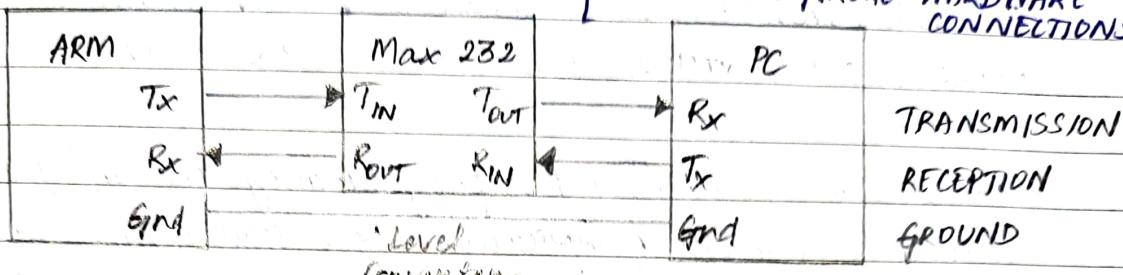


LECTURE 20 - VART

(UNIVERSAL ASYNCHRONOUS RECEIVER AND TRANSMITTER)

- * ARM IS PC SERIAL COMMUNICATION: [each bit of data will be sent serially one after another.]
- ↳ PARALLEL COMMUNICATION [FASTER SPEED / MORE HARDWARE CONNECTIONS]



Common ground of sharing

RS232

Logic 1 - V_{cc} (3.3V)

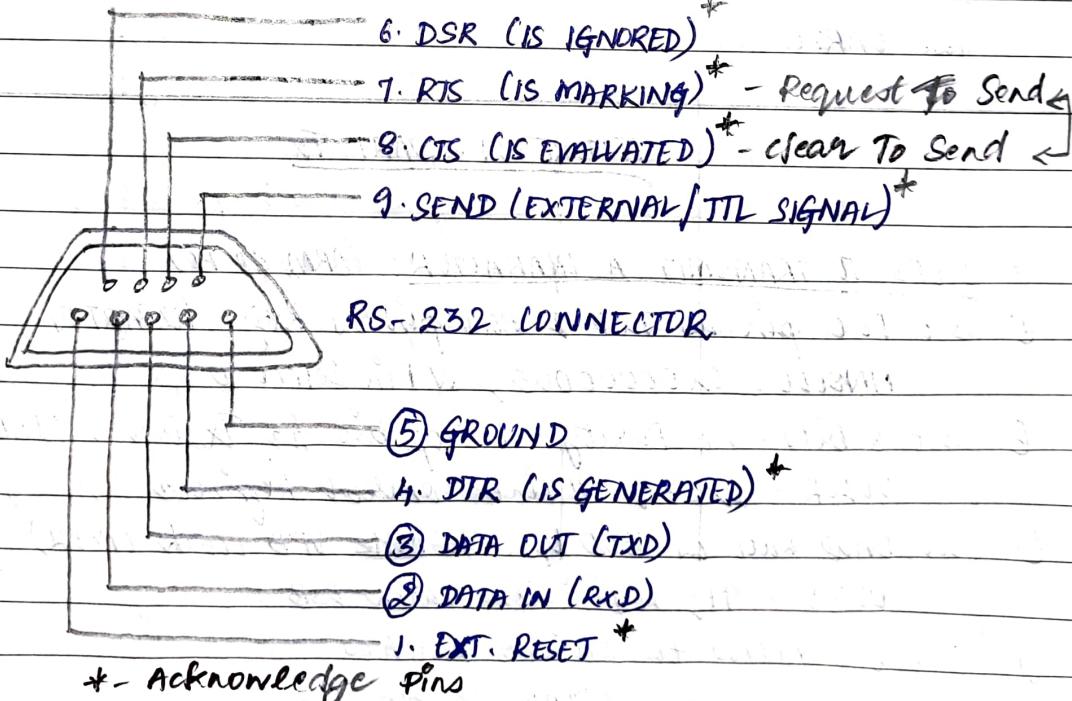
Logic 1 - -3V to -15V

Logic 0 - 0V

Logic 0 - +3V to +15V

[error correction mechanism]

- * DB9 CONNECTOR DETAILS: [Hyper Terminals Software - Virtual Terminal]



- Q. Why do we have a range of voltage level for RS232 as opposed to TTL?
- A. The cable length between ARM and PC can be 10 m. Therefore, to reduce voltage drop in the data cable over this distance, we have such a range.

* VARTO AND VART1 :

VARTO

P0.0/TXDO/PWM1 [19]
 P1.31/FRT [20]
 P0.1/RXDO/PWM3/EINT0 [21]

VART1

[34] P0.9/RXD1/PWM6/EINT3
 [33] P0.8/TXD1/PWM5

VARTO can also be used for programming the microcontroller in boot load mode of operation. Boot loader is a small program that is already burnt in the microcontroller and instead of using an external programmer, program file is supplied through the serial port via the boot loader.

Boot loader will receive the data coming through the serial port and write to the actual flash memory of the microcontroller. For the boot loader to receive the file, we use VARTO.

LECTURE 21 - VART Tx

* STEPS TO TRANSMIT A CHARACTER: (ARM TO PC)

- ① Set P0.0 pin as TxD & P0.1 pin as RxD - (V.SARTO)
 $\text{PINSEL0} = 0x00000005$; // Pin Select 0
- ② Set 8 bits - no Parity - 1 Stop bit for Txision & DLAB = 1
 $\text{VOLR} = 0x83$; // Line Control Register
- ③ Set BAUD Rate as 9600 bps - 15 MHz VPB Clock (PCLK)
 $\text{VDLL} = 91$; // Division Latch LSB
- ④ Disable Access to Divisor Latches
 $\text{VDLR} = 0x03$; // DLAB = 0 Bit - 1
- ⑤ Wait until VARTO ready to send character
 $\text{while } ((\text{VOLSR} \& 0x20))$; // Line status Register - THRE
- ⑥ Send a character to VART Transmit Register
 $\text{VTHR} = 'a'$; // Transmit Holding Register

* STEP 1 - PINSEL0 = 0x00000005;

→ PINSEL0 Register -

Bit	Symbol	Value	Function	Reset value
1:0	P0.0	00	GPIO Port 0.0	0
		01	TxD (UART0)	
		10	PWM1	
		11	Reserved	
3:2	P0.1	00	GPIO Port 0.1	0
		01	RxD (UART0)	
		10	PWM3	
		11	Reserved	
17:16	P0.8	00	GPIO Port 0.8	0
		01	TxD (UART1)	
		10	PWM4	
		11	Reserved or AD 1.1	
19:18	P0.9	00	GPIO Port 0.9	0
		01	RxD (UART1)	
		10	PWM6	
		11	EINT3	

PINSEL0 = 0x00000005; //UART0

PINSEL0 = 0x00050000; //UART1

* STEP 2 - VOLCR = 0x83;

→ Parameters - Serial Communication -

① Data Length - Bits for Tx or Rx (8 bits)

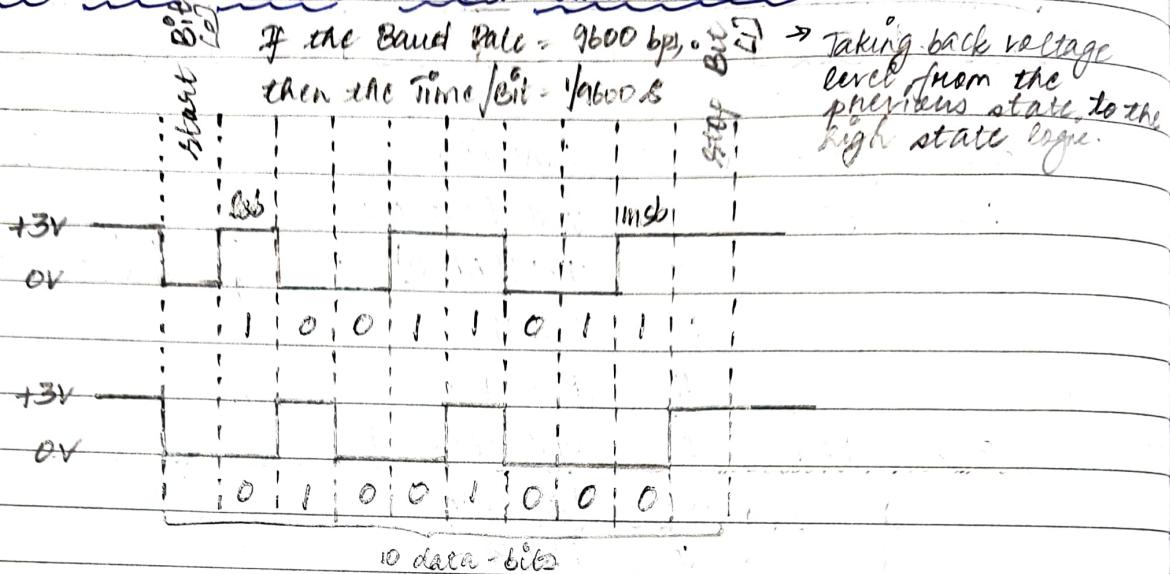
② Parity Bit - (Eg. 1110 - odd, 0101 - even) (Error correction)

③ Stop Bits

④ Baud Rate - Number of bits Tx/Rx per second (bps) [9600 bps]

PTD

→ Data Frame Format - Serial Communication -



→ VOTER (Line Control Register) - (32-bit)

Bit	Symbol	Value	Description	Reset Value
5:0	Word length	00	5 bit character length	0
	Length Select	01	6 bit character length	
	Select	10	7 bit character length	
		11	8 bit character length	
2	stop bit	0	1 stop bit	0
	Select	1	2 stop bits (1.5 if VOTER[5:0]=00).	
3	Parity enable	0	Disable parity generation & checking	0
		1	Enable parity generation & checking.	
5:4	Parity Select	00	odd parity: number of 1s in the transmitted character and the attached parity bit will be odd.	0
		01	Even parity: Number of 1s in the transmitted character and the attached parity bit will be even.	
		10	Forced "1" stick parity.	
		11	Forced "0" stick parity.	
6	Break control	0	Disable break transmission.	0
		1	Enable break transmission. output pin VARIO/TxD is forced to logic 0 when VOTER[6] is active high.	
7	Divisor latch Access E&T (DLAB)	0	Disable access to Divisor Latches.	0
		1	Enable access to Divisor Latches.	

↓
Enable access to write in DTR/DSR

VOLCR - DLAB3	1	0	0	0	0	0	1	1
	8						3	

* STEP 3 - VODLL = 97:

VARTO divisor latch
→ MSB - Bit

Table - VARTO divisor latch LSB register (VODLL - address 0xED00 C000, when DLAB=1) bit description

Bit	Symbol	Description	Reset Value
7:0	DLL	The VARTO divisor latch LSB Register, along with the VODLM register, determines the baud rate of the VARTO.	0x01

VARTO divisor latch
→ MSB - Bit

Table - VARTO divisor latch MSB register (VODLM - address 0xED00 C004, when DLAB=1) bit description

Bit	Symbol	Description	Reset Value
7:0	DLM	The VARTO divisor latch MSB Register, along with the VODLL register, determines the baud rate of the VARTO.	0x00

peripheral for generating the clock
for baud rate

→ Baud Rate calculation - Fractional Baud Rate Generator -

$$\text{VARTO baudrate} = \frac{\text{PCLK}}{16 \times (16 \times \text{VODLM} + \text{VODLL})} \times \frac{\text{MulVal}}{(\text{MulVal} + \text{DivAddVal})}$$

(Default value of VODLL = 0x01)

→ Example 1 -

PCLK = 30 MHz and Required Baud Rate is 9600 bauds.

start with VODLM = 0, DIVADDVAL = 0 and MULVAL = 1

We have PCLK = 30 MHz = 30×10^6 Hz

We get $VODLL = 195.3125$, since it must be an integer, we use 195, i.e., $VODLL = 195$.

On substituting $VODLL = 195$ in Baud Rate equation, we get -

$$\text{Baud Rate} = 9615.38, \text{Error} = +15.98 \text{ from } 9600.$$

Error - as low as possible by adjusting MULVAL and DIVADDVAL. Maximum value of MULVAL and DIVADDVAL is 15.

Substituting $MULVAL = 15$ and $DIVADDVAL = 1$, we get $VODLL = 183$

Substituting $VODLL = 183$ in Baud Rate equation, we get -

$$\text{Baud Rate} = 9605.53, \text{i.e. } \approx 9605!$$

The final configuration is as follows -

$$PCLK = 30 \times 10^6 \text{ Hz}$$

$$VODLM = 0$$

$$MULVAL = 15$$

$$DIVADDVAL = 1$$



Example 2 -

PCLK = 60 MHz and Required Baud Rate is 9600 bauds.

Start with $VODLM = 0$, $DIVADDVAL = 0$ & $MULVAL = 1$ with $PCLK = 60 \times 10^6 \text{ Hz}$

We get $VODLL = 390.625$ (i.e. ≈ 390). But VODL is 8-bit!

so, we set $VODLM = 1$ and find $VODLL$. New $VODLL = 135$

Substituting $VODLM$ and DLL in Baud Rate equation, we get -

$$\text{Baud Rate} = 9590.89, \text{Error} = -9.21$$

Substituting $MULVAL = 15$, we find that New $VODLL = 110$

Substituting New $VODLL$ in Baud Rate equation, we find that -

$$\text{Baud Rate} = 9605.53 \text{ which is } \approx 9605!$$

Hence, the final configuration is as follows-

$$\text{PCLK} = 60 \times 10^6 \text{ Hz}$$

$$\text{UDLL} = 110$$

$$\text{UDLM} = 1$$

$$\text{MULVAL} = 15$$

$$\text{DIVADDVAL} = 0$$

→ Example 3 -

USART0 baudrate = 9600 bps

$$\text{PCLK} = 15 \text{ MHz}$$

$$\text{UDLM} = 0$$

$$\text{MulVal} = 1$$

$$\text{DivAddVal} = 0$$

$$\text{UDLL} = ?$$

$$\text{UDLL} = 97$$

* STEP 4 - VOLCR = 0x03; - Disable the access for Divisor Latch

VOLCR = 0x03	0	0	0	0	0	0	1	1
	0	0	0	0	0	0	1	1

* STEP 5 - while ((VOLSR & 0x20) == 0); - checks whether shift register is empty or not
It can also be written as while (! (VOLSR & 0x20));

→ VOLSR (line status Register) -

Bit	Symbol	Value	Description	Reset value
0	Receiver data ready (RDR)	0	VOLSR0 is set when the VDRBR holds an unread character and is cleared when the VARTO RBR FIFO is empty.	0
		1	VDRBR is empty.	
1	Overrun	1	VDRBR contains valid data.	
			The overrun error condition is set	0

Error
(DE)

as soon as it occurs. An VOLSR read clears VOLSR1. VOLSR1 is set when VARTD RSR has a new character assembled and the VARTD RBR FIFO is full. In this case, the VARTD RBR FIFO will not be overwritten and the character in the VARTD RSR will be lost.

0

Overrun error status is inactive.

1

Overrun error status is active.

2 Parity
error
(PE)

When the parity bit of a received character is in the wrong state, a parity error occurs. An VOLSR read clears VOLSR[2]. Time of parity error detection is dependent on VDFCR[6].

0

Note - A parity error is associated with the character at the top of the VARTD RBR FIFO.

0

Parity error status is inactive.

1

Parity error status is active.

5 Transmitter
Holding
Register
Empty
(THRE)

0

THRE is set immediately upon detection of an empty VARTD THR and is cleared on a VOTHR write.

1

VOTHR contains valid data

✓

VOTHR is empty.

VOLSR & CX20	0	0	THRE	0	0	0	0	0	0
	2/0						0		

* STEP 6 - VOTHR = 'a';

ASCII value of character will be sent to parallel-to-serial shift register. Based on clock given, character will be sent out serially through the TxPin.

shadow
register

LECTURE 22 - VART Rx

CLASSMATE

Date _____

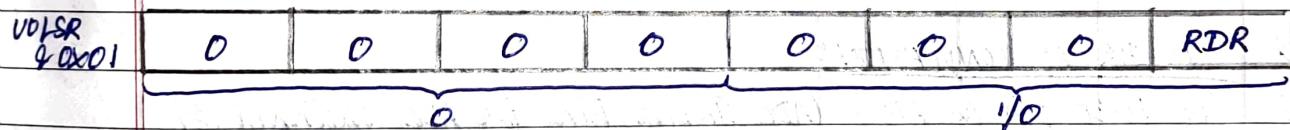
Page _____

07

* STEPS TO RECEIVE A CHARACTER : (PC TO ARM)

- ① Set P0.0 pin as Tx.D & P0.1 pin as Rx.D - (USARTD)
PINSEL0 = 0x00000005; // Pin Select 0
- ② Set 8 bits - no Parity - 1 stop bit for Txision & DLAB = 1
VOLCR = 0x83; // Line Control Register
- ③ Set BAUD Rate as 9600 bps - 15 MHz VFB clock (PCLK)
VODLR = 91; // Divider Latch LSB
- ④ Disable Access To Divider Latches
VOLCR = 0x03; // DLAB = 0 Bit = 1
- ⑤ Wait until VART is ready with the received data
while (! (VOLSR & 0x01)); // Line Status Register - RDR
- ⑥ Read the received data from VDRBR
data = VDRBR; // Receive Buffer Register

* STEP 5 - while (! (VOLSR & 0x01));



* REGISTER SUMMARY - VART:

	7	6	5	4	3	2	1	0
VDRBR								
VOTHR								
VODLR								
VODMR								
VDFCR	Rx Trigger					Tx Fifo Reset	Rx Fifo Rst	Rx Enable
VOLCR	DLAB	Set Break	sticky Parity	Even Parity	Parity Enab.	No. of StopBit	Word length Select	
VOLSR	RX PIFDEN	TEINT	THRE	BI	FE	PE	OE	RDR
VDFDR		mulVal					DivAddVal	
ODTER	TXEN							

LECTURE 23 - ADC

(ANALOG - TO - DIGITAL CONVERTER)

LPC2148 contains only one ADC, i.e., ADC0. The resolution of the ADC is 10-Bit, i.e., the smallest possible value that can be measured by the ADC is 10-bit. Voltage is measured from 0V to $\pm 3.3V$. The ADC can be 2^{10} values, therefore the voltage can be divided from 0V to $\pm 3.3V$ in 2^{10} steps.

Therefore, smallest voltage level measured = $\frac{\pm 3.3V}{10}$

The ADC has 6 channels, i.e., six inputs can be taken but only one output will be converted to corresponding digital data for which we use a multiplexer. Each channel is referred by the terminology of ADD.14 & ADD.6:7 ; ADD.5 is not present.

★ STEPS FOR AD CONVERSION:

- ① Set P0.30 pin as ADD.3 (ADC Input Pin for ADC channel 3)
 $\text{PINSEL1} \&= 0xFFFFFFFF; // x0101xx xxxx xxxx ... xxxx}$
 $\text{PINSEL1} |= 0x10000000; // x01xxxx xxxx xxxx xxxx ... xxxx}$
- ② Enable Power / clock to ADC0
 $\text{PCONP} |= (\text{unsigned long})(0x00000001) \ll 12; // \text{Power control for peripheral}$
- ③ Select ADC channel 3 (ADD.3)
 $\text{ADOCR} |= 0x00000008; // \text{control Register (bits 7:0)}$
- ④ Set ADC clock as IPB or (PCLK) /8
 $\text{ADOCR} |= 0x00000700; // (\text{CLKDIV} = 00000111) (\text{bits 15:8})$
- ⑤ Disable Burst mode
 $\text{ADOCR} \&= 0xFFFFEFFF; // \text{Burst} = 0 (\text{bit 16})$
- ⑥ Set Resolution as 10 bit in 11 clock cycles
 $\text{ADOCR} \&= 0xFFFF1FFF; // \text{CLKS} = 000 (\text{bits 19:17})$
- ⑦ Activate ADC module
 $\text{ADOCR} |= 0x00200000; // \text{PDN} = 1 (\text{bit 21})$

⑨ Start the conversion Now

ADOCR 1 = 0x01000000; //START = 001 (Bits 26:24)

⑩ Wait ADC conversion to complete (global data register)

while ((ADOGDR & 0x80000000) == 0); //DONE = 1 (bit 31)

⑪ Read ADC Data Register

val = (ADD.R3 >> 6) & 0x000003FF; //RESULT = 10 Bit Data (15:6)

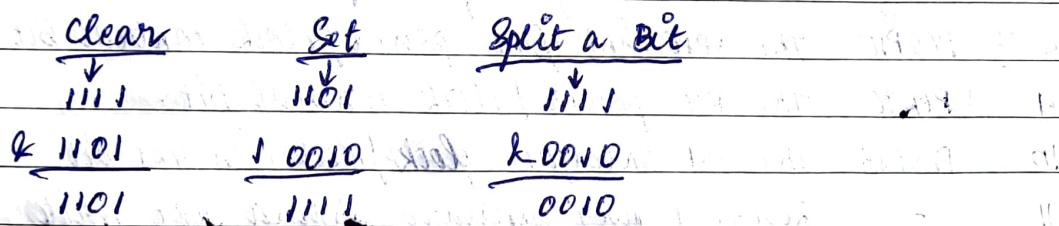
⑫ Deactivate ADC module (No Start)

ADOCR 4 = 0xF8FFFFFF; //START = 000 (Bits 26:24)

* STEP 1 - PINSEL1 <29:28> = 01 :

PINSEL1 & = 0xFFFFFFFF; //ND1 1111 1111 1111 1111 1111 1111 1111

PINSEL1 >= 0x10000000; //0000 0000 0000 0000 0000 0000 0000 0000



Bit clear/set

11100101 \times 1 000001010
11111100 \rightarrow 11100110
 11100000
1 000001010
 11100010

* STEP 2 - PCONP 1 = (unsigned long)(0x00000001) << 12;

→ PCONP (Power Control for Peripherals) -

PTD

Bit	Symbol	Description	Reset Value
0	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
1	PC0TMO	Timer / Counter 0 power / clock control bit.	1
2	PC1TMI	Timer / Counter 1 power / clock control bit.	1
3	PCVARTD	VARTD power / clock control bit.	1
4	PCVARTI	VARTI power / clock control bit.	1
5	PCPWMD	PWMD power / clock control bit.	1
6	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
7	PC12CO	The I ² C interface power / clock control bit.	1
8	PCSP10	The SPI0 interface power / clock control bit.	1
9	PERTC	The RTC power / clock control bit.	1
10	PCSP11	The SSP interface power / clock control bit.	1
11	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
12	PCADO	A/D converter 0 (ADC0) power / clock control bit.	1

Note - clear the PDN bit in the ADDCR before clearing this bit and set this bit before setting PDN.

1	0	0	0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---

* STEP 3 - ADDCR If. 0x00000008;

→ ADCR (Control Register) -

Bit	Symbol	Value	Description	Reset value
7:0	SEL		<p>Selects which of the ADO.1:0/ADI.7:0 pins is (are) to be sampled and converted. For ADO, bit 0 selects pin ADO.0, and bit 1 selects pin ADO.1.</p> <p>In software-controlled mode, only one of these bits should be 1. In hardware scan mode, any value containing 1 to 8 ones. All zeroes is equivalent to 0x01.</p>	0x01

00000001	ADO.0 (ADCO channel 0)
00000010	ADO.1 (ADCO channel 1)
00000100	ADO.2 (ADCO channel 2)
00001000	ADO.3 (ADCO channel 3)
00010000	ADO.4 (ADCO channel 4)
00100000	ADO.5 (ADCO channel 5)
01000000	ADO.6 (ADCO channel 6)
10000000	ADO.7 (ADCO channel 7)

* STEP 4 - ADCCR \leftarrow 0x00000100;
 → ADCCR (control Register) -

15:8	UKDIV	The VFB clock (PCLK) is divided by (this value plus one) to produce the clock for the A/D converter, which should be less than or equal to 4.5 MHz. Typically, software should program the smallest value in this field that yields a clock of 4.5 MHz or slightly less, but in certain cases, (such as a high-impedance analog	0
------	-------	---	---

source) a slower clock may be desirable:

00000000 ADC Clock = (PCLK)/1

00000001 ADC Clock = (PCLK)/2

00000010 ADC Clock = (PCLK)/3

00000011 ADC Clock = (PCLK)/4

00000100 ADC Clock = (PCLK)/5

00000101 ADC Clock = (PCLK)/6

00000110 ADC Clock = (PCLK)/7

00000111 ADC Clock = (PCLK)/8

STEP 5 - ADDCR 9 = 0xFFFFFFF ;
 → ADDCR (control Register) -

16

BURST

The AD converter does repeated conversions at the rate selected by the CKS field, scanning (if necessary) through the pins selected by 1s in the SEL field. The first conversion after the start corresponds to the least-significant 1 in the SEL field, then higher numbered 1-bits (pins) if applicable. Repeated conversions can be terminated by clearing this bit, but the conversion that's in progress when this bit is cleared will be completed.

Important - START bits must be 000 when BURST=1 or conversions will not start.

0	Conversions are software controlled and require 11 clocks.
---	--

* STEP 6 - ADDCR 2 = 0xFFFFFFF;

→ ADDCR (Control Registers) -

19:17	CLKS.	This field selects the number of clocks used for each conversion in burst mode, and the number of bits of accuracy of the result in the RESULT bits of ADDR, between 11 clocks (10 bits) and 4 clocks (3 bits).	000
-------	-------	---	-----

000 11 clocks / 10 bits

001 10 clocks / 9 bits

010 9 clocks / 8 bits

011 8 clocks / 7 bits

100 7 clocks / 6 bits

101 6 clocks / 5 bits

110 5 clocks / 4 bits

111 4 clocks / 3 bits

* STEP 7 - ADDCR 1 = 0x00200000;

21	PDN	1	The A/D converter is operational.	0
		0	The A/D converter is in power-down mode.	

* STEP 8 - ADDCR 1 = 0x01000000;

→ ADDCR (Control Registers) -

26: 24

START

When this BURST bit is 0, these bits control whether and when an A/D conversion is started:

000	No start (this value should be used when clearing PDN to 0).
001	start conversion now.
010	start conversion when the edge selected by bit 27 occurs on P0.16/EINT0/MAT0.2/CAP0.2 pin.
011	start conversion when the edge selected by bit 27 occurs on P0.22/TD3/CAP0.0/MAT0.0 pin.
100	start conversion when the edge selected by bit 27 occurs on MAT0.1.
101	start conversion when the edge selected by bit 27 occurs on MAT0.3.
110	start conversion when the edge selected by bit 27 occurs on MAT1.0.
111	start conversion when the edge selected by bit 27 occurs on MAT1.1.

* STEP 9 - while (ADGDR & 0x80000000) $\neq 0$:

→ ADGDR (global data Register) -

Bit	Symbol	Description	Reset Value
5:0	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
15:6	RESULT	When DONE is 1, this field contains a binary fraction representing the voltage on the Ain pin selected by the SEL field, divided by the voltage on the VDDA pin (V/V_{REF}). Zero	NA

in the field indicates that the voltage on the AIN pin was less than, equal to, or close to that on VREF, while 0x3FF indicates that the voltage on AIN was close to, equal to, or greater than that on VREF.

23:16

- Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.

26:24

CHN

These bits contain the channel from which the RESULT bits were converted (e.g. 000 identifies channel 0, 001 channel 1...).

29:27

- Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.

30. OVERRUN

This bit is 1 in burst mode if the results of one or more conversions was (were) lost and overwritten before the conversion that produced the result in the RESULT bits. This bit is cleared by reading this register.

31

DONE

This bit is set to 1 when an A/D conversion completes. It is cleared when this register is read and when the ADCR is written. If the ADCR is written while a conversion is still in progress, this bit is set and a new conversion is started.

* STEP 10 - val = (ADDDR3 >> 6) & 0x000003FF;

→ ADDDR3 (Data Register) -

Bit	Symbol	Description	Reset Value
5:0	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
15:6	RESULT	When DONE is 1, the field contains a binary fraction representing the voltage of the A_{in} pin, dividing by the voltage on the V_{REF} pin (V/V_{REF}). Zero in the field indicates that the voltage on the A_{in} pin was less than, equal to, or close to that on V_{SS} , while 0x3FF indicates that the voltage on A_{in} was close to, equal to, or greater than that on V_{REF} .	NA
29:16	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
30	OVERRUN	This bit is 1 in burst mode if the results of one or more conversions was (were) lost and overwritten before the conversion that produced in the result in the RESULT bits. The bit is cleared by reading this register.	
31	DONE	The bit is set to 1 when an A/D conversion completes. It is cleared when this register is read.	NA

* STEP 11 - ADDCR & = 0xFFFFFFFF;

LECTURE 24 - EXTERNAL INTERRUPTS

Whenever a signal is given to a pin, the processor will identify that an external interrupt is generated, thereby automatically stopping the main program and interrupt service routine is executed.

In LPC2148, we have 9 external interrupt pins. It is named as -

EINT0 - P0.1, P0.16

EINT1 - P0.3, P0.14

EINT2 - P0.1, P0.15

EINT3 - P0.9, P0.20, P0.30

* STEPS TO CONTROL AN LED WITH EXTERNAL INTERRUPT:

① Configure P0.14 as EINT1

PINSEL0 <29:28> = 10

② Configure Edge sensitive / Level sensitive Interrupt for EINT1

EXTMODE <1> = 1 (Edge sensitive)

③ Configure Rising / Falling edge sensitive Interrupt for EINT1

EXTPOLAR <1> = 0 (Falling)

④ Clear EINT1 Interrupt Flag Bit

EXTINT <1> = 1

⑤ Configure EINT1 (Interrupt No. 15) as F1Q1IRQ

VICIntSelect <15> = 0 (IRQ)

⑥ Enable EINT1 Interrupt

VICIntEnable <15> = 1

⑦ Configure the Interrupt Number of Interrupt Request assigned to Vectored IRQ slot (IRQ slot 5)

VICVectCtrl5 <4:0> = 1111 (Decimal 15)

⑧ Enable Vectored IRQ slot (IRQ slot 5)

VICVectCtrl5 <5> = 1

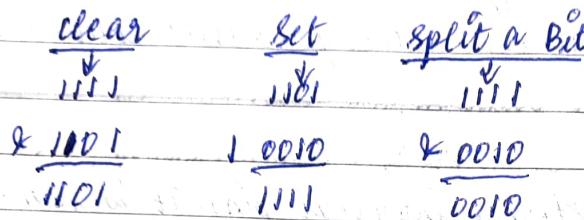
⑨ Configure ISR Vector Address to VIC IRQ slot

VICVectAddr5 = (unsigned int)Ext_ISR;

* STEP 1 - PINSEL0 <29:28> = 10

$$\text{PINSEL0} = (\text{PINSEL0} \& 0x(CFFFFFFF)) | (1 \ll 29);$$

First make Bit 28 & 29 as '0' then make Bit 29 as '1'



EXTERNAL INTERRUPT REGISTERS -

Table - External Interrupt registers

Name	Description	Access	Reset value	Address
EXTINT	The External Interrupt Flag Register contains interrupt flags for EINT0, EINT1, EINT2 and EINT3.	R/W	0	0xE01F 0140
INTNAKE	The Interrupt Wakeup Register contains four enable bits that control whether each external interrupt will cause the processor to wake up from Power-down mode.	R/W	0	0xE01F 0144
EXTMDDE	The External Interrupt mode Register controls whether each pin is edge- or level sensitive.	R/W	0	0xE01F 0148
EXTPOLAR	The External Interrupt Polarity Register controls which level or edge on each pin will cause an interrupt.	R/W	0	0xE01F 014C

* STEP 2 - EXTMODE $\langle 1 \rangle = 1$ (edge sensitive)

$\text{EXTMODE} = 0x2j$

Table - External Interrupt Mode register (EXTMODE - address 0x EDIF (148)) bit description

Bit	Symbol	Value	Description	Reset value
0	EXTMODE0	0	level-sensitivity is selected for EINT0.	0
		1	EINT0 is edge sensitive.	
1	EXTMODE1	0	level-sensitivity is selected for EINT1.	0
		1	EINT1 is edge sensitive.	
2	EXTMODE2	0	level-sensitivity is selected for EINT2.	0
		1	EINT2 is edge sensitive.	
3	EXTMODE3	0	level-sensitivity is selected for EINT3.	0
		1	EINT3 is edge sensitive.	
7:4	-	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA

+ STEP 3 - EXTPOLAR $\langle 1 \rangle = 0$ (Falling)

$\text{EXTPOLAR} b = 0xFFFFFFF0;$

Table - External Interrupt Polarity register (EXTPOLAR - address 0x EDIF (14C)) bit description

Bit	Symbol	Value	Description	Reset Value
0	EXTPOLARD	0	EINTD is low-active or falling-edge sensitive (depending on EXTMODED).	0
		1	EINTD is high-active or rising-edge sensitive (depending on EXTMODED).	
1	EXTPOLARI	0	EINTI is low-active or falling-edge sensitive (depending on EXTMODE1).	0
		1	EINTI is high-active or rising-edge sensitive (depending on EXTMODE1).	
2	EXTPOLAR2	0	EINT2 is low-active or falling-edge sensitive (depending on EXTMODE2).	0
		1	EINT2 is high-active or rising-edge sensitive (depending on EXTMODE2).	
3	EXTPOLAR3	0	EINT3 is low-active or falling-edge sensitive (depending on EXTMODE 3).	0
		1	EINT3 is high-active or rising-edge sensitive (depending on EXTMODE 3).	
7:4	-	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA

* STEP 4 - EXTINT <1> = 1

EXTINT |= 0x02;

Flag bit is cleared by writing '1' to it.

Table - External Interrupt Flag register (EXTINT - address 0XE01F (140) bit description

Bit	Symbol	Description	Reset value
0	EINT0	<p>In level-sensitive mode, this bit is set if the EINT0 function is selected for its pin, and the pin is in its active state. In edge-sensitive mode, this bit is set if the EINT0 function is selected for its pin, and the selected edge occurs on the pin.</p> <p>Up to two pins can be selected to perform the EINT0 function (see PD.1 and PD.16 description in "Pin configuration").</p> <p>This bit is cleared by writing a one to it, except in level sensitive mode when the pin is in its active state (e.g. if EINT0 is selected to be low level sensitive and a low level is present on the corresponding pin, this bit can not be cleared; this bit can be cleared only when the signal on the pin becomes high).</p>	0
1	EINT1	<p>In level-sensitive mode, this bit is set if the EINT1 function is selected for its pin, and the pin is in its active state. In edge-sensitive mode, this bit is set if the EINT1 function is selected for its pin, and the selected edge occurs on the pin.</p> <p>Up to two pins can be selected to perform the EINT1 function (see PD.3 and PD.14 description in "Pin configuration").</p> <p>This bit is cleared by writing a one to it, except in level sensitive mode when the pin is in its active state (e.g. if EINT1</p>	0

is selected to be low level sensitive and a low level is present on the corresponding pin, this bit can not be cleared; this bit can be cleared only when the signal on the pin becomes high).

2 EINT2

In level-sensitive mode, this bit is set if the EINT2 function is selected for its pin, and the pin is in its active state. In edge-sensitive mode, this bit is set if the EINT2 function is selected for its pin, and the selected edge occurs on the pin.

Up to two pins can be selected to perform the EINT2 function (see P0.7 and P0.15 description in "Pin Configuration")

This bit is cleared by writing a one to it, except in level sensitive mode when the pin is in its active state (e.g. if EINT2 is selected to be low level sensitive and a low level is present on the corresponding pin, this bit can not be cleared; this bit can be cleared only when the signal on the pin becomes high).

3 EINT3

In level-sensitive mode, this bit is set if the EINT3 function is selected for its pin, and the pin is in its active state. In edge-sensitive mode, this bit is set if the EINT3 function is selected for its pin, and the selected edge occurs on the pin.

Up to three pins can be selected to perform the EINT3 function (see P0.9, P0.20 and P0.30).

description in "Pin configuration").

This bit is cleared by writing a one to it, except in level sensitive mode when the pin is in its native state i.e.g. If TINT3 is selected to be low level sensitive and a low level is present on the corresponding pin, this bit can not be cleared; this bit can be cleared only when the signal on the pin becomes high).

1:4	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
-----	---	--	----

→ VIC Registers -

Table - VIC register map .

Name	Description	Access	Reset Value	Address
VICIntSelect	Interrupt Select Register. This register classifies each of the 32 interrupt requests as contributing to FIG or IRg.	R/W	0	0xFFFF F00C
VICIntEnable	Interrupt Enable Register. This register controls which of the 32 interrupt requests and software interrupts are enabled to contribute to FIG or IRg.	R/W	0	0xFFFF F010
VICVectCtrl0	Vector control 0 register. Vector Control Registers 0-15 each control one of the 16 reserved IRg slots. Slot 0 has the highest priority and slot 15 the lowest.	R/W	0	0xFFFF F200

VICVectCtrl1	Vector control 1 register.	R/W	0	0xFFFF F204
VICVectCtrl2	Vector control 2 register.	R/W	0	0xFFFF F208
VICVectCtrl15	Vector control 15 register.	R/W	0	0xFFFF F23C
VICVectAddr	Vector Address Register. When an IRQ interrupt occurs, the IRQ service routine can read this register and jump to the value read.	R/W	0	0xFFFF F030
VICVectAddr0	Vector address 0 register. Vector Address Registers 0-15 hold the addresses of the interrupt service routines (ISRs) for the 16 vectored IRQ slots.	R/W	0	0xFFFF F100
VICVectAddr1	Vector address 1 register.	R/W	0	0xFFFF F104
VICVectAddr2	Vector address 2 register.	R/W	0	0xFFFF F108
VICVectAddr15	Vector address 15 register.	R/W	0	0xFFFF F13C

STEP 5 - VIC Int Select & = 0xFFFF1FFF ; VICIntSelect $\ll 15$ > 0 (Rg)
Table- Interrupt Select Register /VICIntSelect - address 0xFFFF F000
bit allocation
Reset value: 0x0000 0000

bit	7	6	5	4	3	2	1	0
symbol	VARTI	VARTD	TIMER1	TIMER0	ARMCore1	ARMCore0	-	WDT
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	symbol	value	Description	Reset value
31:0	See VICIntSelect bit allocation table	0	The interrupt request with this bit number is assigned to the IRQ category.	0
		1	The interrupt request with this bit number is assigned to the FIQ category.	

* STEP 6 - VICIntEnable $\ll 15 = 1$

$$\text{VICIntEnable} = 1 \ll 15;$$

Table - Interrupt Enable register (VICIntEnable - address 0xFFFFF010) bit allocation

Reset value: 0x0000 0000

[same as Interrupt Select Register]

bit	symbol	Description	Reset value
31:0	See VICIntEnable bit allocation table	When this register is read, it indicates interrupt requests or software interrupts that are enabled to contribute to FIQ or IRQ.	0
		When this register is written, ones enable interrupt requests or software interrupts to contribute to FIQ or IRQ, zeroes have no effect.	

* STEP 7 - VICVectCntl5 <4:0> = 1111 (Decimal 15)

Table - Vector control registers 0-15 (VICVectCntl 0-15 - 0xFFFF F200-23C) bit description

Bit	Symbol	Description
4:0	int-request	The number of the interrupt request or software interrupt assigned to this vectored IRQ slot. As a matter of good programming practice, software should not assign the same interrupt number to more than one enabled vectored IRQ slot. But if this does occur, the lower numbered slot will be used when the interrupt request or software interrupt is enabled, classified as IRQ, and asserted.
5	IRQ slot-en	When 1, this vectored IRQ slot is enabled, and can produce a unique ISR address when its assigned interrupt request or software interrupt is enabled, classified as IRQ, and asserted.
31:6	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.

* STEP 8 - VICVectCntl5 <5> = 1

VICVectCntl5[1] = 1 & 5;

* STEP 9 - VICVectAddr5 = (unsigned int) Ext_ISR;
ISR should have "irq" as the keyword.
void Ext_ISR() - irq

→ VIC Vector Address Registers (VICVectAddr0-15)

Table - Vector Address registers (VICVectAddr 0-15 - addresses 0xFFFF F100-13C) bit description

Bit	Symbol	Description	Reset Value
31:0	IRQ-Vector	When one or more interrupt request or software interrupt is (are) enabled, classified as IRQ, asserted, and assigned to an enabled vectored IRQ slot, the value from this register for the highest-priority such slot will be provided when the IRQ service routine reads the Vector Address register - VICVectAddr	0x0000 0000

* EXTERNAL INTERRUPT REGISTERS :

31-4	3	2	1	0
EXTMODE	EXTMODE3	EXTMODE2	EXTMODE1	EXTMODE0
EXTPolar	EXTPOLAR3	EXTPOLAR2	EXTPOLAR1	EXTPOLAR0
EXTINT	EINT3	EINT2	EINT1	EINT0

* VIC CONTROL AND ADDRESS REGISTERS :

31-6	5	4-0
VICVectCntl0-15	IRQslot-en	Int-request
		31-0
VICVectAddr0-15		IRQ-Vector

LECTURE 25 - NXP DAC

(Digital - To - Analog Converter)

We use DAC to generate analog signals, say sound for an example. It can also be used to generate sinusoidal waves, triangular waves, etc. This is the simplest peripheral in NXP LPC2148 ARM. It contains only one DAC, i.e., one channel output.

P0.25/AD0.4/ADOUT - DAC Pin

STEPS FOR DAC CONFIGURATION :

① Configure P0.25 as DAC pin (ADOUT)

PINSEL1 |= (1<<19);

② Configure value of DAC output

DACKR = (value << 6);

+ STEP 1 - PINSEL1 |= (1<<19);

Bit	Symbol	Value	Function	Reset Value
19:18	P0.25	00	GPIO Port 0.25	0
		01	AD0.4	
		10	Reserved or Adout (DAC)	
		11	Reserved	

+ STEP 2 - DACKR = (value << 6); [10-bit value input]

Table - DAC Register (DACKR - address 0x1006 0000) bit description

Bit	Symbol	Value	Description	Reset Value
5:0	-		Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
15:6	VALVE		After the selected settling time after this field is written with a new VALVE, the voltage on the ADOUT pin (with respect to VSSA) is $VALVE / 1024 * VREF$.	0

16	BIAS	0	The settling time of the DAC is 1 μ s max, and the maximum current is 100 μ A.	0
		1	The settling time of the DAC is 2.5 μ s and the maximum current is 350 μ A.	
31:17	-		Reserved, user software should not write NA ones to reserved bits. The value read from a reserved bit is not defined.	

* DAC - REGISTER SUMMARY:

	31	17	16	15	6	5	0
DACR			BIAS		VALVE		