

LKV-408

SW 사용자 매뉴얼

Board Rev. 2.4
BSP Version 2.2.2

2009년 02월 11일



알림

여기에 실린 내용은 제품의 성능 향상과 신뢰도의 증대를 위하여 예고없이 변경될 수도 있습니다.

여기에 실린 내용의 일부라도 엘케이일레브의 사전 허락없이 어떠한 유형의 매체에 복사되거나 저장될 수 없으며 전기적, 기계적, 광학적, 화학적인 어떤 방법으로도 전송될 수 없습니다.

㈜엘케이일레브

경기도 성남시 중원구 상대원동 191-1 SKn테크노파크 메가동 1306호

LKV-408은 ㈜엘케이일레브의 등록상표입니다.

차 례

1. INTRODUCTION	6
1.1. LKV-408 보드의 소개	6
1.2. 본 문서의 내용	6
2. BOOT PARAMETER	7
2.1. BOOT PARAMETER 정보	7
2.2. BOOT PARAMETER 입력 방법	15
2.3. BOOTPARAMETER FLAG 관련 추가정보 (2009/02/18 추가됨).....	16
3. LKV-408 CONTROL METHOD	18
3.1. LKV-408 MEMORY MAP	18
3.1.1. FLASH MEMORY	18
3.1.1.1. Flash Memory 기능	18
3.1.1.2. Flash Memory Control Fucntion	19
3.1.1.3. Flash Memory 관련 추가정보 (2009/02/18 추가됨).....	20
3.1.2. NVRAM MEMORY	21
3.1.2.1. NVRAM의 기능	21
3.1.2.2. NVRAM Register Map	21
3.1.2.3. NVRAM Control Function	22
3.1.4. SRAM MEMORY.....	23
3.1.4.1. SRAM의 기능	23
3.1.5. SERIAL EEPROM(93c46)	23
3.2. THE CONTENTS OF A LKV-408 B/D I/O CONTROL	24
3.2.1. WATCHDOG TIME OUT RESET CONTROL	24
3.2.1.1. Watchdog time Reset 기능	24
3.2.1.2. WatchDog time control Function	24
3.2.1.3. WatchDog time out Test Example	24
3.2.2. DIGITAL IN/OUT CONTROL	25
3.2.2.1. Function Description.....	25
3.2.3 DIP SWITCH CONTROL	25
3.2.3.1. Function Description.....	25

3.2.3.2. Dip Switch Test Example	26
3.2.4. ENCODER SWITCH CONTROL	26
3.2.4.1. Function Description	26
3.2.5. LEDTOGGLE (2009/02/19 추가됨)	27
3.2.5.1. Function Description	27
4. LKV-408 SERIAL CONTROL METHOD & USER API	28
4.1. LKV-408 B/D SERIAL DRIVER INITIALIZATION	28
4.2. VxWORKS SERIAL I/O SYSTEM	29
4.2.1. OPEN ()	29
4.2.1.2. CLOSE ()	29
4.2.3. READ ()	30
4.2.4. WRITE ()	30
4.2.5. IOCTL ()	31
4.2.6 HDLC I/O CONTROL METHOD	32
4.2.6.1. HDLC Frame	32
4.3. VxWORKS SERIAL I/O SYSTEM	34
5. 주의 사항	35

그림 목차

그림 1. LKV-408 Boot Parameter 예제	7
그림 2. LKV-408 부팅 화면	15
그림 3. LKV-408 네트워크 부팅 예제	16
그림 4. Flash 메모리 정보 보기	21
그림 5. LKV-408 B/D Serial Driver Initial Sequence	28
그림 6. HDLC Frame Structure	32
그림 7. LKV-408 B/D HDLC Serial Driver Initial Sequence	33

표 목차

표 1. LKV-408 Board IP Encoder Switch Setting Value	7
표 2. Boot Parameter 에 따른 부팅 방법	17
표 3. LKV-408 Memory Map	18
표 4. DS1647(RTC) Register Map	22

1. Introduction

1.1. LKV-408 보드의 소개

LKV-408 B/D는 VMEbus용 Master/Slave B/D로써 Motorola사의 PowerPC core가 내장된 embedded communication processor인 MPC860 CPU를 사용하여 고성능의 다양한 기능을 구현한 B/D입니다. LKV-408의 가장 큰 특징은 Manchester code를 지원하는 RS485 port를 내장하고 있다는 것입니다. 또한 Ethernet 2port, RS232 4port를 지원하여 여타 추가의 통신보드를 사용하지 않고도 다양한 통신을 지원하도록 하였으며 사용자의 편의성을 고려하여 Front panel과 P2 connector 모두에서 통신이 가능하도록 하였습니다.

VMEbus에서는 system controller 기능과 Master기능, Slave기능, Interrupt handler 기능을 가지고 있으며 B/D내에는 32 MB SDRAM, 512KB EPROM, 512 KB RTC/NVRAM, 4 MB Flash Memory, 2MB SRAM을 가지고 있어서 다양한 application의 사용이 가능하도록 설계되어 있습니다.

1.2. 본 문서의 내용

LKV-408 보드는 보드 내에 32MByte의 SDRAM, 4MB의 Flash Memory, Battery Backup 된 2MB의 SRAM, 512KB의 RTC/NVRAM, 1Kbyte의 EEPROM이 내장되어 있다. 또한 Serial Interface와 VMEBus Interface가 구현되어 있어 Master/Slave 보드로 사용 가능합니다. 따라서 본 문서에서는 이러한 보드의 구성에 의해 1장에서는 LKV-408 보드의 소개 및 본 문서의 구성에 대한 내용이 수록 되어있으며, 2장에서는 보드 사용시 처음에 셋팅 되어야 할 Boot Parameter 에 관해 자세한 설명이 되어 있다. 3장에서는 LKV-408 보드에서 지원되는 다양한 메모리 종류에 따른 컨트롤 방법 및 예가 주어지며, 그 밖의 기능의 IO(Digital I/O, DipSW, WatchDog, EncoderSW) 컨트롤 방법에 대해 설명한다. 4장에서는 Serial Port에 관한 사용법 및 셋팅 방법에 대해 소개하며, 그에 따른 예가 주어진다.

2. Boot Parameter

2.1. Boot Parameter 정보

boot device	: cpm
unit number	: 0
processor number	: 0
host name	: LKV-408
file name	: C:\Wftproot\WLVK-408\WvxWorks
inet on ethernet (e)	: 220.76.45.178:ffffff00
host inet (h)	: 220.76.45.242
user (u)	: hjahn
ftp password (pw)	: 1234
flags (f)	: 0x0
other (o)	: 10.0.0.178:ffffff00

그림 1. LKV-408 Boot Parameter 예제

위의 내용은 Boot Parameter VxWorks BSP를 이용하여 부팅하기 위한 정보들이다. 여기서 Boot device는 Ethernet 드라이버의 name을 가리키며 보드나 드라이버에 따라서 달라질 수 있다. [그림 2]는 VxWorks BSP를 이용한 부팅 화면이다. BSP는 FTP를 이용하여 host로부터 OS+Application image를 다운로드 받으며, 후에 이를 실행한다. 위의 “inet on ethernet (e)”의 IP중 Net Work IP의 (“XXX.XXX.XXX.XXX”) 마지막 Address는 Encoder S/W 에 의해 셋팅 된 값이 등록된다. “other(o)”에 설정된 Ethernet 1의 IP되 “10.0.0.XXX” 까지만 입력되며, “XXX”는 Encoder S/W value에 의해서 셋팅 된다. 위의 경우는 현재 Encoder S/W가 0xB2(B10110010)으로 셋팅 되어 있다. Encoder S/W Value는 [표 1]과 같습니다.

표 1. LKV-408 Board IP Encoder Switch Setting Value

DECIMAL	HEX	ENCODER S/W IP0	ENCODER S/W IP1
000	0x00	0	0
001	0x01	0	1
002	0x02	0	2
003	0x03	0	3
004	0x04	0	4
005	0x05	0	5
006	0x06	0	6
007	0x07	0	7
008	0x08	0	8

009	0x09	0	9
010	0x0a	0	A
011	0x0b	0	B
012	0x0c	0	C
013	0x0d	0	D
014	0x0e	0	E
015	0x0f	0	F
016	0x10	1	0
017	0x11	1	1
018	0x12	1	2
019	0x13	1	3
020	0x14	1	4
021	0x15	1	5
022	0x16	1	6
023	0x17	1	7
024	0x18	1	8
025	0x19	1	9
026	0x1a	1	A
027	0x1b	1	B
028	0x1c	1	C
029	0x1d	1	D
030	0x1e	1	E
031	0x1f	1	F
032	0x20	2	0
033	0x21	2	1
034	0x22	2	2
035	0x23	2	3
036	0x24	2	4
037	0x25	2	5
037	0x26	2	6
039	0x27	2	7
040	0x28	2	8
041	0x29	2	9
042	0x2a	2	A
043	0x2b	2	B
044	0x2c	2	C
045	0x2d	2	D

046	0x2e	2	E
047	0x2f	2	F
048	0x30	3	0
049	0x31	3	1
050	0x32	3	2
051	0x33	3	3
052	0x34	3	4
053	0x35	3	5
054	0x36	3	6
055	0x37	3	7
056	0x38	3	8
057	0x39	3	9
058	0x3a	3	A
059	0x3b	3	B
060	0x3c	3	C
061	0x3d	3	D
062	0x3e	3	E
063	0x3f	3	F
064	0x40	4	0
065	0x41	4	1
066	0x42	4	2
067	0x43	4	3
068	0x44	4	4
069	0x45	4	5
070	0x46	4	6
071	0x47	4	7
072	0x48	4	8
073	0x49	4	9
074	0x4a	4	A
075	0x4b	4	B
076	0x4c	4	C
077	0x4d	4	D
078	0x4e	4	E
079	0x4f	4	F
080	0x50	5	0
081	0x51	5	1
082	0x52	5	2

083	0x53	5	3
084	0x54	5	4
085	0x55	5	5
086	0x56	5	6
087	0x57	5	7
088	0x58	5	8
089	0x59	5	9
090	0x5a	5	A
091	0x5b	5	B
092	0x5c	5	C
093	0x5d	5	D
094	0x5e	5	E
095	0x5f	5	F
096	0x60	6	0
097	0x61	6	1
098	0x62	6	2
099	0x63	6	3
100	0x64	6	4
101	0x65	6	5
102	0x66	6	6
103	0x67	6	7
104	0x68	6	8
105	0x69	6	9
106	0x6a	6	A
107	0x6b	6	B
108	0x6c	6	C
109	0x6d	6	D
110	0x6e	6	E
111	0x6f	6	F
112	0x70	7	0
113	0x71	7	1
114	0x72	7	2
115	0x73	7	3
116	0x74	7	4
117	0x75	7	5
118	0x76	7	6
119	0x77	7	7

120	0x78	7	8
121	0x79	7	9
122	0x7a	7	A
123	0x7b	7	B
124	0x7c	7	C
125	0x7d	7	D
126	0x7e	7	E
127	0x7f	7	F
128	0x80	8	0
129	0x81	8	1
130	0x82	8	2
131	0x83	8	3
132	0x84	8	4
133	0x85	8	5
134	0x86	8	6
135	0x87	8	7
136	0x88	8	8
137	0x89	8	9
137	0x8a	8	A
139	0x8b	8	B
140	0x8c	8	C
141	0x8d	8	D
142	0x8e	8	E
143	0x8f	8	F
144	0x90	9	0
145	0x91	9	1
146	0x92	9	2
147	0x93	9	3
148	0x94	9	4
149	0x95	9	5
150	0x96	9	6
151	0x97	9	7
152	0x98	9	8
153	0x99	9	9
154	0x9a	9	A
155	0x9b	9	B
156	0x9c	9	C

157	0x9d	9	D
158	0x9e	9	E
159	0x9f	9	F
160	0xa0	A	0
161	0xa1	A	1
162	0xa2	A	2
163	0xa3	A	3
164	0xa4	A	4
165	0xa5	A	5
166	0xa6	A	6
167	0xa7	A	7
168	0xa8	A	8
169	0xa9	A	9
170	0xaa	A	A
171	0xab	A	B
172	0xac	A	C
173	0xad	A	D
174	0xae	A	E
175	0xaf	A	F
176	0xb0	B	0
177	0xb1	B	1
178	0xb2	B	2
179	0xb3	B	3
180	0xb4	B	4
181	0xb5	B	5
182	0xb6	B	6
183	0xb7	B	7
184	0xb8	B	8
185	0xb9	B	9
186	0xba	B	A
187	0xbb	B	B
188	0xbc	B	C
189	0xbd	B	D
190	0xbe	B	E
191	0xbf	B	F
192	0xc0	C	0
193	0xc1	C	1

194	0xc2	C	2
195	0xc3	C	3
196	0xc4	C	4
197	0xc5	C	5
198	0xc6	C	6
199	0xc7	C	7
200	0xc8	C	8
201	0xc9	C	9
202	0xca	C	A
203	0xcb	C	B
204	0xcc	C	C
205	0xcd	C	D
206	0xce	C	E
207	0xcf	C	F
208	0xd0	D	0
209	0xd1	D	1
210	0xd2	D	2
211	0xd3	D	3
212	0xd4	D	4
213	0xd5	D	5
214	0xd6	D	6
215	0xd7	D	7
216	0xd8	D	8
217	0xd9	D	9
218	0xda	D	A
219	0xdb	D	B
220	0xdc	D	C
221	0xdd	D	D
222	0xde	D	E
223	0xdf	D	F
224	0xe0	E	0
225	0xe1	E	1
226	0xe2	E	2
227	0xe3	E	3
228	0xe4	E	4
229	0xe5	E	5
230	0xe6	E	6

231	0xe7	E	7
232	0xe8	E	8
233	0xe9	E	9
234	0xea	E	A
235	0xeb	E	B
236	0xec	E	C
237	0xed	E	D
237	0xee	E	E
239	0xef	E	F
240	0xf0	F	0
241	0xf1	F	1
242	0xf2	F	2
243	0xf3	F	3
244	0xf4	F	4
245	0xf5	F	5
246	0xf6	F	6
247	0xf7	F	7
248	0xf8	F	8
249	0xf9	F	9
250	0xfa	F	A
251	0xfb	F	B
252	0xfc	F	C
253	0xfd	F	D
254	0xfe	F	E
255	0xff	F	F

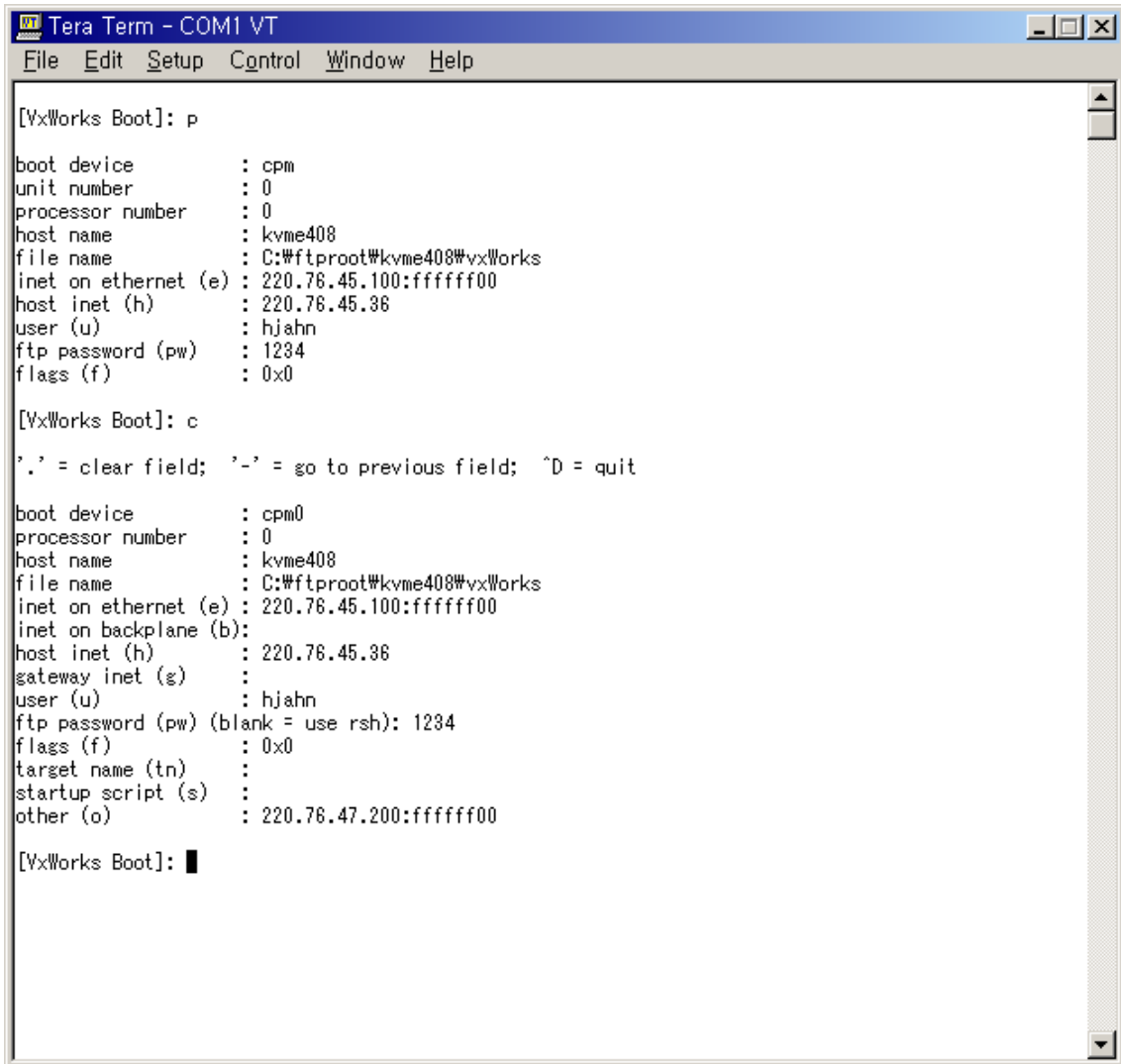


그림 2. LKV-408 부팅 화면

2.2. Boot Parameter 입력 방법

VxWorks Boot prompt([VxWorks Boot]:) 상에서 'p'를 입력하면 [그림 2]과 같이 Boot Parameter에 관한 정보를 볼 수 있다. Prompt 상에서 'c'를 입력하면 Boot Parameter 에 대한 정보를 user에 맞게 Setting 가능하도록 되어 있다. 다음 [그림 3]은 실제 vxWorks image가 boot된 후에 target Shell이 실행된 모습이다. Parameter의 입력이 끝난 후 Prompt 상에서 '@'를 입력하면 Ethernet을 통해 부팅이 실행되는 것을 확인할 수 있다.

```

Tera Term - COM1 VT
File Edit Setup Control Window Help
[VxWorks Boot]: @
boot device      : cpm
unit number     : 0
processor number : 0
host name       : kvme408
file name       : C:\ftproot\kvme408\vxWorks
inet on ethernet (e) : 220.76.45.100:ffffff00
host inet (h)   : 220.76.45.36
user (u)        : hjahn
ftp password (pw) : 1234
flags (f)       : 0x0
other (o)       : 220.76.47.200:ffffff00

-----Ethernet Address -----
NV enet = 220.76.45.100:ffffff00
nvIp == 220.76.45.100
Encoder rotary switch value = 0x64[100]
changing IP : 220.76.45.100
net mask = 0xffffffff00
final address == 220.76.45.100:ffffff00
struct ead : 220.76.45.100:ffffff00

boot device      : cpm
unit number     : 0
processor number : 0
host name       : kvme408
file name       : C:\ftproot\kvme408\vxWorks
inet on ethernet (e) : 220.76.45.100:ffffff00
host inet (h)   : 220.76.45.36
user (u)        : hjahn
ftp password (pw) : 1234
flags (f)       : 0x0
other (o)       : 220.76.47.200:ffffff00

Attached TCP/IP interface to cpm0.
Attaching network interface lo0... done.
Loading... 959808
Starting at 0x10000...

Target Name: vxTarget
Attached TCP/IP interface to cpm unit 0
Attaching network interface lo0... done.
NFS client support not included.

-----Other Address -----
NV enet = 220.76.47.200:ffffff00
nvIp == 220.76.47.200
Encoder rotary switch value = 0x64[100]
changing IP : 220.76.47.100
net mask = 0xffffffff00
final address == 220.76.47.100:ffffff00
struct ead : 220.76.45.100:ffffff00

boot device      : cpm
unit number     : 0
processor number : 0
host name       : kvme408
file name       : C:\ftproot\kvme408\vxWorks
inet on ethernet (e) : 220.76.45.100:ffffff00
host inet (h)   : 220.76.45.36
user (u)        : hjahn
ftp password (pw) : 1234
flags (f)       : 0x0
other (o)       : 220.76.47.100:ffffff00

Attached TCP/IP interface to cpm1.
Loading symbol table from kvme408:C:\ftproot\kvme408\vxWorks.sym ...done
    
```

그림 3. LKV-408 네트워크 부팅 예제

2.3. bootParameter flag 관련 추가정보 (2009/02/18 추가됨)

LS 산전의 요구사항에 따라서 기존의 LKV-408 보드의 부팅 방법이 변경되었다.

기존 LKV-408 보드는 J12 섀트를 EPROM 또는 FLASH 위치로 변경하여 EPROM, FLASH 부팅을

변경할 수 있었지만 LKV-408 V2.2.0 부터는 **선택의 위치는 항상 EPROM 으로 고정하고** bootParameter의 값을 변경함에 따라서 EPROM, FLASH 부팅이 지원된다. 다음은 booting 방법에 따른 bootParameter의 값을 나타내는 테이블이다.

표 2. Boot Parameter에 따른 부팅 방법

BootParameter	값	부팅 방법
Flags	0x0	EPROM 으로 network 부팅 (counting)
Flags	0x8	EPROM 으로 network 부팅 (no counting)
Flags	0x1000	Flash 부팅 (counting)
Flags	0x1008	Flash 부팅 (no counting)

bootParameter flag 값은 아래와 같다.

- ① 0x0, 0x8 : 기존에 사용하던 방식 그대로 사용
- ② 0x1000, 0x1008 : booting 한 후, standalone image가 저장되어 있는 Flash의 0xFA000100 번지로 바로 jump 하므로 flash에 Standalone image를 makeFlashBoot() 함수를 사용하여 download 한 후 실행해야 정상적인 booting이 가능하다.

3. LKV-408 Control Method

3.1. LKV-408 Memory Map

표 3. LKV-408 Memory Map

Address Range	Size(Byte)	Description
0x00000000 ~ 0x01FFFFFF	32M	SDRAM
0x10000000 ~ 0xEFFFFFFF	3.6G	VMEBus Extended Address
0xF0000000 ~ 0xF0FFFFFF	16M	VMEBus Standard Address
0xF1000000 ~ 0xF11FFFFF	2M	SRAM
0xF2000000 ~ 0xF207FFFF	512K	NVRAM
0xF3000000	-	RESERVED
0xF4000000 ~ 0xF400FFFF	64K	VMEBus Short Address
0xF5000000	-	FAIL LED OFF
0xF6000000	-	Watch dog timer Enable
0xF7000000	-	FAIL LED ON
0xF800000X	-	VMEBus Interrupt Acknowledge
0xF9000000	-	DIP Switch & DI Read
0xFA000000 ~ 0xFA3FFFFFF	4M	Flash Memory
0xFB000000	-	VMEBus Interrupt Vector register
0xFC000000	-	VMEBus Interrupt Request register
0xFD000000	-	VMEBus AM Code 2 generate register
0xFE000000	-	RESERVED
0xFF000000	-	IMMR
0xFFF00000 ~ 0xFFFFFFFF	1M	EPROM

3.1.1. Flash Memory

3.1.1.1. Flash Memory 기능

LKV-408 보드의 Flash Memory는 총 4Mbyte의 용량을 가지고 있으며, 각 블록은 128Kbyte의 용량을 가지며 32블록으로 구성되어 있다. 또한 LKV-408 보드는 Flash Memory를 이용하여 Flash Boot 기능으로 사용할 수도 있다.

3.1.1.2. Flash Memory Control Function

1) makeFlashBoot();

Flash에 bootrom.bin 파일을 플래쉬 메모리에 다운로드하는 함수이다.
LKV-408 보드의 Flash Booting을 하기 위해서는 아래와 같은 절차를 수행해야 한다.

순서 1) VxWorks StandAlone image를 만들어야 하며, 명령은 다음과 같다. **Config.h** 파일에 있는 “**#define FLASH_BOOT_MODE**” 를 Define 한 후 Dos 상에서 BSP 파일이 있는 디렉토리에서 다음을 실행한다.

```
Ex) make vxWorks.st_Rom.hex
```

순서 2) Dos상에서 "WWINDBASEWhostWx86-win32WbinWtorVars.bat"를 실행하여 Path를 잡아준다.

순서 3) 아래와 같은 명령으로 kv408.bin 파일을 생성한다.

```
Ex) elfToBin <inFile> outfile_bsd
    elfToBin <vxWorks.st_rom> kv408.bin
```

실행 시 elf file 이 Binary 파일 형식으로 바뀜. 도스 명령의 가운데 인자는 롬파일을 만들 때 생성된 vxWorks standalone image 파일이다.

순서 4) 생성된 Binary 파일을 Target Board에서 Load 할 수 있는 위치로 옮긴다.

순서 5) Console 상에서 “ls” 명령시 다음과 같이 binary 파일이 보이는지 확인한다

```
-> ls
kv408.bin
```

순서 6) makeFlashBoot 함수를 이용하여 Binary 파일을 Flash Memory 로 기록 한다.

순서 7) 아래의 makeFlashBoot Test와 같은 메시지가 나오면 정상적으로 플래쉬 메모리에 기록된다. 아래 상태에서 롬의 우측 하단부에 있는 점퍼(J13) 2개를 FLASH MODE로 셋팅, 점퍼(J12)도 FLSAH MODE로 셋팅 하고 Board를 Rebooting하면 Flash memory 부팅이 이루어진다.

2) makeFlashBoot Test

```

-> makeFlashBoot("kv408.bin")
Start binary file to Flash
Load File Name : kv408.bin - size : 479092 bytes
Open & read binary file OK
sysFlashSet => offset : 0x300100, strLen : 479092, flash_size : 0x400000
Flash Sector [Block] Erase OK!
FlashWrite ==> pFA : 0xfa300100, end : 0xfa375074, size : 479092[479092]
Programming.. 478000 Bytes. adrs : 0xfa374c2e
Start Adrs : 0xfa300100, End Adrs : 0xfa375074, size : 479092 bytes
Flash Write 479092 byte OK!!
value = 0 = 0x0
->
    
```

3.1.1.3. Flash Memory 관련 추가정보 (2009/02/18 추가됨)

LS 산전의 요구사항이 추가됨에 따라서 Flash memory의 1M 영역만 standalone image를 저장하고, 나머지 영역은 사용자의 필요에 의한 DB처럼 사용할 수 있도록 구조가 변경됐다. Flash의 image와 사용자 DB 영역은 다음과 같다.

```

Flash image size : 0xFA000000 ~ 0xFA100000
User DB area      : 0xFA100000 ~ 0xFA3E0000
    
```

LKV-408 BSP V2.2.1부터는 makeFlashBoot() 함수를 사용하여 flash에 Standalone image를 write 하면 0xFA000000부터 최대 0xFA100000 까지의 영역만 사용한다.

PrintFlashInfo() 함수를 호출하면 flash의 정보와 Block 별로 data가 write 된 정보가 다음 [그림 4]와 같이 테이블 형식으로 표시되므로 참고하면 된다.

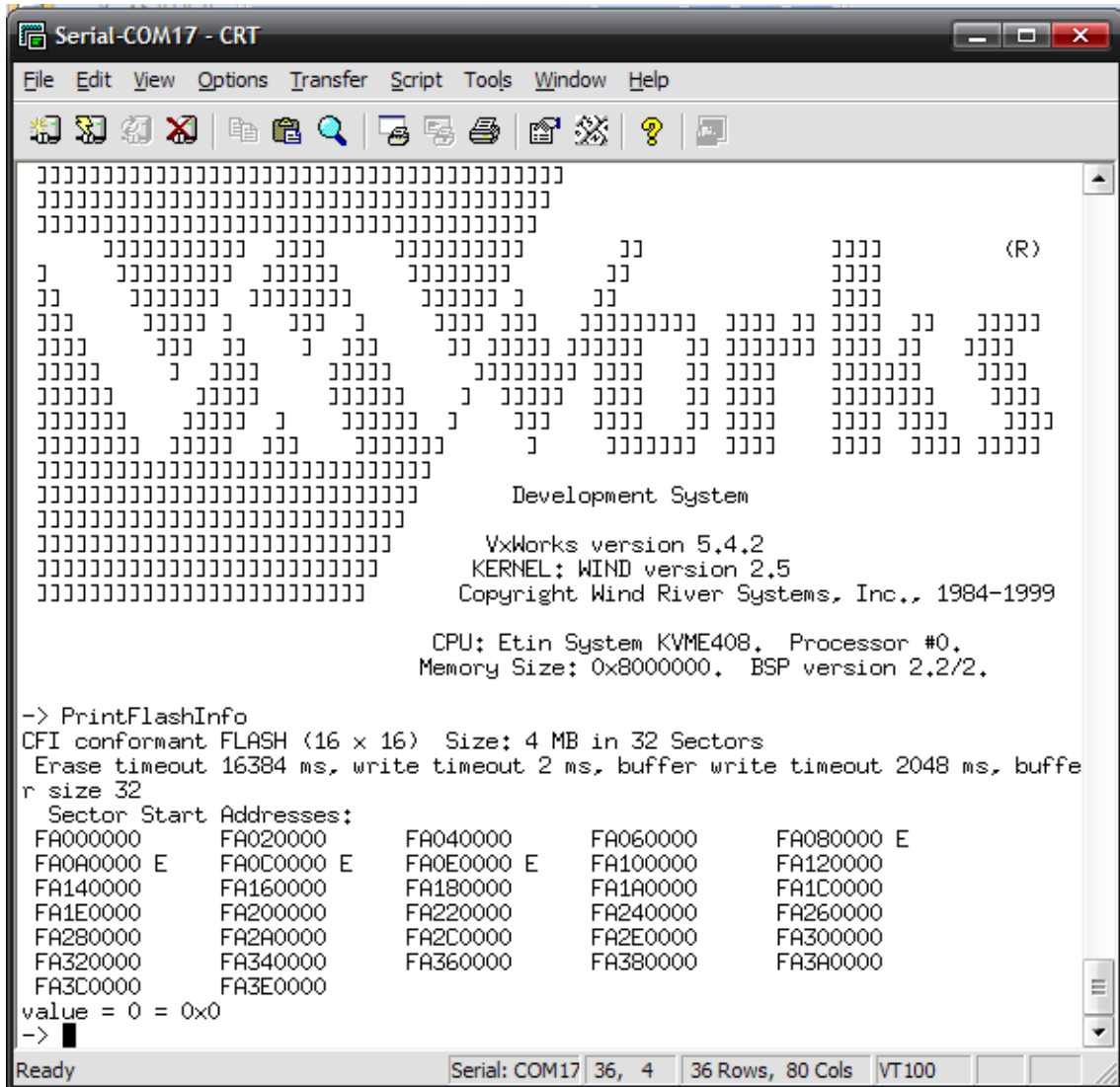


그림 4. Flash 메모리 정보 보기

3.1.2. NVRAM Memory

3.1.2.1. NVRAM의 기능

NVRAM은 Boot Parameter의 저장을 주 목적으로 사용하며, Real Time Clock을 지원하는 Timer 영역을 제공하고 있다.

LKV-408 보드에서는 Dallas 사의 DS1647을 사용하며, 메모리는 512 Kbyte, access time은 120ns ~ 150ns 이하의 것을 사용한다.

3.1.2.2. NVRAM Register Map

NVRAM 의 Register에 시간정보를 셋팅 할 수 있으며, 시간 정보는 [표 4]와 같다.

표 4. DS1647(RTC) Register Map

ADDRESS	DATA								FUNCTION	
	B7	B6	B5	B4	B3	B2	B1	B0		
7FFFF	-	-	-	-	-	-	-	-	YEAR	00-99
7FFFE	X	X	X	-	-	-	-	-	MONTH	01-12
7FFFD	X	X	-	-	-	-	-	-	DATE	01-31
7FFFC	X	FT	X	X	X	-	-	-	DAY	01-07
7FFFB	X	X	-	-	-	-	-	-	HOUR	00-23
7FFFA	X	-	-	-	-	-	-	-	MINUTES	00-59
7FFF9	ST	-	-	-	-	-	-	-	SECONDS	00-59
7FFF8	W	R	X	X	X	X	X	X	CONTROL	A

ST : Stop Bit **R** : Read Bit **FT** : Frequency Test
W : Write Bit **X** : Unused

3.1.2.3. NVRAM Control Function

1) STATUS date (char *p)

RTC의 시간 정보를 설정하는 함수이다. 설정값은 순서는 MM DD YY HH MM SS의 순서로 설정된다.

Ex) date("051120121010")

MM : Month
 DD : Day
 YY : Year
 HH : Hour
 MM : Minutes
 SS : Second

2) void date (NULL)

현재 NVRAM에 저장되어 있는 시간 정보를 출력해 주는 함수이다.

->date
SUN NOV 20 12:11:14 2005

3) STATUS sysRtcGet(struct tm *tp)

NVRAM(DS1647)의 Time 을 가져오는 함수이다.

4) STATUS sysRtcSet(const struct tm *tp)

NVRAM(DS1647)에 Time Setting 하는 함수이다.

5) calDay function (2009/02/19 추가됨)

LS 산전에서 년, 월, 일을 입력하면 해당하는 일주일의 요일 숫자값을 리턴하는 함수를 추가해달라고 하여 KVME080 BSP 에서 사용하는 calDay 함수를 추가함.

3.1.4. SRAM Memory

3.1.4.1. SRAM의 기능

2 Mbyte의 low power SRAM이 장착되어 있으며, SRAM은 MPC860 또는 VMEbus에서 access 가능하다. SRAM의 access time은 70 ns이다. 또한 SRAM에는 3V의 Lithium battery가 연결되어 있어서 전원이 off 된 상태에서도 데이터를 유지할 수 있다.

3.1.5. Serial EEPROM(93c46)

Ethernet 의 MAC Address가 저장되는 곳으로 93c46에서 NVRAM으로 복사된다.

3.2. The contents of a LKV-408 B/D I/O Control

3.2.1. WatchDog time out Reset Control

3.2.1.1. Watchdog time Reset 기능

LKV-408 보드에서는 Maxim사의 MAX690A라는 칩을 사용하여 WatchDog time out 기능을 구현하였다. 보드 셋팅은 **J17**의 shunt를 연결하면 reset 기능은 enable 상태로 되며, WatchDog reset time은 1.6초이다.

3.2.1.2. WatchDog time control Function

1) void wdtEnable()

WatchDog timer Enable Function 으로서 이 함수를 Call 하면 WatchDog Timer 가 On 된다.

2) void wdtDisable()

WatchDog timer Disable Function 으로서 이 함수를 Call하면 WatchDog Timer가 Off된다.

3) void wdtReset()

WatchDog Timer가 Enable된 상태라면 약 1.6초 후에 Board는 Reset 된다.

3.2.1.3. WatchDog time out Test Example

```
extern void wdtEnable();
extern void wdtReset();

void watchDogTest()
{
    sysClkRateSet(1000);

    wdtEnable();

    while (1) {
        if (ErrorStatus == ERROR) wdtReset();
    }
}
```


3.2.2. Digital IN/OUT Control

3.2.2.1. Function Description

LKV-408 보드에서는 4개의 Digital Input, 4개의 Digital Output Channel이 존재한다. 이 Channel을 Control 하기 위하여 다음과 같은 Function을 사용하면 된다.

```
#define DI_ON          1
#define DI_OFF         0
#define DO_ON          1
#define DO_OFF         0
```

1) char digInChan(int chan)

Digital Input Channel의 값을 1개 Channel 단위로 Return 시켜주는 Function이다. Argument로 0~3의 Channel 값을 넣어 주어야 한다.

2) char digIn()

Digital Input Channel의 값을 4개 Channel 단위로 Return 시켜주는 Function이다. Argument 없이 사용한다.

3) int digOutChan(int chan, char value)

Digital Input Channel의 값을 1개 Channel 단위로 Fill 시켜주는 Function 이다. Argument로 0~3의 Channel 값과 Channel의 ON/OFF Value을 넣어 주어야 한다.

4) int digOut(char value)

Digital Input Channel의 값을 4개 Channel 단위로 Fill 시켜주는 Function 이다. Argument로 4개 Channel 모두의 ON/OFF Value를 넣어 주어야 합니다.

3.2.3 Dip Switch Control

3.2.3.1. Function Description

LKV-408 보드에서는 8개의 Dip Switch로 구성되어 있으며, Switch on시 data는 0으로 read 되며, switch off시 data는 1로 read 된다. 이 Channel을 Control 하기 위하여 아래와 같은 Function을 사용하면 된다.

1) DipSwRead (int prn)

Dip Switch Value의 셋팅 설정과 Ethernet을 disable 시켜주는 Function 이다. 현재 LKV-408 Board 에서 DIP S/W 1번 의 경우 SCO Select 기능으로 사용되고 있으므로 DIP가 정상적으로 Read 될 경우 0x7F를 Return 하게 됩니다. 또한 “Prn”이 “1”일경우 Read 된 Dip Switch의 값 이 shell을 통해 출력되게 됩니다. **DIP S/W control 시 주의 할 사항은 LKV-408 보드의 DIP7와 DIP8을 ON 하게 될 경우 LKV-408보드의 CH0와 CH1이 VME Bus를 통해 제어 되게 됩니다.**

EX) DIP SW Read Function 실행 출력결과

```
-> DipSwRead 1
dip_sw_value : 0x7f[127]
value = 127 = 0x7f
```

3.2.3.2. Dip Switch Test Example

아래의 테스트 프로그램은 Dip Switch의 셋팅 상태에 대한 정보를 볼 수 있다.

```
=====
DipSwich Setting value Check Test
=====
Dip switch read value = 0x7f
Dip Switch[0] = ON
Dip Switch[1] = OFF
Dip Switch[2] = OFF
Dip Switch[3] = OFF
Dip Switch[4] = OFF
Dip Switch[5] = OFF
Dip Switch[6] = OFF
Dip Switch[7] = OFF
```

3.2.4. Encoder Switch Control

3.2.4.1. Function Description

LKV-408의 Encoder Switch는 Ethernet IP Setting시 사용된다. 보드상에 IP0와 IP1으로 2개가 있으

며, IP0는 상위 4bit, IP1은 하위 4bit로 IP0와 IP1을 합쳐서 1byte로 Read 된다.

1) encRead Function 실행 출력 결과

Read 된 값이 Return 된다.

-> encRead 1

Encoder rotary switch value = 0x64[100]

value = 100 = 0x64 = 'd'

3.2.5. ledToggle (2009/02/19 추가됨)

3.2.5.1. Function Description

LS 산전 측에서 KVME080 과 같은 Diag 0, 1 LED 를 on/off 시킬 수 있는 함수를 추가해 달라고 하여 KVME080 에서 사용하는 ledToggle 함수를 LKV-408 BSP 에 추가함.

ledToggle 함수의 인자에 0을 대입하면 Diag 0 LED가, 1을 대입하면 Diag 1 LED 가 on/off 된다.

4. LKV-408 Serial Control method & USER API

4.1. LKV-408 B/D Serial Driver Initialization

LKV-408 보드에는 6개의 Serial Port가 기본적으로 장착되어 있으며 각각의 Serial Port의 모든 신호선에는 surge protection chip이 장착되어 있다. Baud rate generation을 위한 기본 클럭은 7.3728Mhz가 공급된다. 6개의 Serial Port중 CH5은 보드 Debug를 위한 Console 포트로 BaudRate는 38400으로 셋팅되어 사용되며, CH2, CH3, CH4와 함께 RS232 전용이다. 그리고 Ethernet0와 Ethernet1은 Ethernet전용 Port이며, CH0와 CH1은 HDLC 고속 통신용 485 Serial Port로 사용된다.

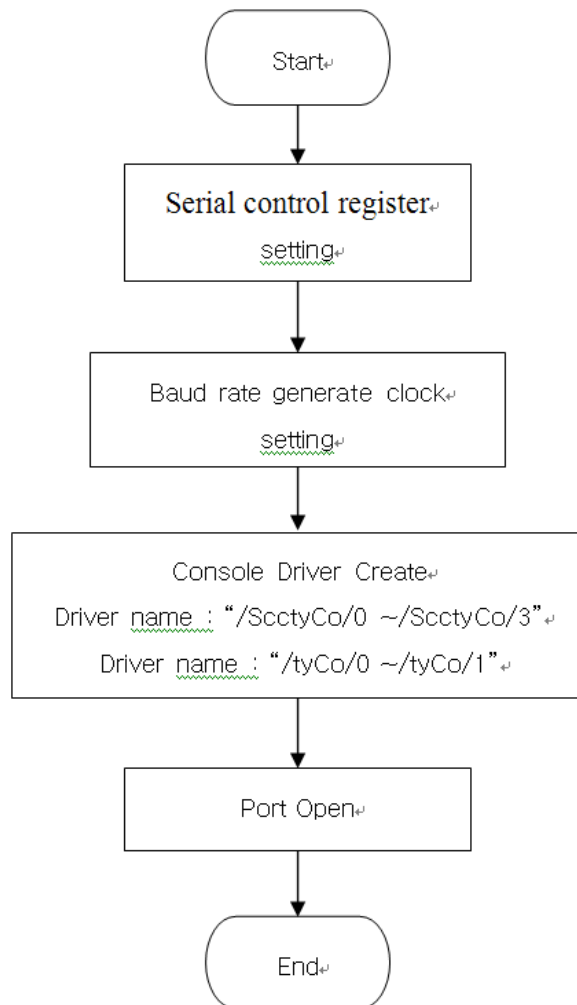


그림 5. LKV-408 B/D Serial Driver Initial Sequence

4.2. VxWorks Serial I/O System

LKV-408 보드는 IO System을 통해서 create(), remove() open(), close(), read(), write(), ioctl() 함수를 지원하며 VxWorks에서 일반적으로 지원하는 기능과 같다.

4.2.1. open ()

1) Synopsis

```
int open ( const char * name, int flag, int mode )
```

2) Description

ⓐ name

현재 생성되어야 될 프로세서의 드라이버를 가리킨다. 드라이버의 이름은 각각의 프로세서마다 다르며 파일단위로 생성됨. Name을 통해서 드라이버에 접근 가능함.

ⓑ flag

O_RDONLY(0) : 드라이버 읽기 전용으로 생성

O_WRONLY(1) : 드라이버 쓰기 전용으로 생성

O_RDWR(2) : 드라이버 읽기/쓰기로 생성

O_CREATE(0x0200) : 드라이버를 파일단위로 생성

ⓒ mode

UNIX 시스템에서 파일접근 허가를 나타내는 숫자를 수록함.

3) Return Value

ⓐ number

파일단위로 생성된 디바이스의 개수를 나타냄.

ⓑ ERROR

해당 디바이스가 없거나 파일네임과 맞지 않을 경우를 나타냄.

4.2.1.2. close ()

1) Synopsis

```
STATUS close ( int fd )
```

2) Description

ⓐ fd

시스템으로부터 드라이버 파일의 사용이 끝났음을 알림.

2) Return Value

ⓐ OK

호출이 성공적으로 이루어 졌을 경우를 나타냄.

ⓑ ERROR

해당 드라이버가 없을 경우 또는 파일 기술자가 아닐 경우.

4.2.3. read ()

1) Synopsis

```
int read ( int fd, char * buffer, size_t maxbytes )
```

2) Description

개방된 드라이버 파일로부터 일정 수의 바이트를 버퍼로 복사하기 위해 사용됨.

ⓐ fd

버퍼로 읽어 들일 드라이버의 파일명.

ⓑ buffer

읽어 들일 파일의 저장공간으로 char type의 포인터로 정식 선언되며 1문자와 1바이트는 구별 없이 사용할 수 있다. 즉 buffer는 자료가 복사되어질 문자배열에 대한 포인터 임.

ⓒ maxbytes

파일로부터 읽혀질 바이트의 수를 나타내는 양의 정수임.

3) Return Value

ⓐ number

파일단위로 생성된 디바이스에서 버퍼에 저장된 바이트의 개수를 나타냄.

ⓑ ERROR

해당 디바이스가 없거나 파일네임과 맞지 않을 경우를 나타냄.

4.2.4. write ()

1) Synopsis

```
int write ( int fd, char * buffer, size_t nbytes)
```

2) Description

write의 호출은 read와는 반대로 문자배열로 선언된 프로그램 버퍼로부터 개방된 파일 드라이버를 통해 일정 수의 바이트를 출력하거나 쓰기 위해 사용됨.

ⓐ fd

출력하거나 쓰기 위한 드라이버의 파일명.

ⓑ buffer

문자배열로 선언된 프로그램 버퍼로 char type의 포인터로 정식 선언되며 1문자와 1바이트는 구별 없이 사용할 수 있다. 즉 buffer는 출력될 자료의 문자배열에 대한 포인터 임.

ⓒ nbytes

파일로부터 출력되어야 할 바이트의 수를 나타내는 양의 정수임.

3) Return Value

ⓐ number

프로그램 버퍼로부터 디바이스에 쓰여진 바이트의 개수를 나타냄.

ⓑ ERROR

해당 디바이스가 없거나 파일네임과 맞지 않을 경우를 나타냄.

4.2.5. ioctl ()

1) Synopsis

int ioctl (int fd, int function, int arg)

2) Description

Device 의 I/O 컨트롤 함수로 매우 유용하게 쓰인다.

ⓐ fd

컨트롤 할 디바이스의 터미널 명.

ⓑ function

I/O 컨트롤 함수는 디바이스 드라이버에 따라 각 함수에 대한 옵션이 정해져 있으며 제공되는 드라이버의 옵션은 다음과 같다.

- FIOBAUDRATE

Baud Rate 을 Set 한다.

- FIOGETOPTIONS

각 채널에 대한 옵션을 얻어옴.

- FIOSETOPTIONS

각 채널에 대한 옵션을 셋팅함.

- FIOFLUSH

RX Buffer clear

- FIONREAD

RX Buffer에 Read되지 못한 바이트의 수를 구함.

※ 주의 : 통신 도중에 에러가 발생해서 데이터 수신에 불안정한 경우에는 기본적으로 사용 중이던 포트를 close() API를 이용해서 사용 중지하고, 다시 open() API를 사용해서 해당 통신 포트를 초기화한 후에 read()/write() API 등을 사용해서 통신을 수행하는 것이 보다 안전합니다.

4.2.6 HDLC I/O Control Method

4.2.6.1. HDLC Frame

KVME860에서 제공되는 HDLC FRAME는 [그림 6]와 같다.

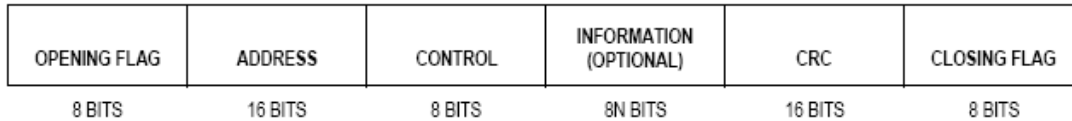


그림 6. HDLC Frame Structure

현재 Driver단에서 셋팅 되어 있는 레지스터는 다음과 같으며, 860chip 의 “gsmr_l”, “gsmr_h” 레지스터를 셋팅 하여, HDLC FRAM을 구현하였다.

- C_MASK : 0xF0B8 , CRC mask. For the 16 bit CRC-CCITT
- C_PRES : 0xFFFF , CRC preset. For the 16 bit CRC-CCITT
- DISFC : 0x0 , discard frame counter
- CRCEC : 0x0 , CRC error counter
- ABTSC : 0x0 , abort sequence counter
- NMARC : 0x0 , non matching address Rx counter
- RETRC : 0x0 , frame transmission counter
- MFLR : 1024 , max frame length register
(Write 0x0100 so the max frame size is 256bytes.)
- MAX_CNT : 0x0 , max length count
- RFTHR : 0x1 , received frames threshold
- RFCNT : 0x0 , received frames count
- HMASK : 0x0 , user defined frame address mask
- HADDR1 : 0x0 , user defined frame address 1
- HADDR2 : 0x0 , user defined frame address 2
- HADDR3 : 0x0 , user defined frame address 3
- HADDR4 : 0x0 , user defined frame address 4

HDLC FRAME Structure는 Manual에서 제공되는 Setting Flow 와 같이 셋팅 되며, 실질적으로 HCLD Mode로 통신을 하기 위해서는 GSMR_L 레지스터에서 설정된 형식과 같이 통신이 된다. HDLC BaudRate 는 250kpbs로 셋팅 되어 있습니다.

“GSMR_L” 레지스터에서 설정되는 내용은 다음과 같다.

- [1] Tx preamble pattern
- [2] Receiver decoding/ transmitter encoding method : Manchester code
- [3] SCC HDLC mode
- [4] DPLL RX input invert data

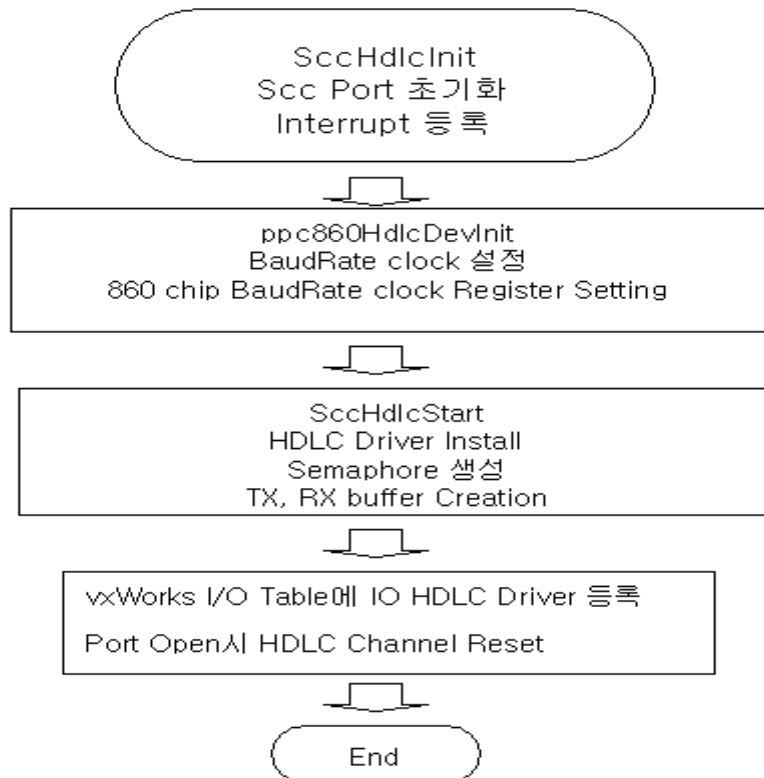


그림 7. LKV-408 B/D HDLC Serial Driver Initial Sequence

4.3. VxWorks Serial I/O System

다음은 user가 사용하기 쉽도록 BSP에서 제공되는 함수들입니다. VxWorks에서 제공되는 함수는 아니며, board dependent한 함수입니다.

(1) STATUS inetIpSetChange (char *ipStr, int what)

LKV-408의 Ethernet 은 2port이며, 2번째 인자 what에 따라서 Ethernet 0의 IP와 Ethernet의 1의 IP를 변경 할 수 있습니다.

EX) 0일 경우 Ethernet IP0가 변경되며, 1일 경우 Ethernet IP1이 변경된다.

```
inetIpSetChange ("192.168. 100. XXX", 0)
```

```
inetIpSetChange("192. 169. 200. XXX", 1)
```

"XXX"는 Encoder S/W 값이 셋팅 되므로, 0과 255 사이의 아무 값이나 써주면 된다.

(2) get_hdlc_ch_id(void)

Encoder Switch 값이 return 됨

(3) STATUS ledCon(char led)

LKV-408 Board 의 DIAG0 LED와 DIAG1의 LED를 Control 할 수 있다

LedCon(0) : DIAG0 LED ON

LedCon(1) : DIAG0 LED OFF

LedCon(2) : DIAG1 LED ON

LedCon(3) : DIAG1 LED OFF

(4) UCHAR DipSwRead (int prn)

DIP SW의 값이 Return됨

5. 주의 사항

CAUTION

- VME Rack에서 가장 왼쪽에 위치한 보드는 System Controller로 설정할 것.
- System Controller 보드는 하나의 Rack에 하나만 장착할 것.
- System Controller 보드 이외의 보드는 마스터/Slave 보드로 설정할 것.
- LKV-408 보드가 장착되는 Rack의 GND는 FGND와 직접 연결 하거나 Capacitor 를 통해 연결할 것.

WARNING

- LKV-408 보드는 정전기(Electrostatic Discharge)에 취약할 수 있으니, 보드 취급 시 주의할 것.
- LKV-408 보드를 Rack에 장착할 시에는 가능하면 전원을 끄고 작업할 것.
- Rack이 접지되지 않았을 경우 감전의 우려가 있으므로, 반드시 접지 여부를 확인 하고, 물이나 땀이 묻은 손으로 작업하지 말 것.



주엘케이일레븐

138-809 서울특별시 송파구 가락2동 545-5번지 동명빌딩 3층

<http://www.lk11.com>, 전화:02-3012-3788