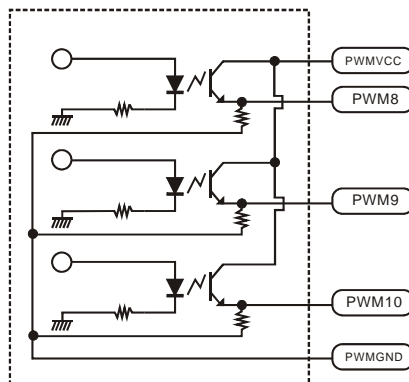
출력포트는 NPN TR OUTPUT 으로 구성되어 있습니다. 예를 들어 CB405 의 포트를 HIGH 로 만들면, NPN TR 이 턴온되어 LOAD (부하)에 24V 가 걸리게 됩니다.



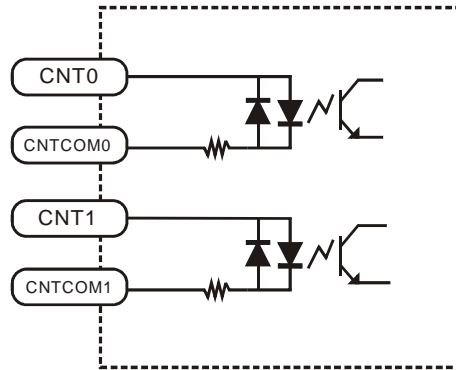
입력포트는 다음 그림과 같이 양방향 DC24V 입력 신호를 받도록 되어 있습니다. 해당 터미널 블록에 DC24V 를 공급해주면, 내부에 있는 포토커플러 LED가 점등하게 되고, 이 신호가 결국 CB405의 해당 I/O 포트에 HIGH 신호로 전달됩니다. 외부 DC24V를 사용해도 무방합니다.



PWM 포트는 다음그림과 같이 NPN TR OUTPUT 으로 되어 있습니다. PWMVCC 에 24V 에 연결하고, PWMGND 를 24V 의 그라운드에 연결한뒤 사용할 수 있습니다. 이렇게 하면 24V 의 PWM 파형이 출력됩니다. PWMVCC 와 PWMGND 를 별도 입력함으로써, PWM 출력전압을 변경하실 수 있습니다. (5V~ 30V 가지 사용가능)

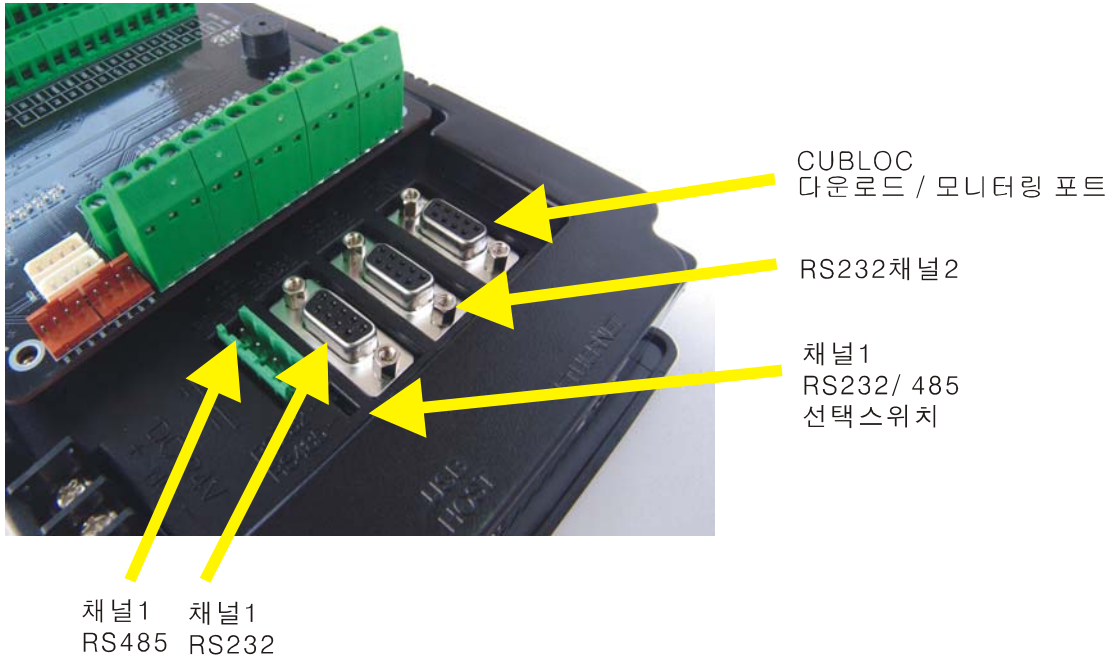


고속펄스 입력포트는 다음과같이 DC24V 입력으로 되어 있습니다.



RS232 / 485 포트

CT2400에는 총 3개의 RS232와 1개의 RS485 포트가 내장되어 있습니다. 이중 1개는 다운로드 / 모니터링을 위한 RS232 포트이며, 나머지 2개는 각각 채널 1, 채널 2와 연결되어 있습니다.



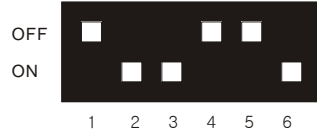
다운로드 포트를 PC와 연결하여, CB405에 프로그램을 다운로드 및 디버깅을 수행합니다.

채널 1 포트는 선택스위치에 의해서 RS232로 사용할 수도 있고, RS485로 사용할 수도 있습니다. RS485 사용시 내부적으로 송신허가포트로 P10이 연결되어 있습니다.

내부적으로 CB405의 채널 3번은 IntelliLCD와 연결되어 있으므로, 유저는 채널 3을 다른 용도로 사용할 수 없습니다.

DIP 스위치 설정

CT2400의 측면에 DIP 스위치가 있습니다. 출하시 아래 그림과 같이 설정되어 있습니다. 만약 아래 그림처럼 설정되어 있지 않다면, 동작을 보증할 수 없으니 주의바랍니다.



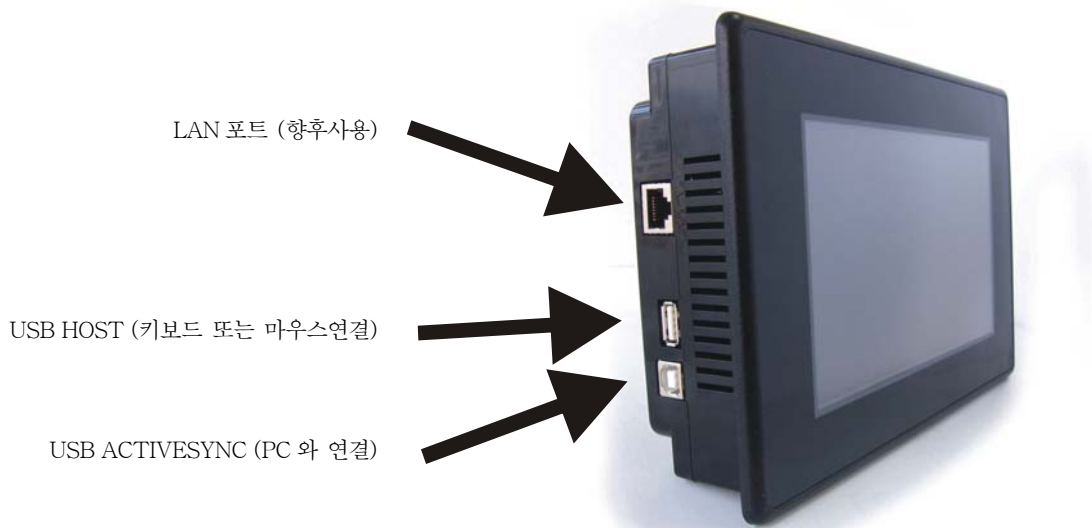
번호	스위치 1	스위치 2	스위치 3	스위치 4	스위치 5	스위치 6
상태	OFF	ON	ON	OFF	OFF	ON
기능	없음	부팅모드 1	부팅모드 2	드레그기능	통신속도 1	통신속도 2

부팅모드 1	부팅모드 2	부팅
ON	ON	INTELLILCD 모드 (디폴트 상태)
ON	OFF	APPLICATION 모드 (사용안함)
OFF	OFF	CUWIN 모드 (사용안함)

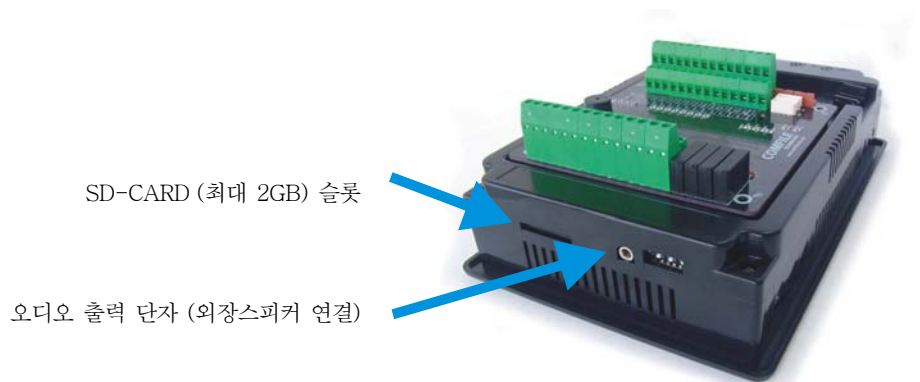
통신속도 1	통신속도 2	통신속도
ON	ON	9600
ON	OFF	19200
OFF	ON	38400 (디폴트 상태)
OFF	OFF	115200

기타연결

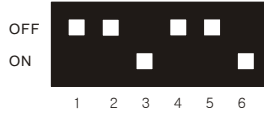
그외에도 USB, SD-CARD, AUDIO OUT 단자등이 있습니다.



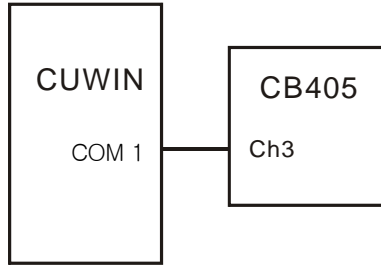
USB ACTIVE SYNC 는 PC 와 연결하여 INTELLILCD 자료를 다운로드 받을 때 사용하는 포트입니다. 자세한 내용은 INTELLILCD 사용설명서를 참조하세요.



TIPS : CUWIN 모드

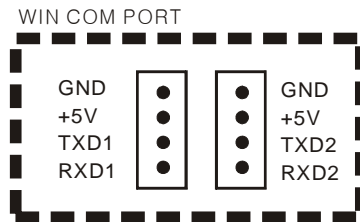


DIP 스위치의 부팅모드를 CUWIN 모드로 세팅해놓고, 전원을 재 인가한다면 WINDOWS CE 상태에서 시작합니다. 이 경우 CUWIN + CB405 조합이 됩니다 .



이 상태에서는 WINDOW CE 의 COM3 포트가 CB405 의 RS232 채널 3 에 연결되어 있습니다. 두 프로세서간의 통신을 하시려면 이 채널을 통해서 할 수 있습니다. (현재 CB405 채널 3 에는 MODBUS 슬레이브 기능이 없으므로, 유저가 직접 통신프로그램을 작성해 주셔야 합니다. 향후 CB405 채널 3 에 MODBUS 슬레이브 기능을 추가 예정입니다.)

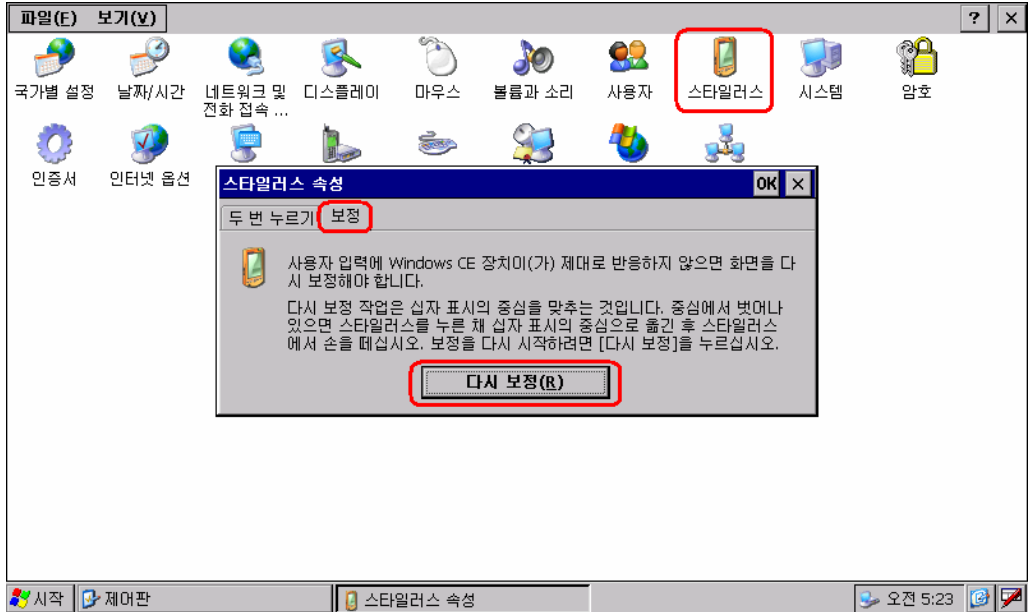
CUWIN 모드에서 I/O 보드상에 있는 WINCOM 포트를 사용할 수 있습니다. 이 포트에서는 +/- 12V 의 RS232 신호를 연결하십시오. (5V 레벨의 UART 신호를 직접 연결하지 마십시오.) 또한 CUWIN 모드에서 LAN 포트를 사용하는 것도 가능합니다.



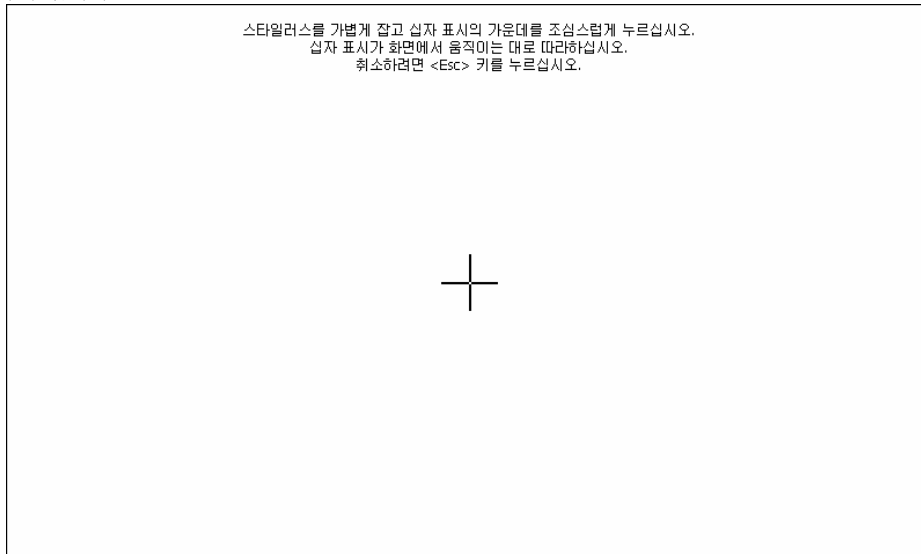
터치보정

CUWIN 모드에서 터치보정을 할 수 있습니다. DIP 스위치를 CUWIN 모드로 설정하고 전원을 재인가합니다.

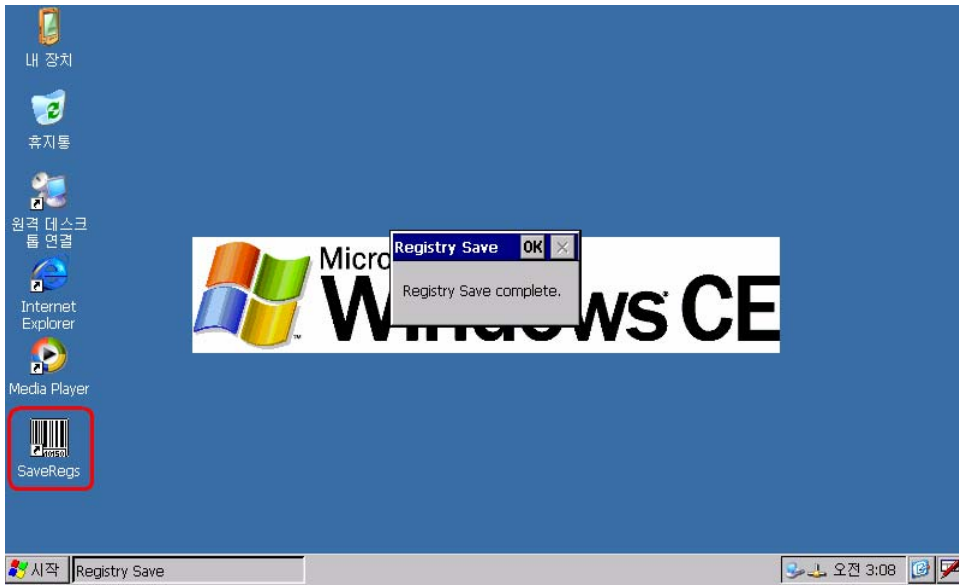
설정의 제어판에 스타일러스를 더블 클릭합니다



스타일러스 속성창의 보정 탭을 선택합니다. 다시 보정 버튼을 누르면 보정작업을 진행합니다. 순차적으로 5 점을 입력 할 것을 나타냅니다. 출력메시지에 따라서 순서대로 입력합니다. 만약 입력 시 잘못 입력될 경우 처음부터 다시 입력하게 됩니다.



터치 보정이 끝나면 레지스트리 저장작업을 반드시 해야 합니다.



제 2 장

디스플레이와

터치입력

디스플레이 명령

CUTOUCH 2400 시리즈에는 INTELLILCD 의 그래픽 명령어를 그대로 사용합니다. 명령어 사용형식은 다음과 같습니다.

@ “LINE 100,100,200,300;”

맨 앞에 @ 는 IntelliLCD 용 코멘트라는 뜻입니다. 내부적으로 “Putstr 채널번호, ”로 번역됩니다. 즉, 시리얼 포트를 통해 IntelliLCD 로 코멘트를 전송합니다.

이 명령어를 사용하기 위해선, 프로그램 맨 앞줄에 Set Ilcd 3 이라는 명령어를 써주어야 합니다. CT2400 에 내장된 CB405 의 채널 3 번이 IntelliLCD 와 연결되어 있습니다.

Set Ilcd 3

다음 소스 프로그램은 CT2400 을 사용하기 위해서 선두에 작성해주어야 하는 소스입니다. 디바이스정의와 RS232 포트정의 화면클리어등의 문장을 포함하고 있습니다.

```
Const Device = CB405

Set ilcd 3
Opencom 3,38400,3,80,180
Wait 200
@ "#@#@;Clear;"
```

* 본 사용설명서에는 IntelliLCD 명령어에 대한 구체적인 설명이 포함되어 있지 않습니다. 별도 매뉴얼인 “IntelliLCD 사용설명서”를 참조해 주시기 바랍니다.

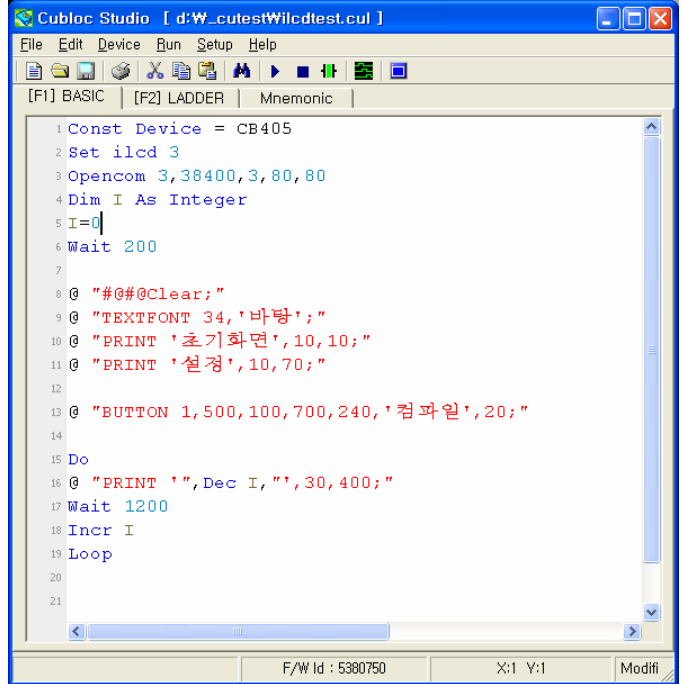
샘플 프로그램 1 : 화면표시

다음은 간단한 예제 프로그램들입니다. 먼저 CT2400 화면상에 글자와 버튼을 표시해보도록 하겠습니다.

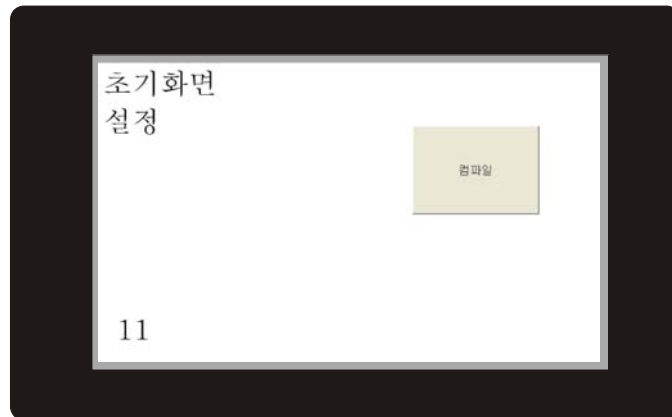
다음 프로그램을 입력한뒤 실행하시면 CT2400 화면상에 아래와 같은 그림이 표시됩니다.

소스 파일명 <ILCDTEST.CUL>

본 사용설명서에 있는 소스파일은 comfile.co.kr 의 자료실에 공개해 놓았습니다. 직접 실행해보고 싶으신 분들은 다운로드 받은 후 사용하세요.



```
1 Const Device = CB405
2 Set ilcd 3
3 Opencom 3,38400,3,80,80
4 Dim I As Integer
5 I=0
6 Wait 200
7
8 @ "#@#@Clear;"
9 @ "TEXTFONT 34,'바탕';"
10 @ "PRINT '초기화면',10,10;"
11 @ "PRINT '설정',10,70;"
12
13 @ "BUTTON 1,500,100,700,240,'컴파일',20;"
14
15 Do
16 @ "PRINT ' ',Dec I,",' ',30,400;"
17 Wait 1200
18 Incr I
19 Loop
20
21
```



소스분석

이 소스 프로그램을 한줄 한줄 살펴보도록 하겠습니다.

```
Const Device = CB405
```

사용 디바이스를 정해주는 명령입니다. 앞서도 언급하였듯이 CT2400 에는 CB405 가 내장되어 있으므로, CONST DEVICE 문에는 CB405 를 선언하여 줍니다.

```
Set ilcd 3
```

CT2400 에 채널 3 에 INTELLILCD 가 접속되어 있습니다. 따라서 INTELLILCD 코멘드 출력 채널은 3 번입니다. 초기에 SET ILCD 를 3 으로 해놓으면 이후 “@” 명령어에서 출력포트는 모두 채널 3 이 됩니다.

```
Opencom 3,38400,3,80,80
```

RS232 통신을 개시하는 오픈명령어 입니다. 채널 3 을 오픈하고, 보레이트는 38400 으로 셋팅합니다. 8 비트 /NONE 패리티/1 스톱비트로 (설정값은 3)하고, 입력 버퍼와 출력버퍼수를 각각 80 바이트로 선언합니다.

```
Dim I As Integer  
I=0
```

변수 I 를 선언하고 초기값을 0 으로 만듭니다.

```
Wait 200
```

초기에 200 밀리초를 딜레이 해줍니다. 나중에 이 부분을 20 초 이상 딜레이 해주도록 바꾸어야 합니다. 초기 전 원 인가시 INTELLILCD 가 부팅이 끝나고 안정되기까지 시간을 기다려 주어야 하기 때문입니다. CUBLOC 은 전 원이 인가되지 마자 짧은 시간 (100 밀리초)에 부팅이 끝나지만, INTELLILCD 는 20 초이상 걸립니다. (최초 부팅 시에는 WAIT 20000 으로 수정)

하지만 계속 전원이 들어가 있는 상태에서 CUBLOC 만 다운로드만 하는 상황이라면 구지 20 초 이상을 기다릴 필요가 없으므로 200 밀리초만 딜레이하도록 한것입니다.

```
@ "#@#@Clear;"
```

최초에 등장하는 IntelliLCD 명령어입니다. “#@#@”는 전 버퍼나 대기열에 있던 내용을 모두 삭제하는 명령어 입니다. 인텔리 LCD 가 어떤 실행하고 있을 경우 이것을 모두 무시하기 위한 명령어라고 보면됩니다. 그뒤에 Clear 를 보내 화면을 모두 지웁니다. (흰색으로 채웁니다.)

```
@ "TEXTFONT 34, '바탕' ;"
```

인텔리 LCD 로 보내는 명령어는 맨앞에 @를 써주고 따옴표로 감싸주어야 합니다. TEXTFONT 는 화면에 표시할 글꼴을 선택하는 명령어입니다. 이 명령어에서는 바탕체, 34 포인트로 설정하였습니다. 이 명령어는 실행결과가 화면상에 표시되지는 않습니다. 다만 앞으로 표시할 모든 글자를 “바탕체, 크기는 34 포인트 “로 표시할 것입니다.

```
@ "PRINT '초기화면',10,10;"  
@ "PRINT '설정',10,70;"
```

PRINT 는 화면에 글자를 표시하는 명령어 입니다. 뒤에 좌표 (X, Y 순)을 써줍니다.

```
@ "BUTTON 1,500,100,700,240,'컴파일',20;"
```

화면에 “버튼 컨트롤” 을 표시하는 명령어 입니다.

파라미터는 ID, X1, Y1, X2, Y2, 제목, 채팅딜레이 순입니다. ID 는 이름표와 같습니다. 나중에 이 버튼을 누르면 이 ID 값을 리턴해줍니다. X1,Y1,X2,Y2 좌표를 대각선으로 하는 사각형의 버튼이 표시됩니다. 그리고 그 가운데에 “제목”이 표시됩니다. 채팅딜레이는 이 버튼을 아주 살짝 눌렀을 때는 눌러지지 않도록 하기 위한 수치입니다.

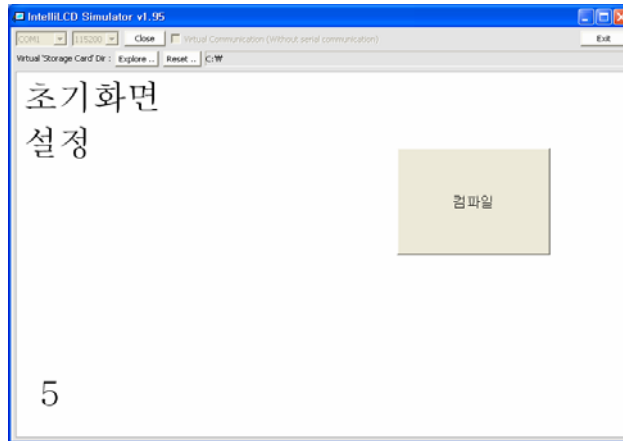
```
Do  
@ "PRINT '",Dec I,"',30,400;"  
Wait 1200  
Incr I  
Loop
```

루프입니다. 1.2 초 간격으로 화면하단에 숫자를 표시합니다. 변수 I 의 값을 증가시키면서 표시하는 것을 알 수 있습니다.

TIPS : IntelliLCD 시뮬레이터 사용

같은 프로그램을 PC 상에서 실행시켜서 결과를 볼 수 있습니다. 위의 소스중 맨앞에 있는 SET ILCD 3 을 SET ILCD 0 으로 바꾸고 OPENCOM 을 막아준다면 (주석처리), CUBLOC STUDIO 에 포함된 IntelliLCD 시뮬레이터 가 실행됩니다. (프로그램을 다운로드해야되기 때문에 CB405 모듈이 연결되어 있어야 실행됩니다.) 이 방법은 CT2400 이 없어도 CB405 만 가지고 CT2400 용 프로그램을 미리 확인해볼 수 있는 방법입니다.

```
Set ilcd 0
'Opencom 0,38400,3,80,80
```



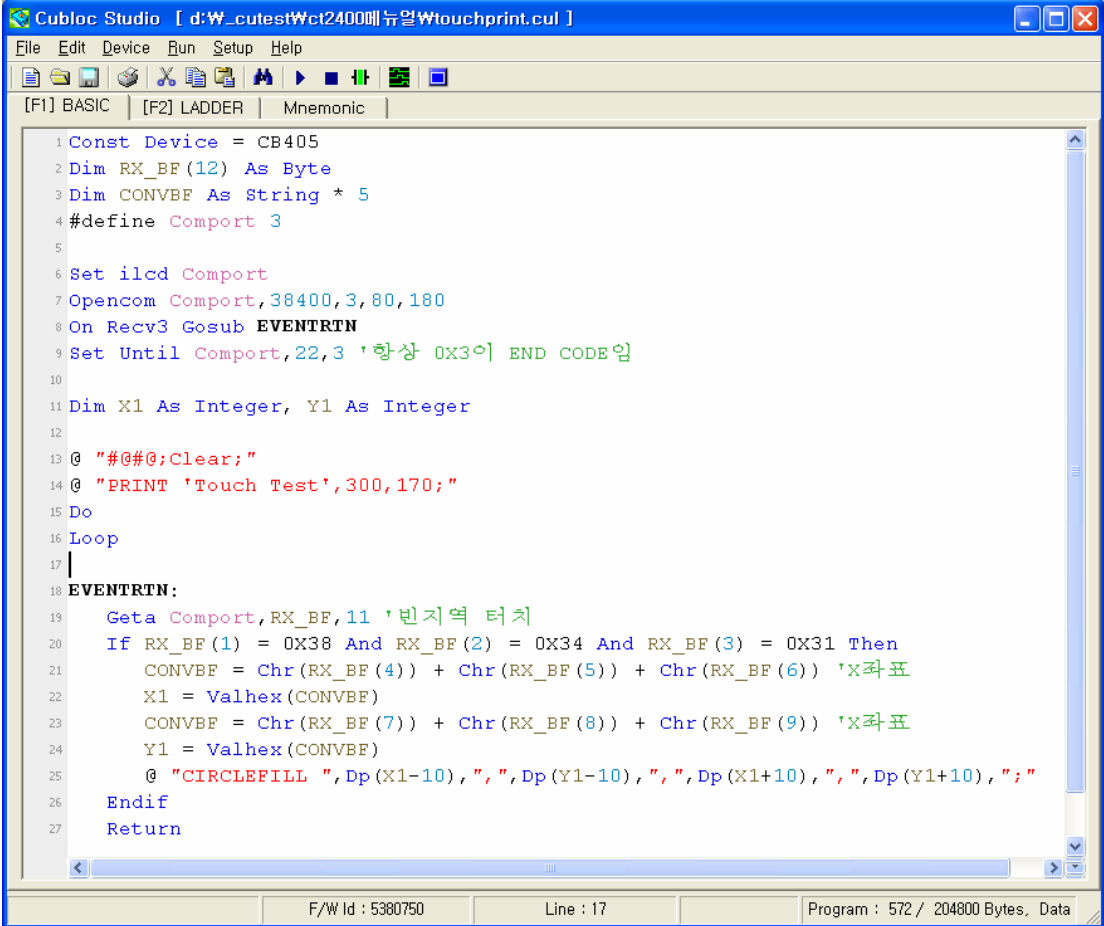
OPENCOM 을 막아주는 이유는 채널 0 의 경우 이미 OPEN 되어 있는 상태로 시작되기 때문입니다. 인텔리 LCD 시뮬레이터를 사용할 때에는 DEBUG 명령어도 사용하지 못합니다. 같은 채널 0 를 사용하기 때문입니다. 만약 사용중인 DEBUG 명령어가 있다면 SET DEBUG OFF 를 프로그램 맨 첫줄에 써주면 됩니다. (모든 DEBUG 명령어 사용을 금지시켜줍니다.)

```
Set Debug Off
```

* 인텔리 LCD 시뮬레이터 실행결과와 CT2400 의 실행결과는 100% 일치하지 않습니다. 이점을 사전에 염두에 두시고 시뮬레이터를 이용하시기 바랍니다. 시뮬레이터는 여러분의 PC 상에서 동작하기 때문에, 실행속도가 CT2400 보다 빠른편입니다.

샘플 프로그램 2 : 터치입력

이번 프로그램은 터치입력 받은 지점에 작은 원을 표시해주는 프로그램입니다.



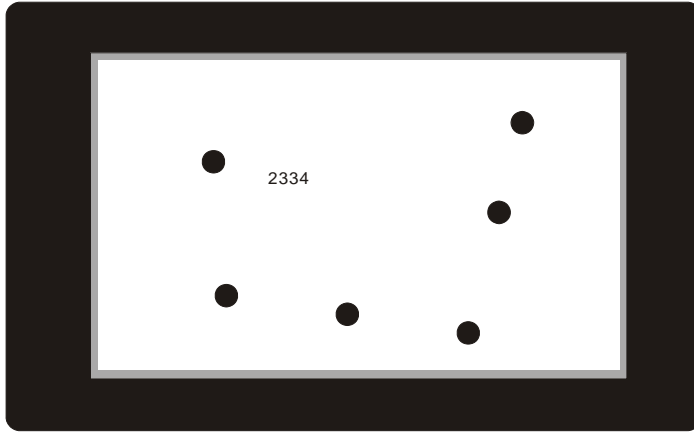
```
1 Const Device = CB405
2 Dim RX_BF(12) As Byte
3 Dim CONVBF As String * 5
4 #define Comport 3
5
6 Set ilcd Comport
7 Opencom Comport,38400,3,80,180
8 On Recv3 Gosub EVENTRTN
9 Set Until Comport,22,3 '항상 0X3이 END CODE임
10
11 Dim X1 As Integer, Y1 As Integer
12
13 @ "#@#@;Clear;"
14 @ "PRINT 'Touch Test',300,170;"
15 Do
16 Loop
17
18 EVENTRTN:
19   Geta Comport,RX_BF,11 '빈지역 터치
20   If RX_BF(1) = 0X38 And RX_BF(2) = 0X34 And RX_BF(3) = 0X31 Then
21     CONVBF = Chr(RX_BF(4)) + Chr(RX_BF(5)) + Chr(RX_BF(6)) 'x좌표
22     X1 = Valhex(CONVBF)
23     CONVBF = Chr(RX_BF(7)) + Chr(RX_BF(8)) + Chr(RX_BF(9)) 'y좌표
24     Y1 = Valhex(CONVBF)
25     @ "CIRCLEFILL ",Dp(X1-10)," ",Dp(Y1-10)," ",Dp(X1+10)," ",Dp(Y1+10)," ";
26   Endif
27   Return
```

F/W Id : 5380750 Line : 17 Program : 572 / 204800 Bytes, Data

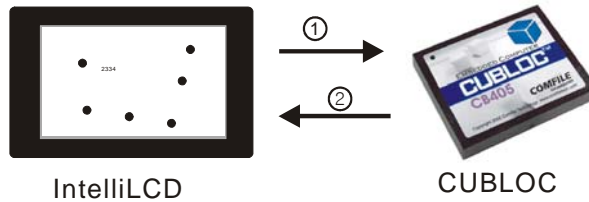
소스 파일명 <TOUCHPRINT.CUL>

본 사용설명서에 있는 소스파일은 comfile.co.kr 의 DOWNLOAD 자료실에 공개해 놓았습니다. 직접 실행해보고 싶으신분들은 다운로드 받은후 사용하세요.

이 프로그램을 실행 시키면 CT2400 화면을 터치한 지점에 원이 표시됩니다. 화면 중앙에는 카운트업증인 숫자가 표시됩니다.



화면 터치 입력지점을 알기 위해서는 “IntelliLCD”가 보내주는 이벤트를 해석해야합니다. IntelliLCD 에서는 화면이 터치될때마다, X 와 Y 좌표를 RS232 통신채널을 통해서 송신해줍니다. 이것을 CB405 에서 받아서 처리하는 것입니다.



- (1) 터치가 발생하면 위치정보를 CUBLOC 으로 송신합니다.
- (2) CUBLOC 에서는 터치 지점에 CIRCLE 을 표시하라고 명령합니다.

소스분석

```
Const Device = CB405
Dim RX_BF(12) As Byte
Dim CONVBF As String * 5
```

디바이스를 CB405 로 설정하고 변수를 선언합니다. RX_BF 라는 12 바이트 바이트 변수는 RS232 수신데이터를 임시저장할 변수입니다. CONVBF 도 수신데이터를 분석하기 위한 문자열 변수 입니다.

```
#define Comport 3
Set ilcd Comport
Opencom Comport,38400,3,80,180
```

SET ITL 명령과 OPENCOM 명령어입니다. COMPORT 라는 상수를 선언해서 정의합니다.

```
On Recv3 Gosub EVENTRTN
Set Until Comport,22,3 '항상 0X3 이 END CODE 임
```

ON RECV3 는 수신인터럽트가 발생되면 EVENTRTN 으로 점프하도록 하는 선언문입니다. 그 밑에 SET UNTIL 은 종료코드인 0X3 이 수신될 때, 인터럽트가 발생하도록 만드는 선언문입니다.

```
Dim X1 As Integer, Y1 As Integer
```

X1,Y1 는 화면 터치지점에 원을 표시하기 위해 좌표를 저장하기 위한 변수입니다.

```
@ "#@#@;Clear;"
@ "PRINT 'Touch Test',300,170;"
Do
Loop
```

화면을 클리어하고, 중간쯤에 “Touch Test”라는 문자를 표시합니다. 그리고 무한루프로 들어갑니다. 이후 터치입력이 있을때까지 CB405 는 아무것도 안하게 됩니다.

```
EVENTRTN:
Geta Comport,RX_BF,11 '빈지역 터치
```

이 부분이 터치입력이 발생되었을 때 실행되는 루틴입니다. 제일 처음 GETA 명령어로 11 바이트를 읽어, RX_BF 배열 변수에 저장합니다. CUBLOC 내부버퍼에 11 바이트가 반드시 쌓여있다는 확실한 검증없이 바로 11 바이트를 읽어오는 이유는 앞서 SET UNTIL 명령에의해 종료코드를 선언해 두었기 때문입니다.

IntelliLCD 의 이벤트는 11 바이트로 고정되어 있고, 맨 뒤에는 0X3 (16 진수 3)이 항상 따라오기 때문에, SET UNTIL 에의해 0X3 이 검출되었다는 것은 앞서 11 바이트의 수신이 모두 이루어졌다는 것을 뜻합니다.

따라서 곧바로 11 바이트를 읽어낼 수 있습니다.

수신된 이벤트는 다음과 같은 구성으로 되어 있습니다. (IntelliLCD 의 ASCII 수신모드 사용)

시작코드	이벤트번호	파라미터	종료코드
0X2	사전에 지정된 번호	이벤트별로 다른 내용	0X3
1 바이트	2 바이트	7 바이트	1 바이트

화면에 아무것도 없는 지역 (추후 설명할 컨트롤이나 스크린영역이 아닌곳)을 터치하면 “이벤트번호”가 캐릭터 “84”가 옵니다. ASCII 코드값으로는 0X38 과 0X34 가 옵니다.

빈 영역의 터치 이벤트발생시 수신되는 정확한 정보는 다음과 같습니다.

캐릭터값	0X2	‘8’	‘4’	1 또는 0	X 좌표	Y 좌표	0X3
ASCII 코드	0X32	0X38	0X34	0X31 또는 0X32			0X33
바이트수	1	1	1	1	3	3	1

수신된 데이터는 RX_BF 배열에 저장되어 있습니다. 여기에서 1 번요소와 2 번요소가 각각 0X38, 0X34 인지 확인하면 이 이벤트가 빈영역의 터치발생 이벤트인지 확인할 수 있습니다.

```
If RX_BF(1) = 0X38 And RX_BF(2) = 0X34 And RX_BF(3) = 0X31 Then
  CONVBF = Chr(RX_BF(4)) + Chr(RX_BF(5)) + Chr(RX_BF(6)) 'X 좌표
  X1 = Valhex(CONVBF)
  CONVBF = Chr(RX_BF(7)) + Chr(RX_BF(8)) + Chr(RX_BF(9)) 'Y 좌표
  Y1 = Valhex(CONVBF)
  @ "CIRCLEFILL ",Dp(X1-10),",",Dp(Y1-10),",",Dp(X1+10),",",Dp(Y1+10),";"
Endif
Return
```

빈 영역의 터치발생 이벤트라면, 터치지점의 X, Y 좌표를 읽어와 CIRCLEFILL 이라는 명령어를 만들어 다시 송신하는 과정입니다.

```
CONVBF = Chr(RX_BF(4)) + Chr(RX_BF(5)) + Chr(RX_BF(6)) 'X 좌표
```

CONVBF 는 문자열 변수 입니다. 수신데이터가 저장된 RX_BF 로부터 데이터를 추출해서 CONVBF 에 저장하고 이것을 다시 VALHEX 명령을 사용해 숫자로 바꾸어 X1 이라는 정수형 변수로 옮기는 과정입니다. 같은 방법으로 Y 좌표도 만들어 옵니다.

```
@ "CIRCLEFILL ",Dp(X1-10),",",Dp(Y1-10),",",Dp(X1+10),",",Dp(Y1+10),";"
```

CIRCLEFILL 명령어는 X1,Y1,X2,Y2 를 파라미터로 주게 되어 있는데, 이 지점을 대각선으로하는 가상의 사각형을 꽉채우는 원을 그립니다. 위 소스에서 마이너스 10 과 플러스 10 을 사용하여 지름 20 의 크기를 갖는 원을 그리게 됩니다.

샘플 프로그램 3 : 컨트롤 표시와 입력

이번 프로그램은 화면상에 여러 개의 버튼 컨트롤을 표시하고, 컨트롤이 터치되었을 때 반응하는 프로그램입니다.

소스 파일명 <CONTROLTEST.CUL>

```
Const Device = CB405
Dim RX_BF(12) As Byte

#define Comport 3

Set ilcd Comport
Opencom Comport,38400,3,80,180
On Recv3 Gosub EVENTRTN
Set Until Comport,22,3 '항상 0x3 이 END CODE 임

Dim I As Integer

#define bt11 "30,30,190,90"
#define bt12 "30,120,190,180"
#define bt13 "30,210,190,270"
#define bt14 "30,300,190,360"
#define bt15 "30,390,190,450"

#define bt21 "220,30,380,90"
#define bt22 "220,120,380,180"
#define bt23 "220,210,380,270"
#define bt24 "220,300,380,360"
#define bt25 "220,390,380,450"

#define bt31 "410,30,570,90"
#define bt32 "410,120,570,180"
#define bt33 "410,210,570,270"
#define bt34 "410,300,570,360"
#define bt35 "410,390,570,450"

#define bt41 "610,30,770,90"
#define bt42 "610,120,770,180"
#define bt43 "610,210,770,270"
#define bt44 "610,300,770,360"
#define bt45 "610,390,770,450"

I=0
@ "#@#@;Clear;"
@ "PRINT 'Wait..';"

Wait 200
@ "#@#@;Clear;"
'@ "ENABLEDEMO 0;"
@ "TEXTFONT 34,'굴림';"
@ "TEXTCOLOR &HF09F00;"
```

```

@ "PRINT '냉매충진 시스템',235,30;"
@ "TEXTFONT 22,'굴림';"
@ "TEXTCOLOR &H009F00;"
@ "PRINT '삼일전장 주식회사',280,410;"
@ "CTRLFONT 16,'바탕',1;"
@ "BUTTON 1,",'bt11','가스충전';"
@ "BUTTON 2,",'bt12','가스재생';"
@ "BUTTON 3,",'bt13','충전량확인';"
@ "BUTTON 4,",'bt14','재생통청소';"
@ "BUTTON 5,",'bt15','충전종료';"
@ "BUTTON 6,",'bt41','엔진체크';"
@ "BUTTON 7,",'bt42','오일체크';"
@ "BUTTON 8,",'bt43','배기량체크';"
@ "BUTTON 9,",'bt45','현재시간설정";

@ "STATIC 10,320,320,500,360,'','$CENTER;"

Do
  @ "PRINT '"+Dp(I)+"'",280,200;"
  Incr I
  Wait 500
Loop

EVENTRTN:
  Geta Comport,RX_BF,11 '콘트롤 터치입력시

  If RX_BF(1) = 0X38 And RX_BF(2) = 0X31 Then
    @ "SETCTRL 10,$CAPTION,'입력 : "+Hp(RX_BF(4))+"'";
  Endif
  Return

```

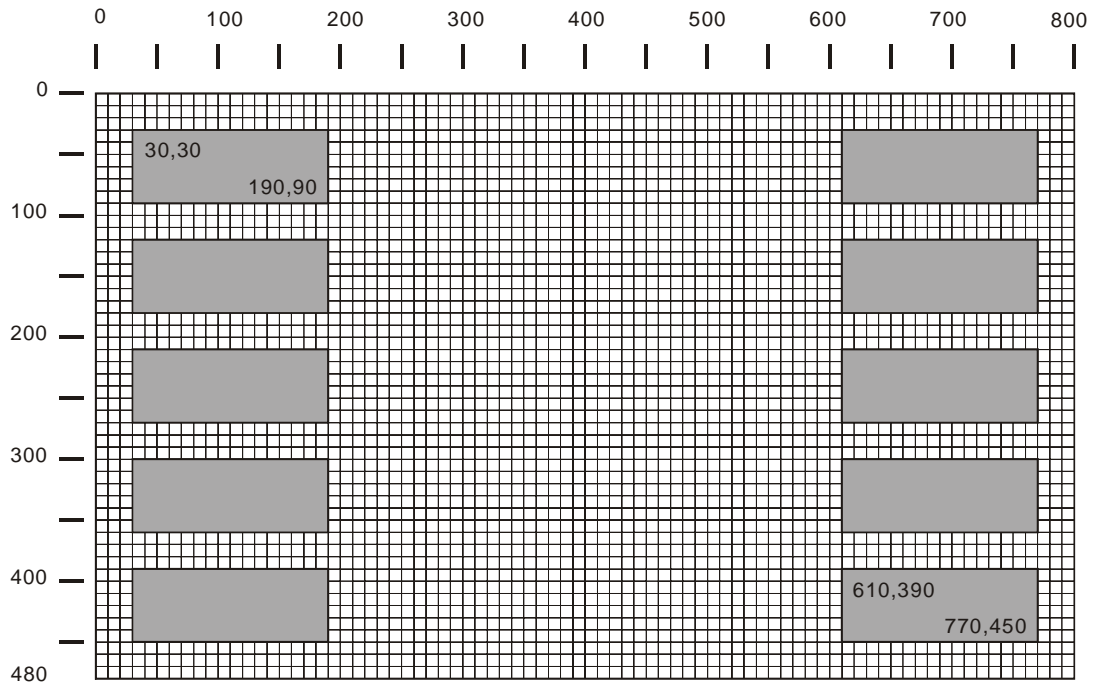
실행결과는 다음과 같습니다. 버튼을 누르면 그 버튼의 콘트롤 번호가 화면 하단에 표시됩니다.



소스분석

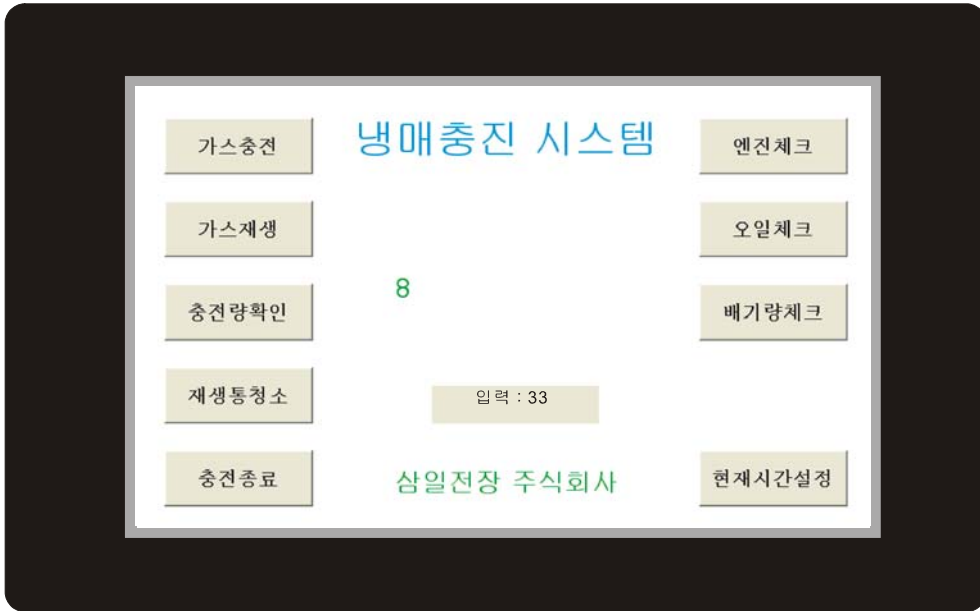
```
#define bt11 "30,30,190,90"  
#define bt12 "30,120,190,180"  
#define bt13 "30,210,190,270"  
#define bt14 "30,300,190,360"  
#define bt15 "30,390,190,450"  
  
#define bt21 "220,30,380,90"  
#define bt22 "220,120,380,180"  
#define bt23 "220,210,380,270"  
#define bt24 "220,300,380,360"  
#define bt25 "220,390,380,450"
```

이 부분은 화면을 일관성있게 나누기위해 사전에 정의한 좌표계입니다. 다음 그림을 보면 어떻게 화면을 분할하였는지 알 수 있습니다.



여러 개의 버튼이 서로 가로, 세로열이 일치하도록 하기 위해서 사전에 모눈종이에 스케치한후, 최종 좌표만을 소스프로그램상에 기록한 것입니다.

그 결과 다음과 같은 화면 구성을 할 수 있습니다.



```
@ "CTRLFONT 16, '바탕', 1;"  
@ "BUTTON 1, ", bt11, ", '가스충전';"
```

CTRLFONT 는 컨트롤에 사용할 글꼴을 선언하는 명령어입니다. 한번 선언해 두면 이후 컨트롤에 모두 적용됩니다. BUTTON 명령어에 의해서 화면상에는 버튼이 표시되고, 이 것은 컨트롤 1 번으로 선언됩니다. 이후 이 컨트롤이 터치입력된다면, 컨트롤 이벤트가 발생합니다.

컨트롤이란..

인텔리 LCD 는 버튼, 슬라이드바, 체크박스, 토글스위치, 진행바 (Xprogress, Yprogress), 스택틱표시 등, 총 5 가지의 컨트롤을 가지고 있습니다. (향후 추가될 수도 있음)

이러한 컨트롤들은 화면에 표시된 뒤 독립적으로 어떤 동작을 수행합니다. BUTTON 을 예로 들면, 터치입력이 있었을 때, 눌러진 모양을 표시하고, 터치입력이 없다면 나온모양을 표시합니다. 즉 유저는 일일이 눌러진 모양과 나온 모양을 그려줄 필요가 없게 됩니다.

토글의 경우에도 한번 터치되면 활성화 (원에 점이 표시됨)되고 다시한번누르면 비활성화 (빈 원만 표시)됩니다.

컨트롤은 각기 자신의 동작을 수행하면서, 터치입력된 결과물을 RS232 로 송신하게 되는데, 이것이 바로 컨트롤 이벤트입니다.

콘트롤 이벤트는 다음과 같은 형식으로 발생되어 집니다.

캐릭터값	0X2	'8'	'1'	콘트롤 ID	예약영역	0X3
ASCII 코드	0X32	0X38	0X31	XX XX	"*"가 4 개	0X33
바이트수	1	1	1	2 (상위가 먼저옴)	4	1

이것을 분석한다면 어떤 콘트롤이 입력되었는지 알 수 있습니다.

```
EVENTRTN:
  Geta Comport,RX_BF,11 '콘트롤 터치입력시

  If RX_BF(1) = 0X38 And RX_BF(2) = 0X31 Then
    @ "SETCTRL 10,$CAPTION,'입력 : "+Hp(RX_BF(4))+"'";"
  Endif
```

소스중 이 부분은 콘트롤이벤트 발생시 처리하는 RS232 수신 인터럽트 루틴입니다. 소스에서 알 수 있듯이 수신된 데이터의 두번째, 세번째 바이트가 각각 0X38, 0X31 이었을때, 콘트롤 ID 의 하위바이트를 읽어서 화면상에 있는 STATIC 콘트롤에 표시합니다.

STATIC 콘트롤은 표시를 위한 전용 콘트롤입니다. PRINT 와 다른점은 항상 고정된 위치를 가지고 있기 때문에, 안정적인 표시가 가능하다는 것입니다. STATIC 의 경우 표시전용이기에 발생하는 이벤트는 없습니다. 화면에 숫자를 표시하기위해서 STATIC 콘트롤을 사용하는 이유는 항상 고정된 위치표시를 보장하기 때문입니다.

샘플 프로그램 4 : 스크린 기능 활용

앞의 방법으로는 시중에서 볼 수 있는 고급스러운 화면을 구사하기 어렵습니다. 포토샵이나 코렐드로우와 같은 그래픽 전용소프트웨어를 사용하여 배경화면과 버튼화면을 따로 그린뒤, 이것을 화면에 표시하는 방법을 많이 사용합니다. 인텔리 LCD 에서는 이러한 방법을 좀더 쉽게 구현할 수 있도록 “스크린”이라는 기능을 지원합니다. 이 기능을 사용하기 위해서 다음과 같은 그림 2 장을 준비합니다.



<화면 A> 기본화면



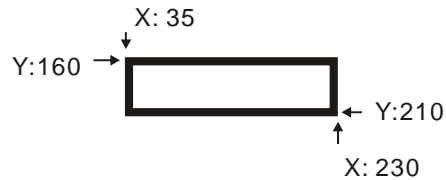
<화면 B> 보턴영역을 다른색으로 만든 화면, 즉 보턴이 눌러졌을때의 화면

이 두장의 그림은 실제로는 같은 그림입니다만, 터치가능한 버튼부분의 색상만 다르게 한 것입니다. 터치입력을 받아들이는 부분 (보턴영역)을 다른색으로 만든 것입니다.

<그림판> 프로그램을 이용하여 이 그림을 불러들인 후 이 지점의 좌표를 조사합니다.



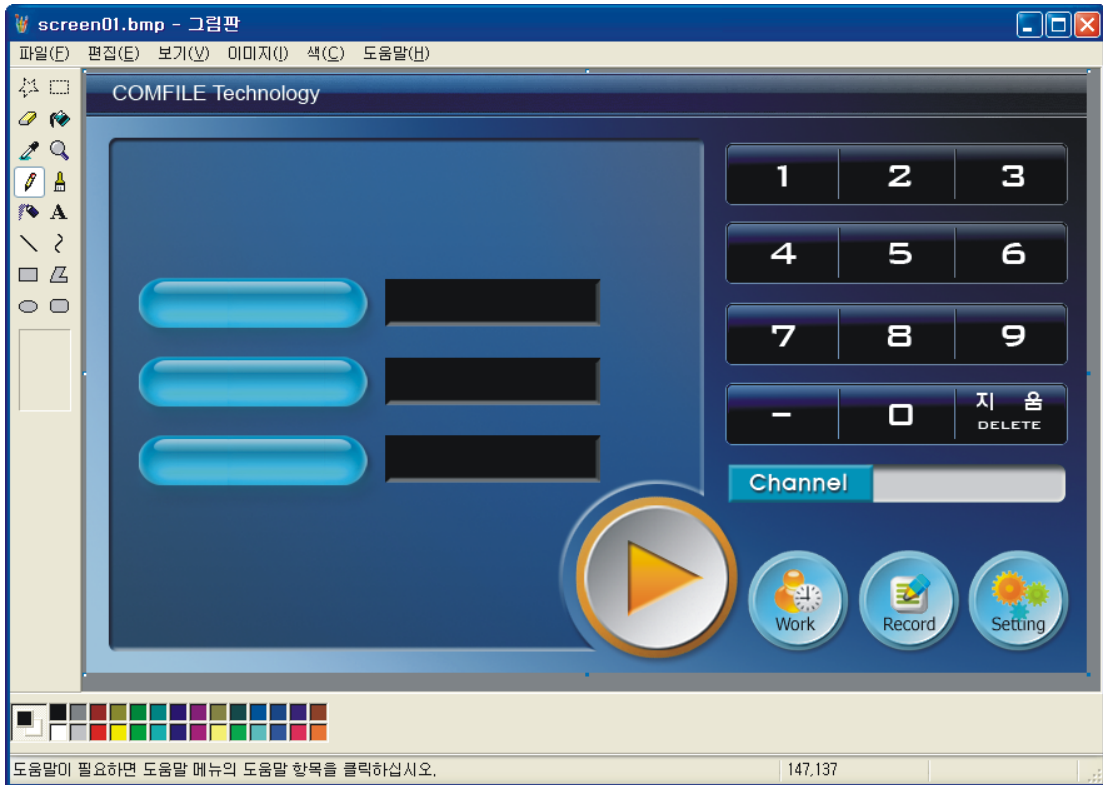
이 화면의 경우 다음과 같은 좌표값으로 확인되었습니다. 이 지점을 터치했을 경우 <화면 B>에서 같은 영역을 카피해서 표시하도록 한다면, 터치입력여부를 확실히 알 수 있도록 다른색으로 이 지점이 표시된 것처럼 보이게 됩니다.



다른 터치영역도 많이 있지만, 일단 이 하나의 버튼만을 가지고 “스크린”기능을 구현해 보도록 하겠습니다.

TIPS: 좌표 조사하기

화면의 좌표를 조사하는 방법은 의외로 간단합니다. <그림판> 프로그램 (윈도우 설치시 기본적으로 포함된 프로그램입니다.)으로 화면 A 를 불러옵니다.



이곳에 좌표가 표시됩니다.

화면에 커서를 위치시키면, 현재 커서가 있는곳의 좌표가 <그림판>프로그램 하단에 표시됩니다.

SCREEN SCRIPT 파일 작성

스크린 기능을 사용하려면, 별도의 스크립트 파일을 작성해야 합니다. 이 스크립트 파일에는 위치정보와 파일명등을 기록합니다.

```
common main_image=screen01.bmp  
common press_image=screen01press.bmp;clip_mode=yes  
area id=1:rect=35,160,230,210;
```

이 스크립트 파일을 CUBLOC STUDIO 상에 입력하는 것이 아니라, 별도의 텍스트 파일로 작성하셔야 합니다. 메모장등의 에디터를 사용하여 작성하고 적당한 파일명으로 저장하십시오.

스크립트 파일을 한줄 한줄 살펴보도록 하겠습니다.

```
common main_image=screen01.bmp;
```

화면 A 에 해당하는 기본화면을 지정하는 라인입니다. 본 예제에서는 screen01.bmp 라는 파일명을 사용하였습니다.

```
common press_image=screen01press.bmp;
```

화면 B 에 해당하는 화면을 지정하는 라인입니다. Screen01press.bmp 라는 파일명을 사용하였습니다.

```
clip_mode=yes;
```

clip_mode 는 반드시 yes 로 선택하십시오.

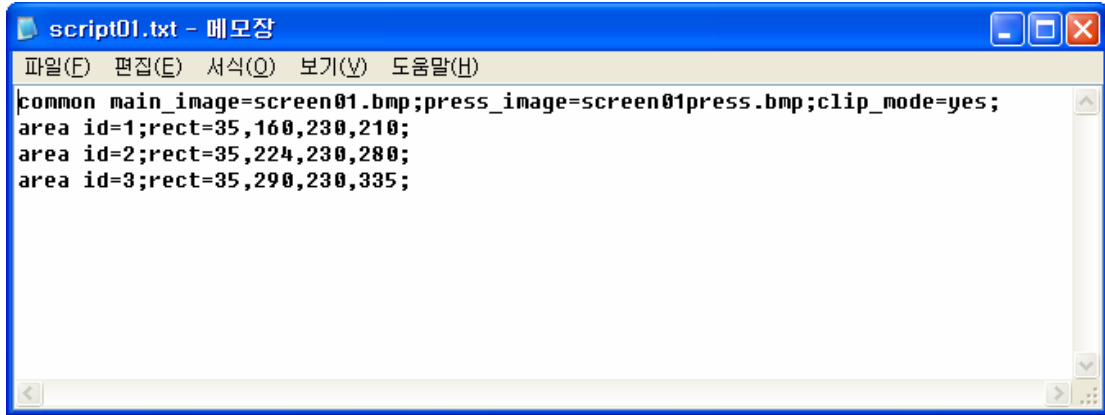
```
area id=1:rect=35,160,230,210;
```

끝으로 영역지정하는 구문입니다. <영역 ID>를 1 로 하고, rect = 에는 X1,Y1, X2, Y2 순으로 좌표값을 적어줍니다. 다음 보턴이 있을 경우 인덱스번호를 2 로 하고 좌표를 적어주면 됩니다. 이런식을 이 화면에서 입력가능한 보턴을 계속 적어주는 것으로 스크립트 파일 작성을 마칩니다.

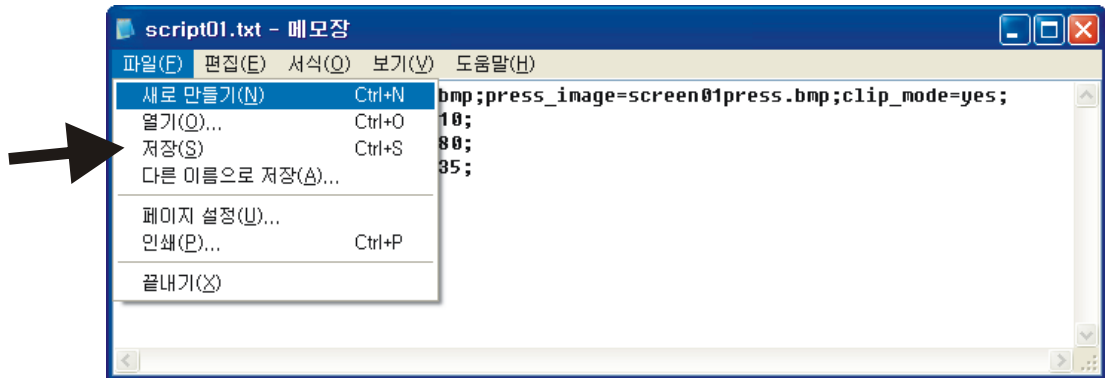
<영역 ID>는 1 부터 255 까지 사용할 수 있습니다. 마치 컨트롤 ID 와 같은 기능을 한다고 이해하시면 됩니다. 다만 하나의 스크린 ID 안에 1 에서 255 까지 사용할 수 있습니다. 다른 스크린 ID 에서는 다시 1 에서 255 까지 사용할 수 있습니다.

TIPS : 메모장으로 스크립트 파일 작성하기

<메모장>은 윈도우에 기본적으로 포함된 프로그램입니다. 주로 간단한 문서를 작성할 때 사용합니다.



메모장을 실행시킨 후 위와 같은 내용을 입력하세요. 중요한 것은 모든입력을 마친후 반드시 세이브 해야한다는 것입니다. 세이브된 파일을 적당한 폴더에 위치시킨 후 이 파일을 CT2400 의 IntellLCD 영역에 다운로드 하면 됩니다.



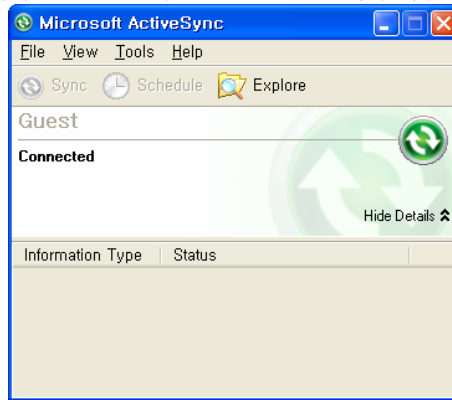
CT2400 의 IntelliLCD 폴더로 다운로드

CT2400 에 있는 INTELLILCD 폴더에는 다음 3 가지 파일 다운로드 되어야 합니다.

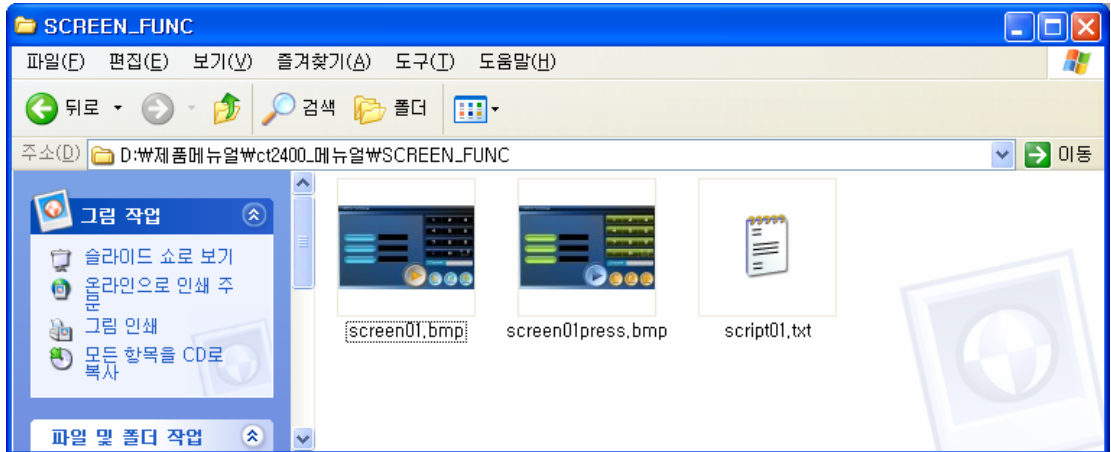
1. 화면 A 에 해당하는 BMP 파일
2. 화면 B 에 해당하는 BMP 파일
3. 스크립트 파일

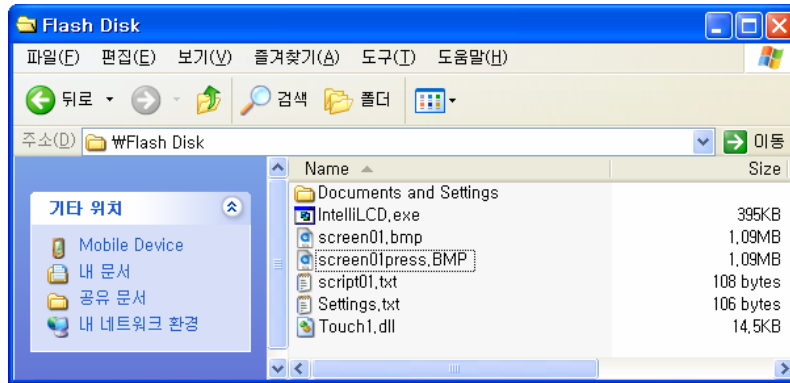
PC 와 CT2400 을 USB 케이블로 연결하시면, 다음 화면과 같은 ACTIVE SYNC 가 표시됩니다. 만약 ACTIVE SYNC 를 설치하지 않으신 분이라면 microsoft.com 에서 ACTIVE SYNC 를 다운로드 받아서 PC 상에 설치하십시오. (무료)

이 화면에서 <Explore>를 클릭하시면 CT2400 의 INTELLILCD 폴더의 내용이 표시됩니다.



그리고 PC 상에 있는 이 3 개의 파일을 INTELLILCD 상의 폴더로 카피하십시오.





<CT2400의 IntelliLCD 폴더의 내용>

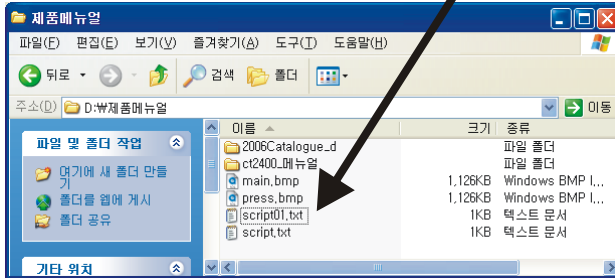
SD CARD 가 있는 경우에는 storage card 폴더 로 복사하고, 없는 경우에는 Flash Disk 폴더로 복사하십시오. Flash Disk 폴더의 용량에는 여유가 없으므로, 많은 양의 bmp 파일을 사용하는 경우에는 반드시 별도의 SD CARD 를 사용하시기 바랍니다.

TIPS : 파일복사하는 방법

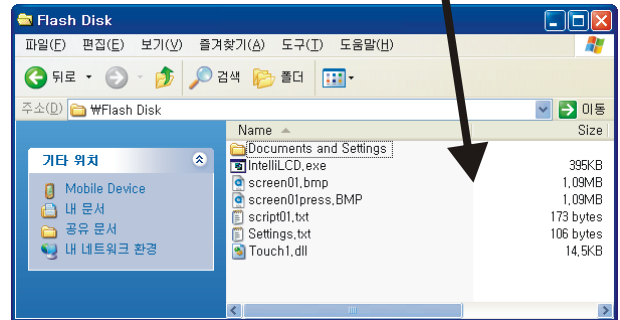
여러분의 PC 에서 CT2400 의 IntelliLCD 폴더로 복사하기 방법은 마우스 드레그/ 드롭을 사용하시면 편리합니다. 탐색기로 두개의 폴더를 오픈합니다. 하나는 PC 상의 파일이 저장된 곳이고, 다른 하나는 CT2400 의 IntelliLCD 의 폴더입니다.

이 곳의 파일을 드레그해서...

이곳에 드롭합니다.



PC상의 폴더를 탐색기로 오픈해놓은것



IntelliLCD상의 폴더를 ActiveSync의 Explore로 오픈해 놓은것

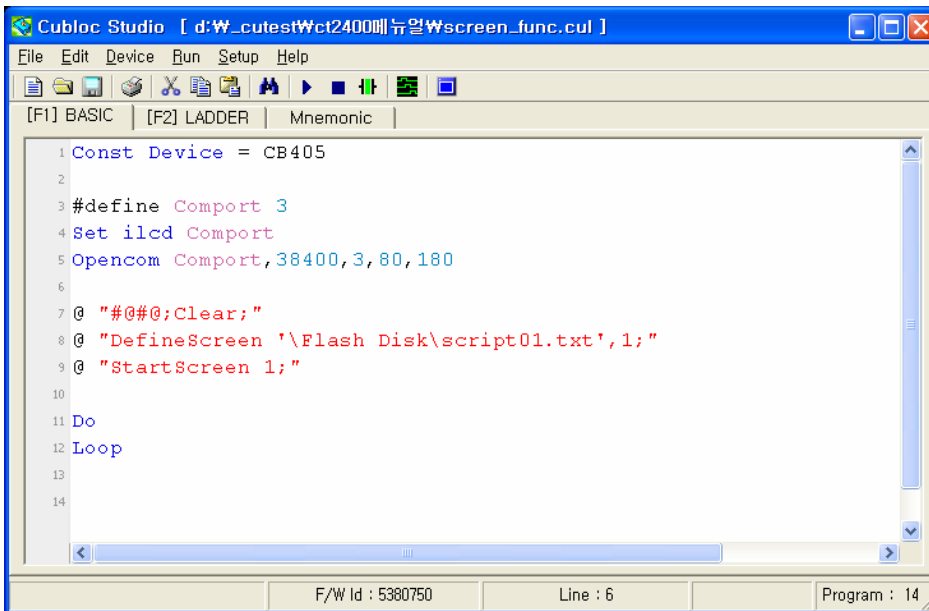
이렇게 하면 쉽게 파일을 복사할 수 있습니다.

CUBLOC 소스작성

이제 CUBLOC STUDIO 에 프로그램을 작성할 차례입니다. 앞서의 방법이 조금 복잡해 보이지만, 이러한 방법을 사용하지 않는다면, 유저여러분께서 일일이 그림을 잘게 분할해서 별도의 파일로 저장한뒤, CUBLOC STUDIO 상에서 터치지점 좌표를 읽어내서, 비교하고, 해당 파일을 표시하는 더욱 복잡한 방법을 사용해야합니다.

SCREEN 기능을 활용하면 단 2 장의 BMP 파일과 SCRIPT 파일 하나만 있으면 CUBLOC STUDIO 상에서는 매우 간단하게 소스를 작성할 수 있습니다. 복잡한 좌표검색과 그림 분할등은 SCREEN 기능이 알아서 다 해주기 때문입니다.

다음은 앞서 다운로드한 자료들을 CT2400 상에서 실행해주는 소스프로그램입니다.



```
Cubloc Studio [ d:\W_cutest\Wct2400메뉴일\Wscreen_func.cul ]
File Edit Device Run Setup Help
[F1] BASIC | [F2] LADDER | Mnemonic
1 Const Device = CB405
2
3 #define Comport 3
4 Set ilcd Comport
5 Opencom Comport, 38400, 3, 80, 180
6
7 @ "#0#0;Clear;"
8 @ "DefineScreen '\Flash Disk\script01.txt', 1;"
9 @ "StartScreen 1;"
10
11 Do
12 Loop
13
14
F/W Id : 5380750 Line : 6 Program : 14
```

<소스파일명 : screen_func.cul>

이 소스를 입력후 다운로드하면 CT2400 화면에는 미리 다운로드해둔 <화면 A>가 보일것입니다. 그리고 첫번째 버튼을 터치하면 그 부분만 다른색으로 바뀌어서 표시됩니다. 화면에서 손을 띄면 다시 원래 버튼이 표시됩니다.

소스를 보면 DefineScreen 이라는 명령어를 보실 수 있습니다. 바로 이 명령어가 앞서작성한 스크립트 파일을 "스크립 id" 1 번에 할당시켜주는 명령어입니다. 이 명령어는 화면에 아무런 영향을 주지 않습니다. 다만 서로 짝지거나 시켜주는 것으로 이해하시면 됩니다.

그리고 StartScreen 1 이라는 명령어에 의해 실제로 화면상에는 <화면 A>가 표시됩니다.

버튼 이벤트발생

스크린 기능을 활용해서 멋있는 화면을 표시하고 버튼을 터치했을 때의 보턴이 하이라이트되는 부분까지 완성하였습니다. 그러면 이제는 어떤 버튼이 터치되었는지 알아내야 합니다. 이를 위해서 스크린기능이 동작중에도 이벤트가 발생하는 기능이 있습니다.

스크린동작중 터치한 지점이 버튼영역에 속해있을 때 다음과 같은 이벤트가 발생합니다.

캐릭터값	0X2	'8'	'C'	스크린 ID	영역 ID	예약영역	0X3
ASCII 코드	0X32	0X38	0X43	XX XX	XX XX	"*"가 2 개	0X33
바이트수	1	1	1	2 (상위가 먼저옴)	2	2	1

테스트를 위해 스크립트 파일에 보턴 2 개를 더 추가하였습니다.

```
common main_image=screen01.bmp;press_image=screen01press.bmp;clip_mode=yes;
area id=1;rect=35,160,230,210;
area id=2;rect=35,224,230,260; // 추가된 부분
area id=3;rect=35,290,230,335; // 추가된 부분
```

스크립트파일에 위 추가내용을 입력한뒤, 저장하고 CT2400 의 IntelliLCD 폴더상에 카피하십시오.

다음 소스를 실행시키면 보턴을 누를 때 마다, 해당보턴의 영역 ID 값이 화면상에 표시됩니다.

<소스파일명 : screen_func2.cul>

```
Const Device = CB405
Dim RX_BF(12) As Byte

#define Comport 3
Set ilcd Comport
Opencom Comport,38400,3,80,180
On Recv3 Gosub EVENTRTN
Set Until Comport,22,3 '항상 0X3 이 END CODE 임

@ "#@##;Clear;"
@ "DefineScreen '\Flash Disk\script01.txt',1;"
@ "StartScreen 1;"
@ "CtrlColor &hfffffff,0;"
@ "STATIC 10,242,170,405,198,',,$CENTER;"

Do
Loop

EVENTRTN:
  Geta Comport,RX_BF,11 '콘트를 터치입력시
```

```
If RX_BF(1) = 0X38 And RX_BF(2) = 0X43 Then
    @ "SETCTRL 10,$CAPTION,'입력 : "+Hp(RX_BF(6))+"'";
Endif
Return
```

다음은 위 소스프로그램 실행화면 입니다.



여러 개의 화면 운용법

하나의 시스템을 만들다 보면 여러 개의 화면으로 유저인터페이스를 구성하는 경우가 대부분입니다. 앞서 설명한 SCREEN 기능을 활용하면, 보다 쉽게 그리고 멋진 화면을 구현할 수 있습니다. 뿐만 아니라 여러 개의 화면을 운용할 수 있는 방법도 제공하고 있습니다.

앞서 설명한 “스크린 id”에 주목하세요. DefineScreen 이라는 명령어는 스크립트 파일을 “스크린 id”와 짝짓기 시켜주는 설정 명령어입니다.

그리고 실제로 스크립트 파일의 내용을 화면상에 표시하고 버튼을 동작시키도록 활성화해주는 명령어가 “StartScreen”이라는 명령어입니다.

이와는 반대로 현재 활성화된 스크린 ID에 해당하는 모든동작을 중지시켜주는 명령어가 “EndScreen”이라는 명령어입니다.

```
@ "DefineScreen '\Flash Disk\script01.txt',1;"
@ "StartScreen 1;"
@ "EndScreen 1;"
```

만약 여러분의 10 개의 화면을 준비중이라면 스크립트 파일도 10 개를 작성해야 합니다. 그리고 DefineScreen 명령어로 10 개의 스크립트를 모두 특정 “스크린 id”에 할당시켜주어야 합니다.

```
@ "DefineScreen '\Flash Disk\script01.txt',1;"
@ "DefineScreen '\Flash Disk\script02.txt',2;"
@ "DefineScreen '\Flash Disk\script03.txt',3;"
@ "DefineScreen '\Flash Disk\script04.txt',4;"
@ "DefineScreen '\Flash Disk\script05.txt',5;"
@ "DefineScreen '\Flash Disk\script06.txt',6;"
@ "DefineScreen '\Flash Disk\script07.txt',7;"
@ "DefineScreen '\Flash Disk\script08.txt',8;"
@ "DefineScreen '\Flash Disk\script09.txt',9;"
@ "DefineScreen '\Flash Disk\script10.txt',10;"
```

이렇게 하면 스크린 id 1 번부터 10 번까지 동작준비가 된 것입니다. 이제 StartScreen 명령어로 어떤 화면이던지 여러분이 원하는데로 표시할 수 있습니다. 해당 화면이 표시되면 스크립트파일안에 있는 내용대로 보턴동작도 수행합니다.

다른 화면으로 넘어가려면 먼저 기존화면을 EndScreen 명령어로 비활성화 시켜준뒤, 다른 화면을 StartScreen 하면 됩니다. 기존 화면의 보턴영역은 모두 비활성화 되므로, 새로운 화면의 보턴영역이 활성화될 것입니다.

스크린 id 는 1 부터 255 까지 사용가능합니다.

예를 들면 다음 화면과 같이, 3 개의 화면이 준비되어 있고, 어떤 버튼을 눌렀을 때 다음화면으로 넘어가게 되는 것을 의미합니다.



Screen ID : 1



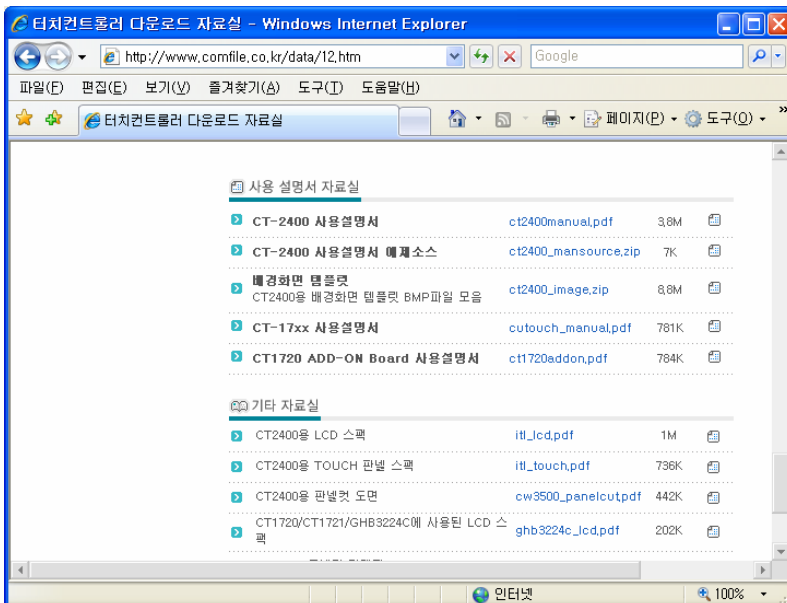
Screen ID : 2

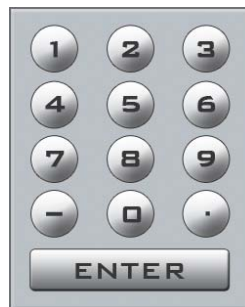
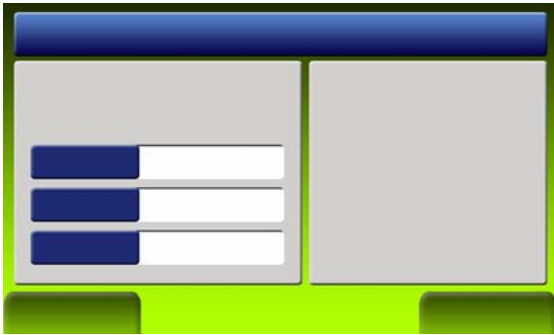


Screen ID : 3

템플릿 파일

www.comfile.co.kr에 배경화면으로 쓸 수 있는 BMP 파일이 공개되어 있습니다. 이중 여러분의 시스템에 적합한 화면을 가지고, 수정해서 사용하십시오. 포토샵등의 프로그램을 이용해서 수정하시면 됩니다. 자료는 다음위치에서 찾을 수 있습니다. <다운로드 자료실>의 TOUCH CONTROL 자료실, 하단에 보시면 <배경화면 템플릿>이라고 있습니다.





이외에도 다양한 템플릿이 있습니다.

제 4 장

I/O 콘트롤

써모커플 입력

CT2400 에는 써모커플 입력을 받을 수 있는 기능이 포함되어 있습니다. 실제로는 CB405 에 써모커플 입력칩인 MAX6675 를 내장하고 있으며, 이 칩을 CB405 와 연결시켜 놓은 것입니다. MAX6675 는 K 타입의 써모커플을 연결하여 온도를 읽어오는 칩입니다. CB405 의 포트 5,6,7 번 과 MAX6675 가 연결되어 있으며 아래 소스는 MAX6675 에서 온도를 읽어 화면에 표시하는 샘플 프로그램입니다.

소스 파일명 <TC_INPUT.CUL>

```
'=====
'
'           κ 써머커플 온도센서 사용하기
'
'=====
' 원칩으로 되어 있는 MAX6675 를 사용하여 간단하게 κ 써머커플 온도센서를 이용할 수 있다.
' κ 써머커플은 +, - 극성이 있기때문에 방향을 확인하고 연결해서 사용해야 한다.
' 방향확인은 연결해서 온도가 정상으로 올라가지 않고 거꾸로 내려가면 극성이 바뀐것이다.
' 최종 온도 출력 값은 실온과의 오차가 발생하기 때문에 보정이 필요하다.

' #define K_Therm_offset   750 에서 750 이 바로 +7.50℃를 낮추기 위한 보정 값이다.

Const Device = CB405

Set Ladder Off
#define SO 5
#define SCK 7
#define CS 6
#define K_Therm_offset      750

High CS

Dim K_Therm As Integer
Dim Display_Value As String*6

Ramclear

Do
  Low CS
  K_Therm = Shiftin(SCK,SO,1,16)
  High CS
  If K_Therm.BIT2 = 1 Then
    Debug "Open Thermocouple!",Cr
  Else

    K_Therm = K_Therm >> 3
    ' 오차값을 빼고 100 배한 온도값을 문자열로 바꿈
    Display_Value = Dec (K_Therm*100/4 - K_Therm_offset)
    ' 소수점 이하 두 자리까지 표시하도록 문자열을 수정함
    Display_Value = Left(Display_Value,Len(Display_Value)-2) + "." +
```



```

Right(Display_Value,2)
    Debug "Temperature: ",Display_Value," C",Cr

    Endif
    Wait 1000
Loop

End

```

CT2400 의 서모커플 사용환경은 다음과 같습니다.

- 측정범위 : 0 도 에서 1024 도
- 분해능 : 0.25 도
- 정확도 : +/- 1.5 도
- 측정주기 : 1 초
- 센서 : K-써머커플 (절연형)

주의할 점은 반드시 절연형 K 타입 서머커플을 사용해야하며 길이는 2 미터보다 작은길이를 사용해야 합니다. 그리고 장비를 세팅이 끝난후 측정지점의 실제온도와 측정값과의 차이가 발생할 수 있는데, 이것은 보정치를 통해서 보정해주어야 합니다.

실제측정된 온도와 MAX6675 가 읽어온 값에서 차이가 발생하는 이유는 케이블길이에 의한 저항치 변화, 그리고 CT2400 자체의 발열때문입니다. 또한, ADD-ON Board 단자의 +5V 전원을 외부 장치에 이용할 경우, 전원 Drop 으 로 인한 온도 편차가 발생할 수 있습니다.

따라서 별도의 온도계로 서머커플설치 지점의 온도를 측정한 뒤, 보정값을 결정하시는 것이 좋습니다.

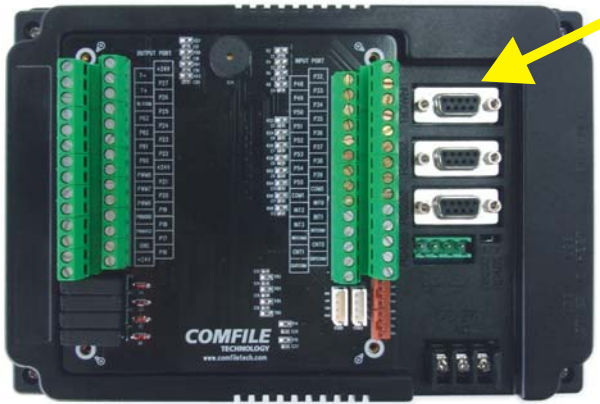
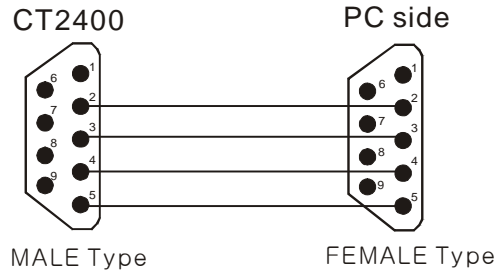
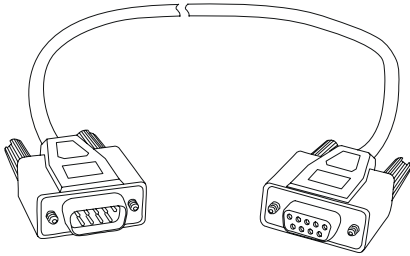
위의 예제 프로그램에서는 +7.5 도를 보정치로 하였습니다.

제 5 장

트러블슈팅

다운로드 케이블 연결

CUBLOC 을 사용하기 위해서 우선적으로 PC 와의 RS232 다운로드 케이블을 연결해야 합니다. 다운로드 케이블은 1:1 연결을 사용하며 총 9가닥 중 4가닥의 신호 선만을 사용합니다.



PC 와 연결

TIPS

다운로드가 안된다면, 우선 SETUP 메뉴에 있는 FIRMWARE DOWNLOAD 를 실행시켜 보시기 바랍니다. 펌웨어가 제대로 인스톨되어 있지 않다면, RS23 통신에러 또는 모듈을 찾을수 없다는 에러메시지가 표시됩니다.

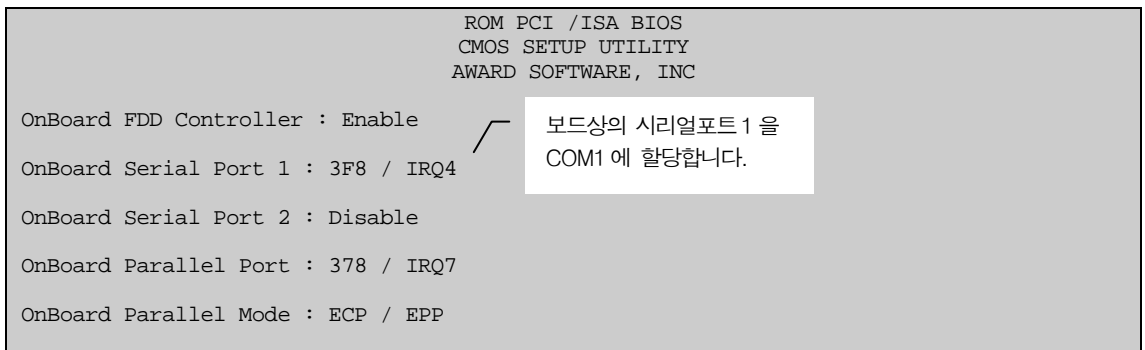
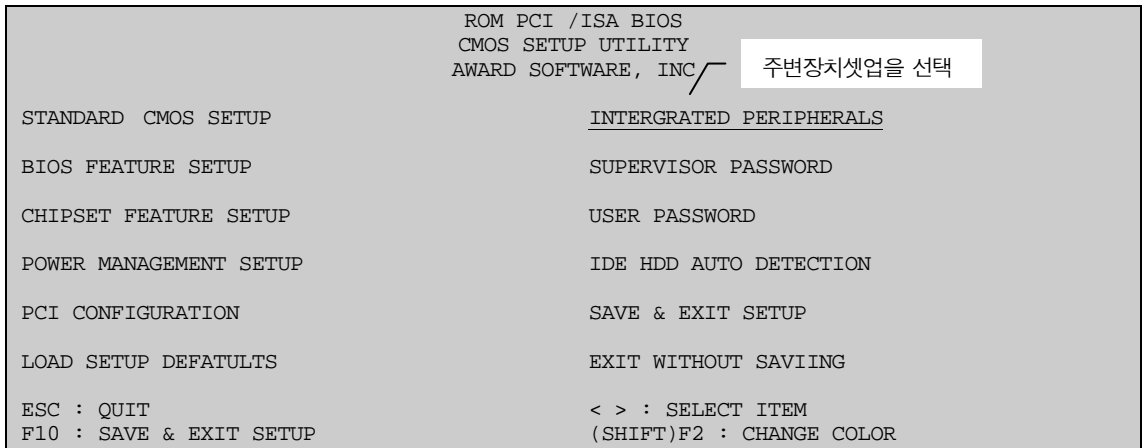
개발이 완료된뒤에는 반드시 DOWNLOAD 케이블을 연결하지 않은 상태에서 운전 (실행)하셔야 합니다. 다운로드포트가 연결된 상태라면, 외부 노이즈 (임펄스등)으로 인한 리셋현상 및 PC 를 리부팅할 때 리셋이 발생할 수 있습니다.

PC 통신포트의 점검

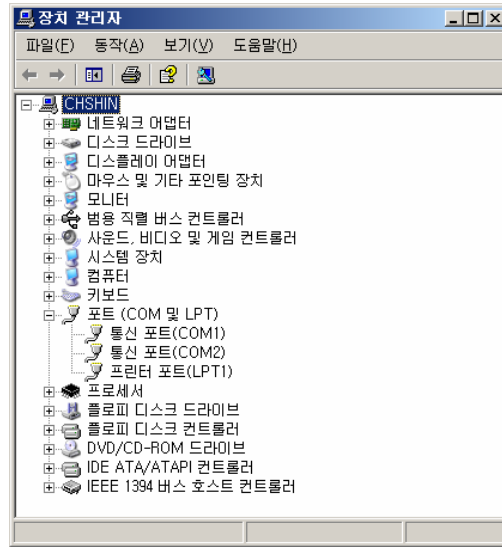
간혹 PC 상의 통신포트 설정이 잘못되어 있어, CUBLOC 다운로드가 안 되는 경우가 있습니다. PC 에는 보통 1~2 개의 RS232 통신포트가 내장되어 있지만, 이 통신포트를 사용 가능한 상태로 만들어 주는 것은 유저의 몫입니다. PC 의 RS232 통신포트를 제대로 동작시키기 위해서는 BIOS 셋업과 윈도우 드라이버설치까지 제대로 구성되어 있어야 하기 때문입니다.

우선 BIOS 상에서 RS232 통신포트 설정방법에 대하여 설명하겠습니다. 컴퓨터 부팅 시에 DEL 키를 누르면 (PC 에 따라서 다른 키를 사용하는 경우도 있음) BIOS SETUP 화면이 표시됩니다. BIOS 의 종류에 따라서 다른 화면 이 표시됩니다.

다음 화면그림을 참조하여, 마더보드상에 있는 RS232 통신포트가 Disable 상태로 되어 있지 않도록 설정하시기 바랍니다.

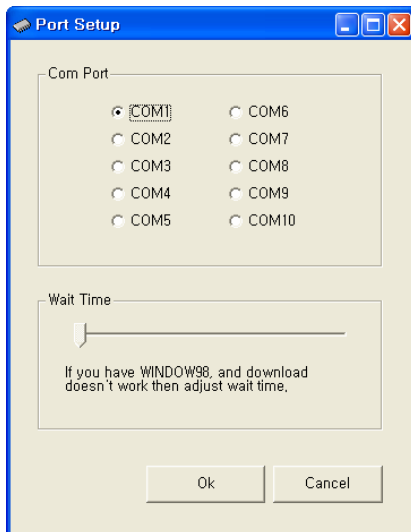


다음은 윈도우가 COM 포트 드라이버를 제대로 인식하고 있는지 체크해보아야 합니다. <내 컴퓨터 - 시스템 정보 표시 - 하드웨어 - 장치관리자>까지 선택하여 들어가 보면, 아래 화면이 표시됩니다. (윈도우 XP 의 경우)



이 다이얼 로그 박스에서 “통신포트 (COM1)”이 표시되어 있는지 확인하시기 바랍니다. 만약 통신포트가 나와있지 않다면 <동작>메뉴에서 <하드웨어 변경사항 검색>을 눌러, COM1 포트를 추가하시기 바랍니다.

끝으로 CUBLOC STUDIO 의 “SETUP”메뉴에서 “PC 인터페이스 설정”에서 COM1 포트로 셋팅해주면 CUBLOC 통신포트 설정이 완료된 것입니다.



아래 부분의 Wait Time 은 Window 98 유저를 위한 셋팅입니다. XP, 2000 유저는 조정할 필요가 없습니다.

Window 98 은 컴퓨터의 속도에 따라서, 다운로드 시 에러가 발생하는 경우가 있습니다. Wait Time 속도를 조정해 가면서, 한번씩 다운로드를 시도해보십시오.

그러면, 동작되는 레벨이 있을 것입니다. 그곳으로 한번만 셋팅되어 있으면, 이후로는 조정할 필요가 없습니다.

RS232 포트가 없는 PC 또는 노트북에서 사용방법

노트북이라면 PCMCIA-TO-RS232 카드를 권장합니다. USB-TO-RS232 보다 안정적이고 빠른 다운로드 속도를 지원합니다. (강원전자 NETMATE C-191)



최근에는 PC 에도 RS232 포트가 없는 경우가 있습니다. 하지만 마더보드에는 최소한 COM 포트 1 개 정도는 구비하고 있는 것이 보통입니다. 다만 외부로 연결되어 있지 않은 경우입니다. 해당 마더보드의 매뉴얼을 잘 참조하시어, 외부로 RS232 포트를 뽑아내어 사용하시는 것이 가장 좋습니다. 아래와 같은 PCI 슬롯방식 RS232 확장카드를 사용하는 방법도 있습니다.



*부득이한 경우가 아니라면 USB-TO-RS232 사용을 권장하지 않습니다. USB-TO-RS232 케이블은 내부에 있는 마이컴에서 버퍼링을 한뒤 재출력하는 방식이라 속도가 느리고, 동작이 불안정합니다.

*주의사항 : RS232 케이블의 길이는 최대 2 미터까지만 사용할 수 있습니다. 그 이상의 거리에서는 다운로드 및 디버깅 / 모니터링의 정확한 동작을 보증할 수 없습니다.

다운로드 시 발생하는 에러메시지

1. RS232 통신에러입니다. 케이블상태나 전원상태, PC의 통신 설정상태를 점검해보십시오.

케이블을 연결하지 않았거나, CUBLOC 코어모듈에 전원이 공급되지 않았을 경우, 또는 PC 상의 통신포트 설정에 이상이 있는 경우 발생하는 에러 메시지입니다. 펌웨어가 제대로 다운로드 안되었을때도 이 메시지가 나옵니다. SETUP 메뉴에 있는 펌웨어 다운로드를 눌러서, 펌웨어를 다운로드한뒤 사용하기 바랍니다.

2. Ladder 에서 END 를 찾을 수 없습니다.

소스파일 자체를 오픈하지 않았거나, LADDER 부에서 END 명령을 작성하지 않은 경우입니다. 이 경우는 통신문제와는 관계없는 경우입니다.

3. Ladder 컴파일 에러

LADDER 부에서 문법상의 오류 및 회로 결선상의 오류가 있는 경우입니다. 이 경우는 통신문제와는 관계없는 경우입니다.

4. 존재하지 않는 RS232 포트입니다.

PC 상의 COM 포트 셋업에 문제가 있는 경우이거나, 존재하지 않는 COM 포트를 선택한 경우입니다. 앞에서 설명한 BIOS 셋업상태와 윈도우 시스템에서의 COM 포트 설정상태를 점검해 보아야 합니다. 앞에서 설명한대로 설정했는데도 불구하고 똑 같은 에러가 발생한다면 PC 전문가의 도움을 받아야 합니다. 이 에러는 CUBLOC STUDIO 에서 COM 포트를 찾지 못하는 문제이므로, CUBLOC 하드웨어 상태와는 무관한 문제입니다.

5. CB405 모듈이 아닙니다. CONST DEVICE 문에서 선언한 모듈과 다른 모듈이 연결되어 있습니다.

CONST DEVICE 선언문을 사용해서 어떤 큐블록 코어모듈을 사용 중인지를 CUBLOC STUDIO 가 알 수 있도록 해주어야 합니다. <사용 예 : CONST DEVICE = CB405 >

6. 체크섬 불일치 발생: 플래쉬 메모리의 일부가 손상되었습니다.

큐블록 코어모듈 내부에 있는 플래쉬 메모리의 일부가 손상되거나, 다운로드 중 전원의 불안이나 기타요인 등으로 인해, 제대로 플래쉬에 기입되지 못한 경우입니다. 이 경우 다시 다운로드 하면 제대로 다운로드 될 수도 있습니다. 반복해서 같은 에러메시지가 발생한다면 메뉴상의 “펌웨어 다운로드”를 실행한 뒤 다시 해보시고, 그래도 안 된다면, 큐블록 코어모듈을 다른 것으로 교체하여 사용하기 바랍니다. 큐블록 내부에 있는 플래쉬 메모리는 최대 10 만번까지 다운로드 가능하며, 그 이상 사용하였을 경우, 특정 메모리 블록이 손상될 수 있습니다.

7. 액세스 코드가 잘못되었습니다.

큐블록 내부에 있는 “슈퍼바이저”칩이 손상된 경우입니다. 이 경우에도 펌웨어 다운로드를 다시 해보시고, 그래도 안 된다면 큐블록 코어모듈을 다른 것으로 교체해야 합니다.

부 록

ActiveSync 설치하기

ActiveSync 는 PC 와 IntelliLCD 를 연결하여 프로그램입니다. 액티브싱크를 통해서 배경화면 (Bmp 파일등)을 IntelliLCD 의 폴더에 저장할 수 있습니다.

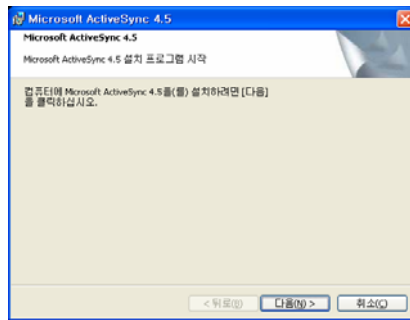
ActiveSync 는 무료이며, 다음 사이트에서 다운로드 받으실 수 있습니다. (URL 은 변경될 수 있습니다.)

Web 다운로드 주소 :

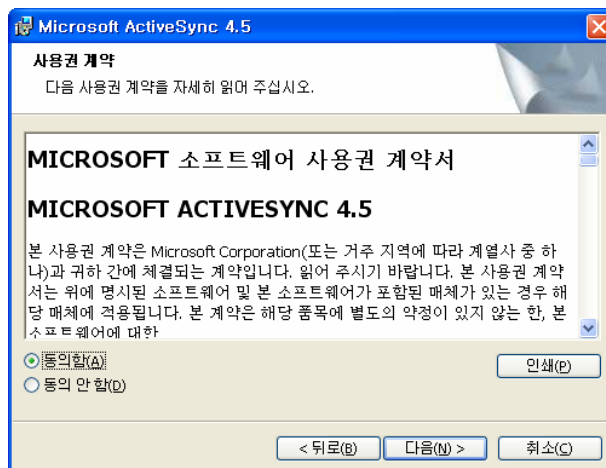
“<http://www.microsoft.com/korea/windowsmobile/activesync/activesync45.msp>”

만약 마이크로소프트사의 웹사이트에서 다운받는 것이 번거로우시다면, NAVER 검색 등을 통해서도 쉽게 구하실 수 있습니다. ActiveSync 4.5 버전을 기본으로 해서 설명 드리겠습니다.

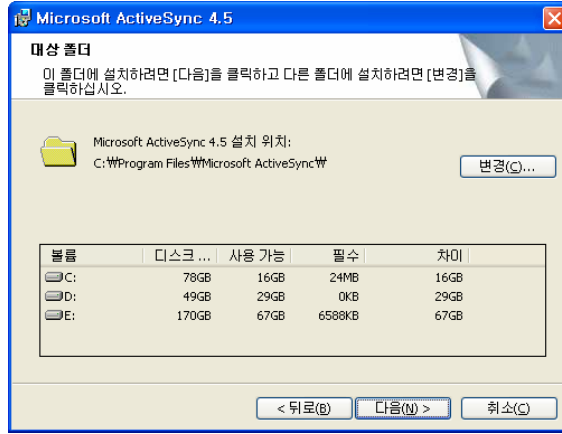
1. ActiveSync 설치 하기 전까지는 CUWIN 과 PC 를 연결하지 않습니다.
2. 다운 받으신 파일을 더블 클릭하시면 아래와 같이 진행 됩니다. 다음을 클릭합니다.



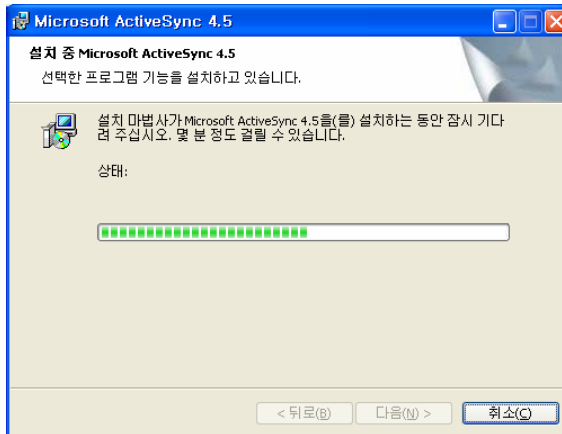
3. 프로그램 설치준비가 끝나면, 아래와 같이 사용권 계약 창이 뜨게 됩니다. 동의 함을 선택합니다. 다음을 누릅니다.



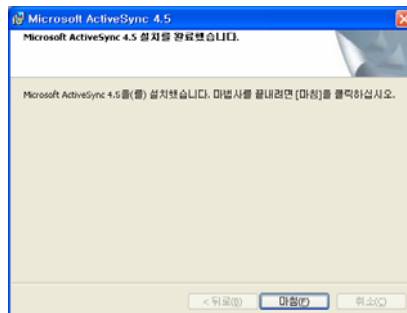
4. 사용자 정보를 입력 합니다. 다음을 클릭 합니다.
5. 다음을 누르게 되면 다음과 같이 나타나는데, 설치할 폴더를 결정할 수 있습니다. (아무 설정 없이 설치할 것을 권장 합니다.)



6. 역시 다음을 누르면, 설치되는 진행 상황 다음과 같이 나타나면서 설치가 진행 됩니다.

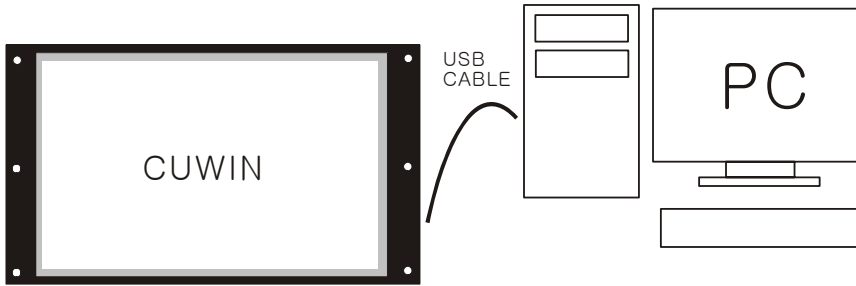


7. 설치가 끝나면 다음 같은 화면을 볼 수 있습니다.

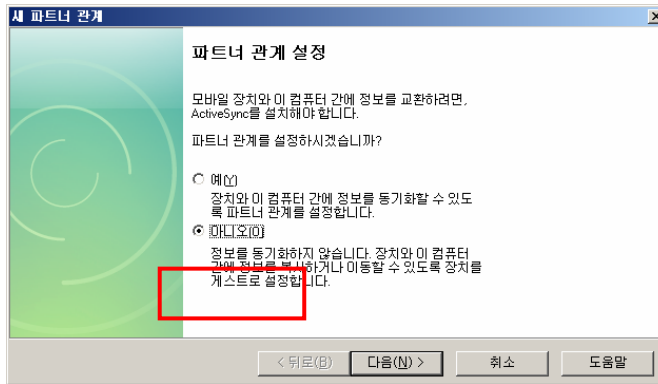


ActiveSync 실행

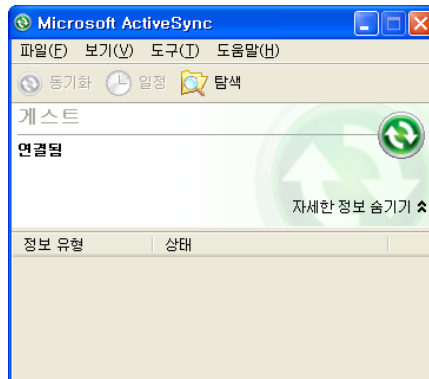
1. CUWIN 와 PC 를 제공된 USB Cable 로 서로 연결합니다.



2. 연결이 정상적으로 이루어지면 다음과 같이 새 파트너 관계 메시지 창이 뜨게 됩니다.
 “아니오”를 선택합니다. (새 파트너 관계설정 창이 나타나지 않을 수도 있습니다.)

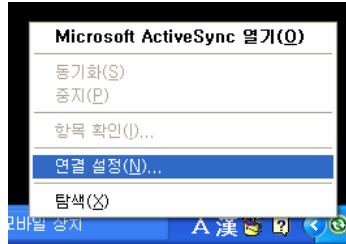


3. 연결이 완료되면 다음과 같은 창이 뜨게 됩니다.

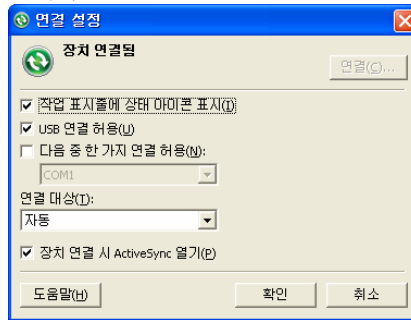


4. 이것으로 CUWIN 와 PC 간의 연결이 완료 된 것입니다. 연결이 된 상태에서는 PC 에서 CUWIN 의 저장장치의 내용을 확인할 수 있습니다. 만약 위와 같이 했는데도 연결이 안될 경우 다음과 같이 확인하세요.

- ActiveSync 트레이 아이콘 위에서 오른쪽버튼을 클릭합니다. 연결설정을 선택하세요.

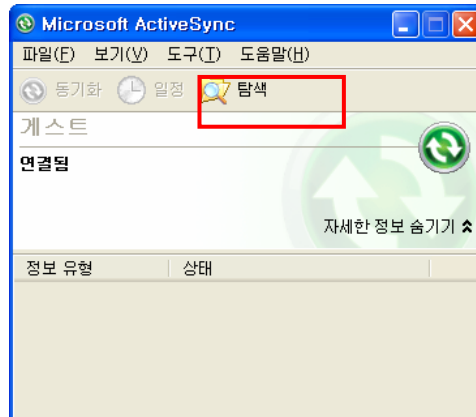


- 아래와 같이 설정되어있는지 확인 하세요.



- 또는 USB 케이블을 재 연결 하시기 바랍니다.

연결된 상태에서 탐색 아이콘을 클릭 하면, IntelliLCD 의 탐색창이 표시됩니다.



<THE END>