

LKP-Serial보드

상세 설계서

Version Information

H/W Version : Version 1.0

소속 : (주)엘케이일레븐 연구소

주소 : 성남시 중원구 상대원동 190-1 SKn테크노파크 메가센터 1306호

전화 : 031-776-4120 / FAX : 031-766-4119

목차

1. 개요.....	6
2. 사양.....	7
3. 블록도.....	8
4. 커넥터 위치	9
5. 외부 연결	10
5.1. PMC 슬롯 핀 번호	10
5.2. LKV-080 보드와 연결	10
6. 메모리 지도	13
6.1. 이미지 0	13
6.2. 이미지 1	13
6.3 이미지 2.....	16
7. 레지스터 설명	17
7.1. PCI 설정 공간	17
7.2. 인터럽트 제어 레지스터(ICR).....	17
7.3. 인터럽트 상태 레지스터(ISR).....	17
7.4. UART 받기 버퍼 레지스터(UART_RBR).....	17
7.5. UART 보내기 홀딩 레지스터(UART_THR).....	17
7.6. UART 인터럽트 가능 레지스터(UART_IER)	18
7.7. UART 인터럽트 구별 레지스터(UART_IIR)	18
7.8. UART FIFO 제어 레지스터(UART_FCR).....	19
7.9. UART 선 제어 레지스터(UART_LCR).....	19
7.10. UART 모뎀 제어 레지스터(UART_MCR)	20
7.11. UART 선 상태 레지스터(UART_LSR)	20
7.12. UART 모뎀 상태 레지스터(UART_MSR).....	21
7.13. UART 스크래치 레지스터(UART_SCR)	21
7.14. UART 나누기 레지스터 LSB(UART_DLL)	21
7.15. UART 나누기 레지스터 MSB(UART_DLM)	21
7.16. DIO_DIR 레지스터	21
7.17. DIO_DOUT 레지스터.....	22
7.18. DIO_DIN 레지스터	22
7.19. NVRAM.....	22
7.20 RTC 레지스터.....	23
8. 디바이스 드라이버.....	24
8.1. 초기화 함수.....	24

8.2. UART 관련 함수	24
8.3. 디지털 입출력 관련 함수	24
8.4. NVRAM 관련 함수	24

그림 목차

그림 1. LKP-Serial	6
그림 2. LKP-Serial 블록도.....	8
그림 3. LKP-Serial 커넥터 배치	9
그림 4. LKP-Serial PMC 슬롯 핀 번호.....	10

표 목차

표 1. LKV-080 P2 핀 번호(RS-232)	10
표 2. LKV-080 P2 핀 번호(RS-422)	11
표 2. 이미지 0 메모리 지도.....	13
표 3. 이미지 1 메모리 지도.....	13
표 3. 이미지 2 메모리 지도.....	16
표 4. ICR 비트 정의	17
표 5. ISR 비트 정의	17
표 6. UART_IER 비트 정의	18
표 7. UART_IIR 비트 정의	18
표 8. IIR_IID.....	18
표 9. UART_FCR 비트 정의	19
표 10. R_TRIG.....	19
표 11. UART_LCR 비트 정의	19
표 12. LCR_WLEN.....	20
표 13. UART_MCR 비트 정의.....	20
표 14. ADC_STATUS 비트 정의.....	21
표 14. ADC_STATUS 비트 정의.....	22
표 14. ADC_STATUS 비트 정의.....	22
표 14. RTC 레지스터 정의.....	23

1. 개요

LKP-Serial 보드는 NVRAM, RS-232 또는 RS-422 시리얼 통신 및 디지털 입출력을 지원하는 보드이다. NVRAM 용량은 512K 바이트 이고, RS-232 또는 RS-422 시리얼 8 채널을 처리 할 수 있고, 8개의 디지털 입출력 포트가 제공되고, PMC 슬롯을 통해 LKV-080 보드와 인터페이스 된다. 시리얼 통신의 경우 각각의 채널에 대해 생산 단계에서 RS-232 또는 RS-422 인터페이스를 선택 할 수 있고 16550 UART와 호환된다. 디지털 입출력 포트는 각 포트마다 소프트웨어로 입력 및 출력 기능을 선택 할 수 있다.

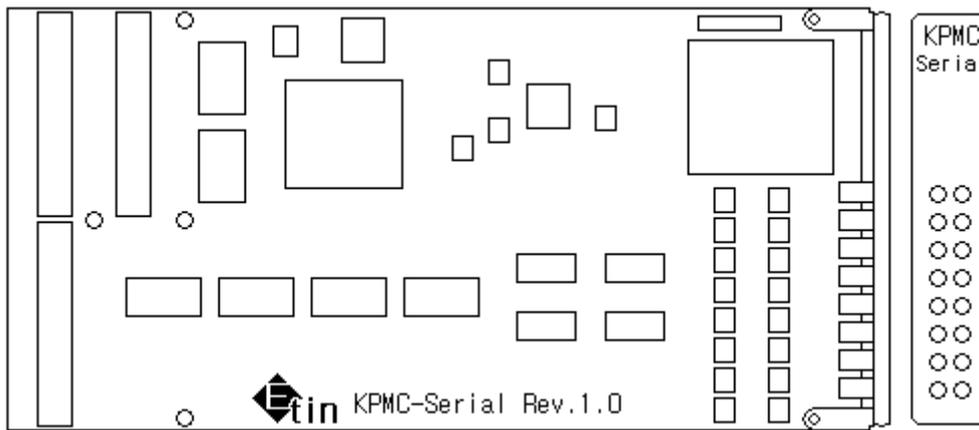


그림 1. LKP-Serial

2. 사양

PMC-Serial보드는 다음과 같은 사양을 가진다.

NVRAM:

512K 바이트

RS-232 또는 RS-422 인터페이스:

8 채널

생산 단계에서 RS-232 또는 RS-422 포트 결정 가능

디지털 I/O:

8 채널

각 채널 별로 소프트웨어로 입출력 선택 가능

3. 블록도

다음은 LKP-Serial 보드의 블록도 이다. NVRAM은 512K이고, 8개의 RS-232 또는 RS-422 통신을 지원하고, 8개의 디지털 입출력을 지원하고, PMC 포트를 통해 외부와 연결된다.

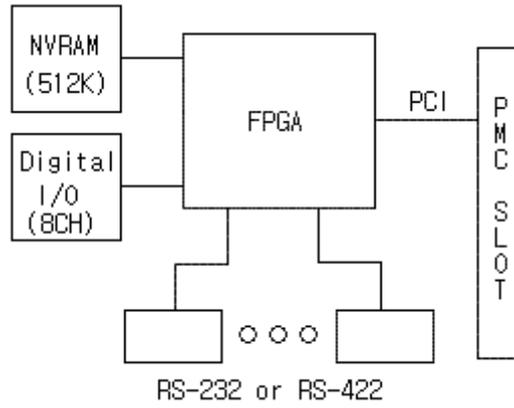


그림 2. LKP-Serial 블록도

4. 커넥터 위치

다음 그림은 LKP-Serial 보드의 커넥터 위치이다. 왼쪽에 3개의 64핀 PMC 커넥터가 배치된다.

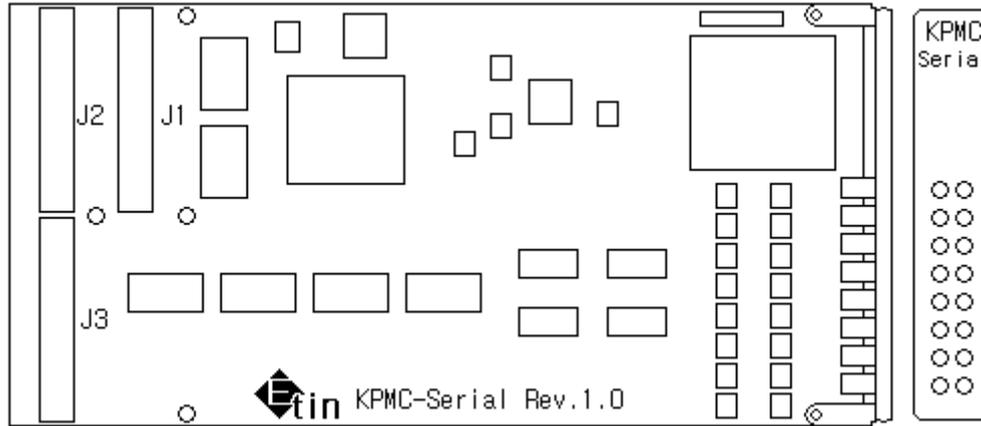


그림 3. LKP-Serial 커넥터 배치

J1과 J2에는 PCI 신호가 인터페이스 되고, J3에는 RS-232 또는 RS-422 포트와 디지털 입출력 포트가 인터페이스 된다.

5. 외부 연결

5.1. PMC 슬롯 핀 번호

PMC 슬롯의 핀 번호는 다음과 같다.

	J1				J2				J3		
TCK	1	2	-12V	+12V	1	2	TRST#	Ground	1	2	TXD#0
Ground	3	4	INTA#	TMS	3	4	TDO	RXD#0	3	4	RTS0
INTB#	5	6	INTC#	TDI	5	6	Ground	CTS0	5	6	Ground
BUSMODE1#	7	8	+5V	Ground	7	8	PCI-RSVD	TXD#1	7	8	RXD1
INTD#	9	10	PCI-RSVD	PCI-RSVD	9	10	PCI-RSVD	RTS1	9	10	CTS1
Ground	11	12	3.3Vaux	BUSMODE2#	11	12	+3.3V	Ground	11	12	TXD#2
CLK	13	14	Ground	RST#	13	14	BUSMODE3#	RXD#2	13	14	RTS2
Ground	15	16	GNT#	3.3V	15	16	BUSMODE4#	CTS2	15	16	Ground
REQ#	17	18	+5V	PME#	17	18	Ground	TXD#3	17	18	RXD3
V(1/0)	19	20	AD[31]	AD[30]	19	20	AD[29]	RTS3	19	20	CTS3
AD[28]	21	22	AD[27]	Ground	21	22	AD[26]	Ground	21	22	TXD#4
AD[25]	23	24	Ground	AD[24]	23	24	+3.3V	RXD#4	23	24	RTS4
Ground	25	26	C/BE[3]#	IDSEL	25	26	AD[23]	CTS4	25	26	Ground
AD[22]	27	28	AD[21]	+3.3V	27	28	AD[20]	TXD#5	27	28	RXD5
AD[19]	29	30	+5V	AD[18]	29	30	Ground	RTS5	29	30	CTS5
V(1/0)	31	32	AD[17]	AD[16]	31	32	C/BE[2]#	Ground	31	32	TXD#6
FRAME#	33	34	Ground	Ground	33	34	PMC-RSVD	RXD#6	33	34	RTS6
Ground	35	36	IRDY#	TRDY#	35	36	+3.3V	CTS6	35	36	Ground
DEVSEL#	37	38	+5V	Ground	37	38	STOP#	TXD#7	37	38	RXD7
Ground	39	40	LOCK#	PERR#	39	40	Ground	RTS7	39	40	CTS7
PCI-RSVD	41	42	PCI-RSVD	+3.3V	41	42	SERR	Ground	41	42	+5V
PAR	43	44	Ground	C/BE[1]#	43	44	Ground	DCD0	43	44	DTR0
V(1/0)	45	46	AD[15]	AD[14]	45	46	AD[13]	R10	45	46	DSR0
AD[12]	47	48	AD[11]	M66EN	47	48	AD[10]	DCD1	47	48	DTR1
AD[09]	49	50	+5V	AD[08]	49	50	+3.3V	R11	49	50	DSR1
Ground	51	52	C/BE[0]#	AD[07]	51	52	PMC-RSVD	DCD2	51	52	DTR2
AD[06]	53	54	AD[05]	+3.3V	53	54	PMC-RSVD	R12	53	54	DSR2
AD[04]	55	56	Ground	PMC-RSVD	55	56	Ground	D100	55	56	D101
V(1/0)	57	58	AD[03]	PMC-RSVD	57	58	PMC-RSVD	D102	57	58	D103
AD[02]	59	60	AD[01]	Ground	59	60	PMD-RSVD	D104	59	60	D105
AD[00]	61	62	+5V	ACK64#	61	62	+3.3V	D106	61	62	D107
Ground	63	64	REQ64#	Ground	63	64	PMC-RSVD	NC	63	64	NC

그림 4. LKP-Serial PMC 슬롯 핀 번호

J1과 J2에는 PCI 신호가 인터페이스 되고, J3에는 시리얼 8 채널과 디지털 입출력 8채널이 인터페이스 된다.

5.2. LKV-080 보드와 연결

LKV-080 보드와 연결 했을 때 LKV-080의 P2로 연결되는 핀 번호는 다음과 같다.

표 1은 RS-232 보드 핀 번호이고, 표 2는 RS-422 보드 핀 번호이다.

표 1. LKV-080 P2 핀 번호(RS-232)

핀	신호	핀	신호
A1	TXD#0	C1	GND
A2	RTS0	C2	RXD#0
A3	GND	C3	CTS0
A4	RXD#1	C4	TXD#1

핀	신호	핀	신호
A5	CTS1	C5	RTS1
A6	TXD#2	C6	GND
A7	RTS2	C7	RXD#2
A8	GND	C8	CTS2
A9	RXD#3	C9	TXD#3
A10	CTS3	C10	RTS3
A11	TXD#4	C11	GND
A12	RTS4	C12	RXD4
A13	GND	C13	CTS4
A14	RXD#5	C14	TXD#5
A15	CTS5	C15	RTS5
A16	TXD#6	C16	GND
A17	RTS6	C17	RXD#6
A18	GND	C18	CTS6
A19	RXD#7	C19	TXD#7
A20	CTS7	C20	RTS7
A21	+5V	C21	GND
A22	DTR0	C22	DCD0
A23	DSR0	C23	RI0
A24	DTR1	C24	CDC1
A25	DSR1	C25	RI1
A26	DTR2	C26	DCD2
A27	DSR2	C27	RI2
A28	DIO1	C28	DIO0
A29	DIO3	C29	DIO2
A30	DIO5	C30	DIO4
A31	DIO7	C31	DIO6
A32	NC	C32	NC

표 2. LKV-080 P2 핀 번호(RS-422)

핀	신호	핀	신호
A1	TxD0-	C1	GND
A2	RxD0-	C2	TxD0+
A3	GND	C3	RxD0+

핀	신호	핀	신호
A4	TxD1+	C4	TxD1-
A5	RxD1+	C5	RxD1-
A6	TxD2-	C6	GND
A7	RxD2-	C7	TxD2+
A8	GND	C8	RxD2+
A9	TxD3+	C9	TxD3-
A10	RxD3+	C10	RxD3-
A11	TxD4-	C11	GND
A12	RxD4-	C12	TxD4+
A13	GND	C13	RxD4+
A14	TxD5+	C14	TxD5-
A15	RxD5+	C15	RxD5-
A16	TxD6-	C16	GND
A17	RxD6-	C17	TxD6+
A18	GND	C18	RxD6+
A19	TxD7+	C19	TxD7-
A20	RxD7+	C20	RxD7-
A21	+5V	C21	GND
A22	-	C22	-
A23	-	C23	-
A24	-	C24	-
A25	-	C25	-
A26	-	C26	-
A27	-	C27	-
A28	DIO1	C28	DIO0
A29	DIO3	C29	DIO2
A30	DIO5	C30	DIO4
A31	DIO7	C31	DIO6
A32	NC	C32	NC

6. 메모리 지도

LKP-Serial의 메모리 지도는 이미지 0, 이미지 1 및 이미지 2의 3개 영역으로 나누어 진다. 각 영역의 크기는 이미지 1과 2는 64K 바이트 이고, 이미지 3은 512K 바이트 이다.

6.1. 이미지 0

이미지 0는 PCI 설정 레지스터, 주소 변환, 인터럽트 설정 등에 관련된 레지스터 이다.

이미지 0의 메모리 지도는 다음과 같다.

표 3. 이미지 0 메모리 지도

이름	주소	넓이	접근	설명
PCI 설정 공간	0x000 - 0x0FF			PCI 규격 Rev. 2.2 설정 공간
ICR	0x1EC	32	R/W	인터럽트 제어 레지스터
ISR	0x1F0	32	R/W	인터럽트 상태 레지스터

6.2. 이미지 1

이미지 1의 메모리 지도는 다음과 같다.

표 4. 이미지 1 메모리 지도

이름	주소	넓이	접근	설명
UART0_RBR	0x0000	8	R	UART 0 받기 버퍼 레지스터
UART0_THR	0x0000	8	W	UART 0 보내기 홀딩 레지스터
UART0_IER	0x0001	8	R/W	UART 0 인터럽트 가능 레지스터
UART0_IIR	0x0002	8	R	UART 0 인터럽트 구별 레지스터
UART0_FCR	0x0002	8	W	UART 0 FIFO 제어 레지스터
UART0_LCR	0x0003	8	R/W	UART 0 선 제어 레지스터
UART0_MCR	0x0004	8	R/W	UART 0 모뎀 제어 레지스터
UART0_LSR	0x0005	8	R/W	UART 0 선 상태 레지스터
UART0_MSR	0x0006	8	R/W	UART 0 모뎀 상태 레지스터
UART0_SCR	0x0007	8	R/W	UART 0 스크래치 레지스터
UART0_DLL	0x0000	8	R/W	UART 0 나누기 레지스터(LSB)
UART0_DLM	0x0001	8	R/W	UART 0 나누기 레지스터(MSB)
UART1_RBR	0x0008	8	R	UART 1 받기 버퍼 레지스터
UART1_THR	0x0008	8	W	UART 1 보내기 홀딩 레지스터
UART1_IER	0x0009	8	R/W	UART 1 인터럽트 가능 레지스터
UART1_IIR	0x000A	8	R	UART 1 인터럽트 구별 레지스터
UART1_FCR	0x000A	8	W	UART 1 FIFO 제어 레지스터

이름	주소	넓이	접근	설명
UART1_LCR	0x000B	8	R/W	UART 1 선 제어 레지스터
UART1_MCR	0x000C	8	R/W	UART 1 모뎀 제어 레지스터
UART1_LSR	0x000D	8	R/W	UART 1 선 상태 레지스터
UART1_MSR	0x000E	8	R/W	UART 1 모뎀 상태 레지스터
UART1_SCR	0x000F	8	R/W	UART 1 스크래치 레지스터
UART1_DLL	0x0008	8	R/W	UART 1 나누기 레지스터(LSB)
UART1_DLM	0x0009	8	R/W	UART 1 나누기 레지스터(MSB)
UART2_RBR	0x0010	8	R	UART 2 받기 버퍼 레지스터
UART2_THR	0x0010	8	W	UART 2 보내기 홀딩 레지스터
UART2_IER	0x0011	8	R/W	UART 2 인터럽트 가능 레지스터
UART2_IIR	0x0012	8	R	UART 2 인터럽트 구별 레지스터
UART2_FCR	0x0012	8	W	UART 2 FIFO 제어 레지스터
UART2_LCR	0x0013	8	R/W	UART 2 선 제어 레지스터
UART2_MCR	0x0014	8	R/W	UART 2 모뎀 제어 레지스터
UART2_LSR	0x0015	8	R/W	UART 2 선 상태 레지스터
UART2_MSR	0x0016	8	R/W	UART 2 모뎀 상태 레지스터
UART2_SCR	0x0017	8	R/W	UART 2 스크래치 레지스터
UART2_DLL	0x0010	8	R/W	UART 2 나누기 레지스터(LSB)
UART2_DLM	0x0011	8	R/W	UART 2 나누기 레지스터(MSB)
UART3_RBR	0x0018	8	R	UART 3 받기 버퍼 레지스터
UART3_THR	0x0018	8	W	UART 3 보내기 홀딩 레지스터
UART3_IER	0x0019	8	R/W	UART 3 인터럽트 가능 레지스터
UART3_IIR	0x001A	8	R	UART 3 인터럽트 구별 레지스터
UART3_FCR	0x001A	8	W	UART 3 FIFO 제어 레지스터
UART3_LCR	0x001B	8	R/W	UART 3 선 제어 레지스터
UART3_MCR	0x001C	8	R/W	UART 3 모뎀 제어 레지스터
UART3_LSR	0x001D	8	R/W	UART 3 선 상태 레지스터
UART3_MSR	0x001E	8	R/W	UART 3 모뎀 상태 레지스터
UART3_SCR	0x001F	8	R/W	UART 3 스크래치 레지스터
UART3_DLL	0x0018	8	R/W	UART 3 나누기 레지스터(LSB)
UART3_DLM	0x0019	8	R/W	UART 3 나누기 레지스터(MSB)
UART4_RBR	0x0020	8	R	UART 4 받기 버퍼 레지스터
UART4_THR	0x0020	8	W	UART 4 보내기 홀딩 레지스터
UART4_IER	0x0021	8	R/W	UART 4 인터럽트 가능 레지스터
UART4_IIR	0x0022	8	R	UART 4 인터럽트 구별 레지스터

이름	주소	넓이	접근	설명
UART4_FCR	0x0022	8	W	UART 4 FIFO 제어 레지스터
UART4_LCR	0x0023	8	R/W	UART 4 선 제어 레지스터
UART4_MCR	0x0024	8	R/W	UART 4 모뎀 제어 레지스터
UART4_LSR	0x0025	8	R/W	UART 4 선 상태 레지스터
UART4_MSR	0x0026	8	R/W	UART 4 모뎀 상태 레지스터
UART4_SCR	0x0027	8	R/W	UART 4 스크래치 레지스터
UART4_DLL	0x0020	8	R/W	UART 4 나누기 레지스터(LSB)
UART4_DLM	0x0021	8	R/W	UART 4 나누기 레지스터(MSB)
UART5_RBR	0x0028	8	R	UART 5 받기 버퍼 레지스터
UART5_THR	0x0028	8	W	UART 5 보내기 홀딩 레지스터
UART5_IER	0x0029	8	R/W	UART 5 인터럽트 가능 레지스터
UART5_IIR	0x002A	8	R	UART 5 인터럽트 구별 레지스터
UART5_FCR	0x002A	8	W	UART 5 FIFO 제어 레지스터
UART5_LCR	0x002B	8	R/W	UART 5 선 제어 레지스터
UART5_MCR	0x002C	8	R/W	UART 5 모뎀 제어 레지스터
UART5_LSR	0x002D	8	R/W	UART 5 선 상태 레지스터
UART5_MSR	0x002E	8	R/W	UART 5 모뎀 상태 레지스터
UART5_SCR	0x002F	8	R/W	UART 5 스크래치 레지스터
UART5_DLL	0x0028	8	R/W	UART 5 나누기 레지스터(LSB)
UART5_DLM	0x0029	8	R/W	UART 5 나누기 레지스터(MSB)
UART6_RBR	0x0030	8	R	UART 6 받기 버퍼 레지스터
UART6_THR	0x0030	8	W	UART 6 보내기 홀딩 레지스터
UART6_IER	0x0031	8	R/W	UART 6 인터럽트 가능 레지스터
UART6_IIR	0x0032	8	R	UART 6 인터럽트 구별 레지스터
UART6_FCR	0x0032	8	W	UART 6 FIFO 제어 레지스터
UART6_LCR	0x0033	8	R/W	UART 6 선 제어 레지스터
UART6_MCR	0x0034	8	R/W	UART 6 모뎀 제어 레지스터
UART6_LSR	0x0035	8	R/W	UART 6 선 상태 레지스터
UART6_MSR	0x0036	8	R/W	UART 6 모뎀 상태 레지스터
UART6_SCR	0x0037	8	R/W	UART 6 스크래치 레지스터
UART6_DLL	0x0030	8	R/W	UART 6 나누기 레지스터(LSB)
UART6_DLM	0x0031	8	R/W	UART 6 나누기 레지스터(MSB)
UART7_RBR	0x0038	8	R	UART 7 받기 버퍼 레지스터
UART7_THR	0x0038	8	W	UART 7 보내기 홀딩 레지스터
UART7_IER	0x0039	8	R/W	UART 7 인터럽트 가능 레지스터

이름	주소	넓이	접근	설명
UART7_IIR	0x003A	8	R	UART 7 인터럽트 구별 레지스터
UART7_FCR	0x003A	8	W	UART 7 FIFO 제어 레지스터
UART7_LCR	0x003B	8	R/W	UART 7 선 제어 레지스터
UART7_MCR	0x003C	8	R/W	UART 7 모뎀 제어 레지스터
UART7_LSR	0x003D	8	R/W	UART 7 선 상태 레지스터
UART7_MSR	0x003E	8	R/W	UART 7 모뎀 상태 레지스터
UART7_SCR	0x003F	8	R/W	UART 7 스크래치 레지스터
UART7_DLL	0x0038	8	R/W	UART 7 나누기 레지스터(LSB)
UART7_DLM	0x0039	8	R/W	UART 7 나누기 레지스터(MSB)
DIO_DIR	0x1001	8	R/W	디지털 입출력 방향 레지스터
DIO_OUT	0x1002	8	R/W	디지털 입출력 출력 레지스터
DIO_IN	0x1003	8	R	디지털 입출력 입력 레지스터

6.3 이미지 2

이미지 2의 메모리 지도는 다음과 같다.

표 5. 이미지 2 메모리 지도

이름	주소	넓이	접근	설명
NVRAM	0x00000-0x7fff0	8	R/W	NVRAM
RTD	0x7fff0-0x7ffff	8	R/W	RTD

7. 레지스터 설명

7.1. PCI 설정 공간

PCI 설정 공간은 PCI 베이스 주소, 인터럽트 등 PCI 버스에서 요구하는 각종 설정을 저장하는 공간으로 PCI 규격 2.2를 따른다.

7.2. 인터럽트 제어 레지스터(ICR)

인터럽트 제어 레지스터(ICR)는 인터럽트를 허용 하거나 금지 하는 역할을 한다.

표 6. ICR 비트 정의

비트 #	이름	설명
0	인터럽트 전달 허용	1이면, UART의 인터럽트가 INTA#로 전달 된다.
2	PCI 에러 인터럽트 허용	1이면, 쓰기 사이클 동안 에러가 발생했을 경우 인터럽트를 발생 시킨다.
3	패리티 에러 인터럽트 허용	1이면, 패리티 에러가 발견되었을 때 인터럽트를 발생 시킨다.
31	소프트웨어 Reset	1이면, 소프트웨어 Reset이 발생한다.

7.3. 인터럽트 상태 레지스터(ISR)

인터럽트 상태 레지스터(ISR)는 인터럽트의 상태를 나타낸다.

표 7. ISR 비트 정의

비트 #	이름	설명
0	인터럽트	1이면, UART에서 인터럽트가 발생했음을 나타낸다.
2	PCI 에러 인터럽트	1이면, 쓰기 사이클 동안 에러가 발견되어 인터럽트가 발생 했음을 나타낸다.
3	패리티 에러 인터럽트	1이면, 패리티 에러에 의해 인터럽트가 발생 했음을 나타낸다.

7.4. UART 받기 버퍼 레지스터(UART_RBR)

UART_RBR은 UART에서 RX를 통해 받은 데이터를 저장한다.

7.5. UART 보내기 홀딩 레지스터(UART_THR)

UART_THR은 UART에서 TX로 보낼 데이터를 저장한다.

7.6. UART 인터럽트 가능 레지스터(UART_IER)

UART_IER은 인터럽트를 허용하거나 금지하는 역할을 한다. UART인터럽트는 데이터 받음 인터럽트, 보내기 홀딩 레지스터 빔 인터럽트, 선 상태 인터럽트 및 모뎀 상태 인터럽트의 4가지가 있는데 각각의 인터럽트에 할당된 비트는 다음 표와 같다.

표 8. UART_IER 비트 정의

비트 #	이름	설명
0	IER_RDI	데이터 받음 인터럽트
1	IER_THRI	보내기 홀딩 레지스터 빔 인터럽트
2	IER_RLSI	선 상태 인터럽트
3	IER_MSI	모뎀 상태 인터럽트

7.7. UART 인터럽트 구별 레지스터(UART_IIR)

UART_IIR은 인터럽트의 발생여부와 어떤 인터럽트가 발생 했는지를 알려주는 역할을 한다. 또한 FIFO가 활성화 되어있는지를 알려준다.

표 9. UART_IIR 비트 정의

비트 #	이름	설명
0	IIR_NO_INT	처리되지 않은 인터럽트가 없으면 1
1	IIR_ID0	인터럽트 ID 비트 0
2	IIR_ID1	인터럽트 ID 비트 1
3	IIR_ID2	인터럽트 ID 비트 2
6	IIR_ENABLE_FIFO	FIFO가 활성화 되어 있으면 1
7	IIR_ENABLE_FIFO	FIFO가 활성화 되어 있으면 1

인터럽트 ID는 다음 표와 같이 정의된다. 여러 인터럽트가 동시에 발생했을 경우 우선순위가 높은 인터럽트 ID가 읽혀진다.

표 10. IIR_ID

IIR_ID	이름	우선순위	설명
011	IIR_RLSI	1	선 상태 인터럽트
010	IIR_RDI	2	데이터 받음 인터럽트
110	IIR_CTOI	2	문자 시간 초과 인터럽트
001	IIR_THRI	3	보내기 홀딩 레지스터 빔 인터럽트
000	IIR_MSI	4	모뎀 상태 인터럽트

문자시간 초과 인터럽트는 4개의 문자를 받을 수 있는 시간 동안 데이터 받기 레지스터에 데이터

가 있는데 읽어가지 않을 경우 발생한다.

7.8. UART FIFO 제어 레지스터(UART_FCR)

UART_FCR은 FIFO를 제어하는 역할을 한다. 각 비트의 정의는 다음 표와 같다.

표 11. UART_FCR 비트 정의

비트 #	이름	설명
0	FCR_ENABLE_FIFO	FIFO 활성화
1	FCR_CLEAR_RCVR	받기 FIFO 초기화
2	FCR_CLEAR_XMIT	보내기 FIFO 초기화
6	FCR_R_TRIG0	받기 방아쇠 비트 0
7	FCR_R_TRIG1	받기 방아쇠 비트 1

받기 방아쇠는 다음 표와 같이 동작한다.

표 12. R_TRIG

FCR_R_TRIG	설명
00	받기 FIFO에 데이터가 1개 이상 있을 때 데이터 받기 인터럽트 발생
01	받기 FIFO가 1/4 이상 찾을 때 데이터 받기 인터럽트 발생
10	받기 FIFO가 1/2 이상 찾을 때 데이터 받기 인터럽트 발생
11	받기 FIFO가 다 찾을 때 데이터 받기 인터럽트 발생

7.9. UART 선 제어 레지스터(UART_LCR)

UART_LCR은 워드 크기, 정지 비트, 패리티 등을 설정하는 역할을 한다. 각 비트의 정의는 다음 표와 같다.

표 13. UART_LCR 비트 정의

비트 #	이름	설명
0	LCR_WLEN0	워드 크기 비트 0
1	LCR_WLEN1	워드 크기 비트 1
2	LCR_STOP	정지 비트
3	LCR_PARITY	패리티
4	LCR_EPAR	짝수 패리티
5	LCR_SPAR	고정 패리티
6	LCR_SBRK	브레이크
7	LCR_DLAB	나누기 레지스터 접근

LCR_WLEN은 다음 표와 같이 설정된다.

표 14. LCR_WLEN

LCR_WLEN	워드 크기
00	5 비트
01	6 비트
10	7 비트
11	8 비트

LCR_STOP은 0일 경우 정지 비트가 1비트로 설정되고, 1일 경우 정지비트가 2비트로 설정된다.

LCR_PARITY는 1일 경우 패리티가 활성화 된다.

LCR_EPAR는 1일 경우 짝수 패리티가, 0일 경우 홀수 패리티가 설정된다.

LCR_SPAR는 1일 경우 LCR_EPAR이 1이면 패리티 비트가 무조건 0이 되고, 0이면 패리티 비트가 무조건 1이 된다.

LCR_BREAK는 1일 경우 TX로 무조건 0이 나간다.

LCR_DLAB는 1이면 UART_DLL, UART_DLM 레지스터가 선택되고, 0이면 UART_RBR, UART_THR, UART_IER 레지스터가 선택된다.

7.10. UART 모뎀 제어 레지스터(UART_MCR)

UART_MCR은 모뎀 제어 신호를 제어하는 역할을 한다. 비트 정의는 다음과 같다.

표 15. UART_MCR 비트 정의

비트 #	이름	설명
0	MCR_DTR	1이면 DTR 신호에 0, 0이면 DTR 신호에 1이 나간다.
1	MCR_RTS	1이면 RTS 신호에 0, 0이면 RTS 신호에 1이 나간다.
4	MCR_LOOP	1이면 TX, RX 신호와 모뎀 제어 신호가 내부적으로 연결된다.
5	MCR_AFE	1이면 자동 흐름 제어가 활성화 된다.

7.11. UART 선 상태 레지스터(UART_LSR)

UART_LSR은 데이터 전송의 상태를 나타낸다. 비트 정의는 다음 표와 같다.

비트 #	이름	설명
0	LSR_DR	데이터가 준비됨
1	LSR_OE	데이터가 RBR에 겹쳐 써짐
2	LSR_PE	패리티 에러
3	LSR_FE	프레임 에러
4	LSR_BI	브레이크 인터럽트

비트 #	이름	설명
5	LSR_THRE	보내기 홀딩 레지스터 빔
6	LSR_TEMT	보내기 시프트 레지스터 까지 빔
7	LSR_FIFO_ERR	FIFO안에 에러가 있음

7.12. UART 모뎀 상태 레지스터(UART_MSR)

UART_MSR은 모뎀 제어 신호들의 상태를 표시한다. 비트 정의는 다음 표와 같다.

비트 #	이름	설명
0	MSR_DCTS	CTS 신호가 변하면 1
1	MSR_DDSR	DSR 신호가 변하면 1
2	MSR_TERI	RI 신호가 변하면 1
3	MSR_DDCD	DCD 신호가 변하면 1
4	MSR_CTS	CTS 신호의 반대
5	MSR_DSR	DSR 신호의 반대
6	MSR_RI	RI 신호의 반대
7	MSR_DCD	DCD 신호의 반대

7.13. UART 스크래치 레지스터(UART_SCR)

UART_SCR은 쓰기 접근할 때 써지지 않고, 읽을 때는 항상 0값이 읽혀진다.

7.14. UART 나누기 레지스터 LSB(UART_DLL)

UART_DLL은 UART 나누기 레지스터의 LSB 값을 저장한다. 접근 할 때는 FCR_DLAB를 1로 설정하고 접근 해야 한다. 나누기 레지스터의 값은 다음과 같은 공식으로 구할 수 있다.

$$DIV = \text{FREQ} / (\text{BAUD} * 16)$$

이때 FREQ 값은 14745600이다.

7.15. UART 나누기 레지스터 MSB(UART_DLM)

UART_DLM은 UART 나누기 레지스터의 MSB 값을 저장한다. 접근 할 때는 FCR_DLAB를 1로 설정하고 접근 해야 한다.

7.16. DIO_DIR 레지스터

DIO_DIR 레지스터는 8비트 읽기/쓰기 레지스터 이다. 비트 정의는 다음과 같다.

표 16. ADC_STATUS 비트 정의

비트 #	이름	설명
7	DIO_DIR7	디지털 입출력 7 방향 설정 (1 : 출력, 0 : 입력)
6	DIO_DIR6	디지털 입출력 6 방향 설정 (1 : 출력, 0 : 입력)
5	DIO_DIR5	디지털 입출력 5 방향 설정 (1 : 출력, 0 : 입력)

비트 #	이름	설명
4	DIO_DIR4	디지털 입출력 4 방향 설정 (1 : 출력, 0 : 입력)
3	DIO_DIR3	디지털 입출력 3 방향 설정 (1 : 출력, 0 : 입력)
2	DIO_DIR2	디지털 입출력 2 방향 설정 (1 : 출력, 0 : 입력)
1	DIO_DIR1	디지털 입출력 1 방향 설정 (1 : 출력, 0 : 입력)
0	DIO_DIR0	디지털 입출력 0 방향 설정 (1 : 출력, 0 : 입력)

7.17. DIO_DOUT 레지스터

DIO_OUT 레지스터는 8비트 읽기/쓰기 레지스터 이다. 비트 정의는 다음과 같다.

표 17. ADC_STATUS 비트 정의

비트 #	이름	설명
7	DIO_OUT7	디지털 입출력 7 출력
6	DIO_OUT6	디지털 입출력 6 출력
5	DIO_OUT5	디지털 입출력 5 출력
4	DIO_OUT4	디지털 입출력 4 출력
3	DIO_OUT3	디지털 입출력 3 출력
2	DIO_OUT2	디지털 입출력 2 출력
1	DIO_OUT1	디지털 입출력 1 출력
0	DIO_OUT0	디지털 입출력 0 출력

7.18. DIO_DIN 레지스터

DIO_DIN 레지스터는 8비트 읽기 레지스터 이다. 비트 정의는 다음과 같다.

표 18. ADC_STATUS 비트 정의

비트 #	이름	설명
7	DIO_IN7	디지털 입출력 7 입력
6	DIO_IN6	디지털 입출력 6 입력
5	DIO_IN5	디지털 입출력 5 입력
4	DIO_IN4	디지털 입출력 4 입력
3	DIO_IN3	디지털 입출력 3 입력
2	DIO_IN2	디지털 입출력 2 입력
1	DIO_IN1	디지털 입출력 1 입력
0	DIO_IN0	디지털 입출력 0 입력

7.19. NVRAM

NVRAM은 전원이 나가도 데이터가 보존되는 메모리 영역으로 크기는 512K이다.

7.20 RTC 레지스터

RTC 레지스터는 다음과 같다.

표 19. RTC 레지스터 정의

어드레스	데이터								기능	범위
	B7	B6	B5	B4	B3	B2	B1	B0		
0x7FFFFF	10 Year				Year				Year	0-99
0x7FFFFE	X	X	X		Month				Month	01-12
0x7FFFFD	X	X			Date				Date	01-31
0x7FFFFC	X	FT	X	X		Day			Day	01-07
0x7FFFFB	X	X	10 Hour		Hour				Hour	00-23
0x7FFFFA	X	10 Minutes			Minutes				Minutes	00-59
0x7FFF9	OSC	10 Seconds			Seconds				Seconds	00-59
0x7FFF8	W	R	10 Century		Century				Control	00-39
0x7FFF0 - 0x7FFF7	Reserved									

8. 디바이스 드라이버

8.1. 초기화 함수

LKP-Serial 보드의 초기화 함수는 다음과 같다.

```
STATUS brockerInit(void);
```

리턴 값은 초기화가 성공하면 OK를 실패하면 ERROR이다.

8.2. UART 관련 함수

UART는 대부분 기능이 16550과 호환 되므로 16550 드라이버를 그대로 사용한다.

8.3. 디지털 입출력 관련 함수

디지털 입출력 관련 함수는 다음과 같다.

```
STATUS brockerDdirWrite(unsigned char dir);  
STATUS brockerDdirRead(unsigned char *dir);  
STATUS brockerDoutWrite(unsigned char dout);  
STATUS brockerDoutRead(unsigned char *dout);  
STATUS brockerDinRead(unsigned char *din);
```

brockerDdirWrite 함수는 디지털 입출력의 방향을 결정하는 함수 이다. 첫째 인자의 해당 비트가 1이면 출력이고 0이면 입력으로 설정된다. 성공하면 OK를 실패하면 ERROR를 돌려준다.

brockerDdirRead 함수는 디지털 입출력의 설정된 방향을 읽어오는 함수 이다. 첫째 인자로 넘겨지는 어드레스에 방향에 대한 데이터가 실려서 들어온다. 해당 비트가 1이면 출력이고, 0이면 입력이다. 성공하면 OK를 실패하면 ERROR를 돌려준다.

brockerDoutWrite 함수는 디지털 입출력의 출력 값을 쓰는 함수이다. 첫째 인자의 해당 비트가 DIO 값으로 출력된다. 성공하면 OK를 실패하면 ERROR를 돌려준다.

brockerDoutRead 함수는 디지털 입출력의 출력되고 있는 값을 읽어오는 함수 이다. 첫째 인자로 넘겨지는 어드레스에 출력에 대한 데이터가 실려서 들어온다. 성공하면 OK를 실패하면 ERROR를 돌려준다.

brockerDinRead 함수는 디지털 입출력의 입력 데이터를 읽어오는 함수 이다. 성공하면 OK를 실패하면 ERROR를 돌려준다.

8.4. NVRAM 관련 함수

NVRAM 관련 함수는 다음과 같다.

```
STATUS brockerNvramRead(char *buf, int size, int offset);  
STATUS brockerNvramWrite(char *buf, int size, int offset);
```

brockerNvramRead 함수는 NVRAM에서 값을 읽는 함수 이다. 첫째 인자는 데이터를 넣을 버퍼이고, 둘째 인자는 크기이고, 셋째 인자는 NVRAM의 base에서 offset 이다.

brockerNvramWrite 함수는 NVRAM에 값을 쓰는 함수 이다. 첫째 인자는 데이터를 넣을 버퍼이고, 둘째 인자는 크기이고, 셋째 인자는 NVRAM의 base에서 offset 이다.