Roll No.													

PRESIDENCY UNIVERSITY
BENGALURU

SCHOOL OF ENGINEERING

TEST –1

Semester: 2019-2020

Course Code: ECE 308 **Date**:30.09.2019

Course Name: Embedded System design using ARM **Time**: 1 hour

Max Marks:40

Program & Sem: Btech & 7th semester **Weightage**:20%

Instructions:

(i) Read Questions carefully and answer accordingly

(ii) Scientific and Non- programmable calculators are permitted

(iii) This question paper contains two pages

Part A

Answer all the Questions. Each question carries 2 marks. **(5Qx2M=10)**

(CO: 1, Knowledge Level)

1. Difference between Embedded and VLSI design.
2. List out the types of Hardware components of Embedded system
3. What are the types of embedded processors?
4. Define embedded system.
5. Give the significance of Embedded C programming

Part B

Answer all the Questions. Each question carries 5 marks. **(4Qx5M=20)**

(CO: 1, Comprehension Level)

1. Briefly discuss about classification of embedded system
2. Difference between Von Neumann and Harward architecture
3. Illustrate the design challenges of embedded system design

4. Explain different types of memories in embedded system

Part C

Answer one Question. One question carries 10 marks. **(1Qx10M=10)**

(CO: 2, Comprehension Level)

1. Describe in Detail about ARM architecture. Compare and contrast of Arm processor with respect to other types of microcontrollers.

SCHOOL OF-ENGINEERING

Semester: 2019-2020

Course Code: ECE 308

Course Name: Embedded System Design using ARM

Date:30.09.19

Time: 1 hour

Max Marks:40

Weightage: 20%

Extract of question distribution [outcome wise & level wise]

.NO	C.O.NO	Unit/Module Number/Unit /Module Title	Memory recall type [Marks allotted] Bloom's Levels			Thought provoking type [Marks allotted] Bloom's Levels			Problem Solving type [Marks allotted]			Total Marks
			K			C			A			
-5	CO 1	Module 1	10									10
-2	CO 1	Module 1				5	5					10
-4	CO 1	Module 1				5	5					10
	CO 2	Module 2				10						10
	Total Marks											40

K =Knowledge Level C = Comprehension Level, A = Application Level

Note: While setting all types of questions the general guideline is that about 60%

Of the questions must be such that even a below average students must be able to attempt, About 20% of the questions must be such that only above average students must be able to attempt and finally 20% of the questions must be such that only the bright students must be able to attempt.

[I hereby certify that All the questions are set as per the above guide lines Mr.J Britto Pari]

Reviewers' Comments

(1) Part 'B' (Qn-01) may take about 15 minute
↳ it should be balanced

(2) For Part 'C' — In solution no comparison with other controllers given but in Q.P. it has been asked.

Annexure- II: Format of Answer Scheme

GAIN MORE KNOWLEDGE REACH GREATER HEIGHTS

SCHOOL OF ENGINEERING

SOLUTION

Semester: 2019-2020

Course Code: ECE 308

Course Name: Embedded System Design using ARM

Date:30.09.19

Time: 1 hour

Max Marks:40

Weightage: 20%

Part A (5Q x 2M = 10 Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
1	**Embedded System** concern more about software programming like C,C++ and how to write efficient code with lowest memory usage and memory management. nearly all embedded systems have a limited storage utilities. **VLSI** concern more	2	2

	about Hardware Design Language HDL like vhdl and verilog which is used to describe the digital components in terms of lines of code. the art of coding using HDL is totally different than the software programming. we concern more about making the code represent some definite design. not just a behavioral syntax as similar to normal software programming		
2	Processor, memory, Input and output devices and application specific circuits	2	2
3	Embedded processor Application specific Instruction set processor Single Processor SOC	2	2
4	Process of embedding software into hardware	2	2
5	Time consuming is less.	2	2

Part B (4Q x 5M = 20Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
1	**Real-time embedded system:** Real-time systems are those which give a quick response to critical situations. They are used in military, medical and industrial applications. Engineers working in these systems have high demand is current days. To develop the real-time embedded system we require timing analysis, multitasking design, debugging, cross-platform testing and architecture design. In these systems, quick response is very important. Better hardware is also used in these systems to avoid failure in performance. Real-time systems control the external environment by input & output interfaces and sensors. The external environment includes human and other animals. Some **examples of real-time embedded systems**	2+3=5	5

include:

- Controlling heat, elevators, lights, and doors in buildings
- Robots
- Traffic control system including railway tracks, airspace, shipping lines, highways
- Radio, satellite and telephone communication
- Patient monitoring system
- Radiation therapy system in the hospital
- Computer games
- Multimedia systems which consist of video audio, text and graphics interfaces
- Military usage that includes tracking, weapons, and command & control

Standalone embedded system:

This type of embedded system works for itself as a device without needing any interconnected computer. It can take data in the form of analog or digital signals. This system first process data and then outputs data by displaying on the screen. It can also output data to any attached device. Examples of standalone embedded systems include:-

- Microwave ovens
- Digital cameras
- Mp3 players
- Video game consoles
- Temperature measurement systems

Networked embedded system:

Networked embedded systems are those systems which are connected to the network to give output to the attached resources. The devices in the networked embedded system are connected to the network with network interfaces. The network can be either a local area network (LAN) or a wide area network (WAN). The connection in networked embedded systems can be wireless or wired. This embedded system is fast and emerging its popularity over days. The embedded web server is that which is connected to network devices and is controlled by the web browser also. Example of this is the office security system. In office security system, different sensors (light sensors, smoke sensors or motion detectors) are networked together through LAN and controlled over the WAN (internet).

Mobile embedded system:

Mobile embedded systems are limited in resources including memory. Examples of mobile embedded systems include:-

- Personal digital assistants (PDA)
- Cellular phones
- Mp3 players
- Digital cameras,

Small-scale embedded system:

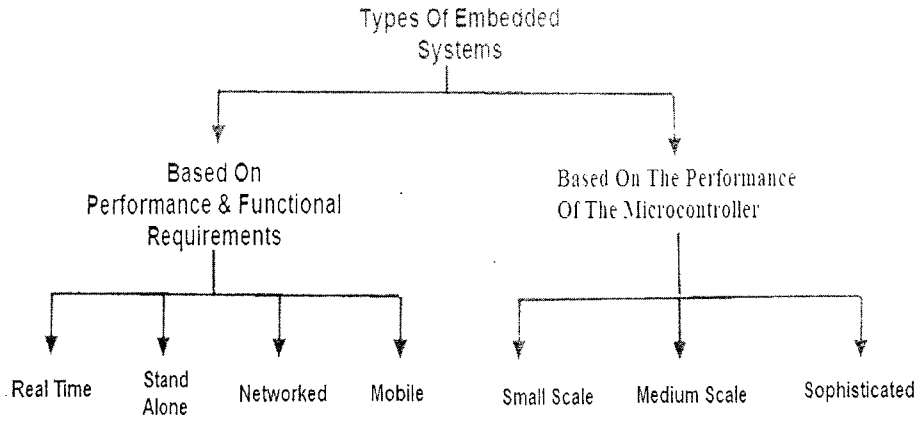
Small-scale embedded systems consist of **8-16 bit** microcontroller. This system can perform tasks at a small level. They have on-chip ROM and RAM. Small-scale systems can be even activated by the battery. The tools used to develop small-scale embedded systems are an editor, cross assembler, assembler and integrated development environment (IDE). The purpose of this system is not computation but to control as a computer embedded inside it. It behaves as a component of a computer and its function is not to compute. The small-scale system is dedicated to some specific task. To apply for the job as a small-scale embedded designer you need **skills** including data communication, digital electronic design, control engineering, software engineering, computer architecture, motors & actuators, analog electronic design, sensors & measurement and IC design & measurement

Medium scale embedded system:

This embedded system has **16-32 bit** microprocessor or microcontroller with external RAM and ROM. They can perform medium to complex level works. The integration between hardware and software is complex in these embedded systems. Programming languages used to develop medium scale embedded systems include Java, C, Visual C++, debugger, C++, RTOS, simulator, source code engineering tool and IDE. The designer of the medium scale embedded system should also know how to use semaphores, queues, mailboxes, pipes, and sockets. Knowing the application programming interface (API) in the RTOS tool for controlling microcontroller is also necessary.

Sophisticated embedded system:

The embedded system which can do large-scale works with multiple **32-64 bit** chips is known as sophisticated embedded systems. They can perform distributed work on a large scale. The complexity of hardware and software is very high in these systems. In sophisticated embedded systems, hardware and software are assembled together on large scale and designing of hardware products is also included in these systems.



5 5

Harvard Architecture	Von Neumann Architecture
Harvard architecture has physically separate pathways for instructions and data.	Von Neumann architecture uses same physical pathways for instructions and data .
It has one dedicated set of address and data bus for reading data from and writing data to memory, and another set of address and data buses for fetching instructions.	It has same set of data and address buses for memory read/write and fetching instructions.
Under harvard architecture the CPU can both read an	Under pure von Neumann architecture the CPU can be

instruction and perform a data memory access at the same time . Harvard architecture machine has distinct code and data address spaces: instruction address zero is not the same as data address zero.	either reading an instruction or reading/writing data from/to the memory. Both cannot occur at the same time since the instructions and data use the same bus system. Von Neumann architecture has same data address and instruction address .		
a) Type and amount of Hardware-SOC b) optimizing the power Dissipation and energy consumption-Phone C) Process deadlines-Video conferencing System d) flexibility-USB e) ability to upgrade –mobile with camera f) Reliability- Validation and verification		5	5
Flash memory: • Remembers data after power is turned off. • Slow read/write speeds compared to RAM/ROM • Can be manufactured in higher storage capacity than RAM/ROM • Used for SSDs, USB drives **RAM (Random Access Memory)** • Loses data when powered off. • Fast read/write speeds • Relatively expensive to manufacture • Used for storing program memory while computer is running **ROM (Read Only Memory)** • Does not lose data when powered off • Hard to write to • Speeds vary from very fast to flash memory speed • Usually used for storing the software to boot a computer/device nowadays. lash memory is a type of electronically-erasable programmable read-only memory (EEPROM), but it can also be a standalone memory storage device such as USB drive. It is a non-volatile memory chip used for storage and for transferring data between a PC and other digital devices. It is often found in USB flash drives, MP3 players, digital cameras and solid-state drives. Toshiba developed flash memory in the early 1980s and introduced it to the market in 1984.		5	5

	Flash memory incorporates the use of floating-gate [illegible] to store data EEPROM is a type of data memory device that uses an electronic device to erase or write digital data. It has per byte erase-and-write capabilities, which makes it slow. Flash memory is a distinct type of EEPROM which is programmed and erased in large blocks. Nonetheless, the trend seems to be of using AND flash for devices that only support large-block erasure. Flash memory has many features. It is a lot cheaper than EEPROM and does not require batteries for solid-state storage such as static RAM. To sum it up, • Flash is just one type of EEPROM • Flash uses NAND-type memory, while EEPROM uses NOR type • Flash is block-wise erasable, while EEPROM is byte-wise erasable • Flash is constantly rewritten, while other EEPROMs are seldom rewritten Flash is used when large amounts are needed, while EEPROM is used when only small amounts are needed.		

Part C (1Q x 10M = 10Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
	The ARM cortex is a complicated microcontroller within the ARM family that has ARMv7 design. There are 3 subfamilies within the ARM cortex family : • ARM Cortex Ax-series • ARM-Cortex Rx-series • ARM-Cortex Mx-series The ARM Architecture • Arithmetic Logic Unit • Booth multiplier • Barrel shifter • Control unit • Register file It is a reduced instruction set computing Controller • 32-bit high performance central processing unit • 3-stage pipeline and compact one It has THUMB-2 technology • Merges optimally with 16/32 bit instructions • High performance	5+5=10	30

It supports tools and RTOS and its core Sight debug and trace

- JTAG or 2-pin serial wire debugs connection
- Support for multiple processors

Low power Modes

- It supports sleep modes
- Control the software package
- Multiple power domains

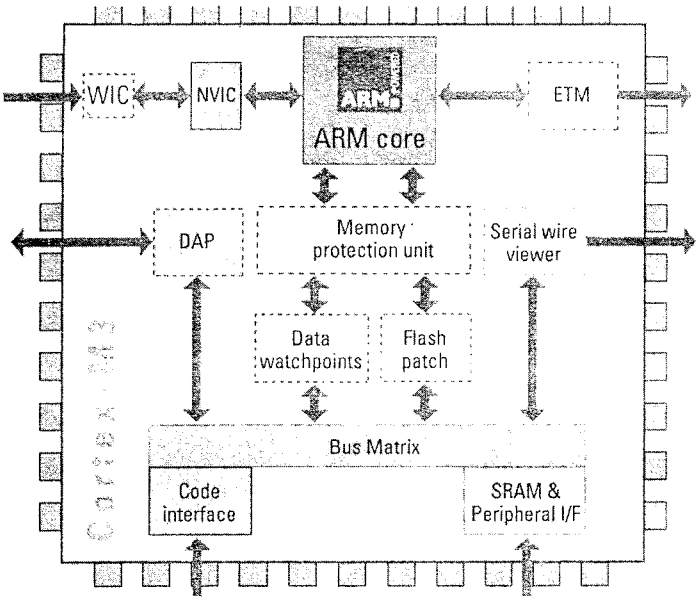
Nested vectored interrupt controller (NVIC)

- Low latency, low noise interrupts response
- No need for assembly programming

1.

- **Embedded System** concern more about software programming like C,C++ and how to write efficient code with lowest memory usage and memory management. nearly all embedded systems have a limited storage utilities.

- **VLSI** concern more about Hardware Design Language **HDL** like vhdl and verilog which is used to describe the digital components in terms of lines of code. the art of coding using HDL is totally different than the software programming. we concern more about making the code represent some definite design. not just a behavioral syntax as similar to normal software programming



Roll No.													

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

PRESIDENCY UNIVERSITY
BENGALURU

SCHOOL OF ENGINEERING

TEST – 2

Sem & AY: Odd Sem 2019-20
Course Code: ECE 308
Course Name: EMBEDDED SYSTEM DESIGN USING ARM
Program & Sem: BTech. (ECE) & VII (DE)

Date: 18.11.2019
Time: 9.30 AM to 10.30 AM
Max Marks: 40
Weightage: 20%

Instructions:

(i) ***Read Questions carefully and answer accordingly***

Part A [Memory Recall Questions]

Answer all the Questions. Each Question carries two marks. (5Qx2M=10M)

1. List out the types of Instruction Sets in ARM processor (C.O.NO.2) [Knowledge]
2. What are the types of flags in ARM processor? (C.O.NO.2) [Knowledge]
3. Difference between ARM processor and ARM core (C.O.NO.2) [Knowledge]
4. Define Thumb instruction set (C.O.NO.2) [Knowledge]
5. Illustrate the advantages of AMBA Bus. (C.O.NO.2) [Knowledge]

Part B [Thought Provoking Questions]

Answer all the Questions. Each Question carries five marks. (4Qx5M=20M)

6. Explain Different types of Addressing mode (C.O.2&3) [Comprehension]
7. Discuss in Detail about Data types and Keywords in Embedded C. (C.O.2&3) [Comprehension]
8. Difference between ARM Cortex-M3 and Arm 7 DMI processors. (C.O.2&3) [Comprehension]
9. Write an ALP program to add the contents of two 64-bit Variables. (C.O.2&3) [Comprehension]

Part C [Problem Solving Questions]

Answer the Question. The Question carry ten marks. (1Qx10M=10M)

10. Describe in detail about Block diagram of LPC 2378. Compare and contrast of ARM Microcontroller with respect to other types of Microcontrollers. (C.O.NO.2) [Comprehension]

SCHOOL OF-ENGINEERING

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

Semester: 2019-2020

Course Code: ECE 308

Course Name: Embedded System Design using ARM

Date: 18.11.2019

Time: 1 hour

Max Marks:40

Weightage: 20%

Extract of question distribution [outcome wise & level wise]

Q.NO	C.O.NO	Unit/Module Number/Unit /Module Title	Memory recall type [Marks allotted] Bloom's Levels	Thought provoking type [Marks allotted] Bloom's Levels	Problem Solving type [Marks allotted]	Total Marks
			K	C	A	
1-5	CO 2	Module 2	10			10
6	CO 2	Module 2		5		5
7	CO 3	Module 3		5		5
8	CO 2	Module 2		5		5
9	CO 3	Module 3		5		5
10	CO 2	Module 2		10		10
	Total Marks					40

K =Knowledge Level C = Comprehension Level, A = Application Level

Note: While setting all types of questions the general guideline is that about 60%

Of the questions must be such that even a below average students must be able to attempt, About 20% of the questions must be such that only above average students must be able to attempt and finally 20% of the questions must be such that only the bright students must be able to attempt.

[I hereby certify that All the questions are set as per above guide lines Mr.J.Brittopari]

Reviewers Comments

(1) Too many questions given.
(2) No Thought Provoking question.
(3) Part-C, Qn-10 is too lengthy for 10 marks. Either Block Diagram or description needed
(4) Marks & time should be balanced
(5) No Application level question given

12/11/19

Annexure- II: Format of Answer Scheme

GAIN MORE KNOWLEDGE REACH GREATER HEIGHTS

SCHOOL OF ENGINEERING

SOLUTION

Semester: 2019-2020

Course Code: ECE 308

Course Name: Embedded System Design Using ARM

Date: 18.11.19

Time: 1 hour

Max Marks:40

Weightage: 20%

Part A (5Q x 2M = 10 Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
1	Data Movement Memory Access Arithmetic Logical and Bit Manipulation Flow Control System Control	2	2
	N Negative (sign) flag, Z Zero flag C Carry flag V Overflow flag	2	2
3	ARM core is the part of the Arm processor. Inside ARM core- memory, ALU, Barrel	2	2

	shifter and instruction register and control unit.		
4	Thumb instruction set: Subset of 16 bit- Arm instruction set Reduced Memory usage	**2**	**2**
5	Higher bandwidth and data rate	**2**	**2**

Part B (4Q x 5M = 20Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
6	Types of Addressing Modes Name — Alternative Name — ARM Examples --- Register to register — Register direct — MOV R0, R1 --- Absolute — Direct — LDR R0, MEM --- Literal — Immediate — MOV R0, #15 ADD R1, R2, #12 --- Indexed, base — Register indirect — LDR R0, [R1] --- Pre-indexed, — Register indirect base with displacement with offset — LDR R0, [R1, #4] --- Pre-indexed, autoindexing — Register indirect pre-incrementing — LDR R0, [R1, #4]! --- Post-indexing, autoindexed — Register indirect post-increment — LDR R0, [R1], #4 --- Double Reg indirect — Register indirect — LDR R0, [R1, R2]	2+3=5	5

	Register indexed --- Double Reg indirect with scaling — Register indirect indexed with scaling — LDR R0, [R1, r2, LSL #2] --- Program counter relative — LDR R0, [PC, #offset] ---		

7	Data types and Keywords in Embedded C:
	
	Data Type | Size | Range
	Char or signed char | 1byte | -128 to +128
	Unsigned char | 1byte | 0 to 255
	Int or singed int | 2byte | -32768 to 32767
	Unsigned int | 2byte | 0 to 65535
	
	Embedded C Additional Keywords
	• Sbit
	sbit a=P0^1
	a=0x01;
	• Bit
	bit c;
	SFR
	SFR port=0x00;
	Port =0X01;
	• Volatile
	Volatile int k;
	• macros define
	#define dat Po;
	dat =0x01;
		5	5

8	Program code size significantly reduced Low cost, size and power Switching from ARM state to Thumb state is removed.	5	5
9		5	5

Program 7.7: add64.s 64 bit addition

```
; 64 bit addition

        TTL     Ch4Ex3 - add64
        AREA    Program, CODE, READONLY
        ENTRY

Main
        LDR     R0, =Value1             ; Pointer to first value
        LDR     R1, [R0]                ; Load first part of value1
        LDR     R2, [R0, #4]            ; Load lower part of value1
        LDR     R0, =Value2             ; Pointer to second value
        LDR     R3, [R0]                ; Load upper part of value2
        LDR     R4, [R0, #4]            ; Load lower part of value2
        ADDS    R6, R2, R4              ; Add lower 4 bytes and set carry flag
        ADC     R5, R1, R3              ; Add upper 4 bytes including carry
        LDR     R0, =Result             ; Pointer to Result
        STR     R5, [R0]                ; Store upper part of result

        STR     R6, [R0, #4]            ; Store lower part of result
        SWI     &11

Value1  DCD     &12A2E640, &F2100123    ; Value to be added
Value2  DCD     &00101080, &40023F51    ; Value to be added
Result  DCD     0                       ; Space to store result

        END
```

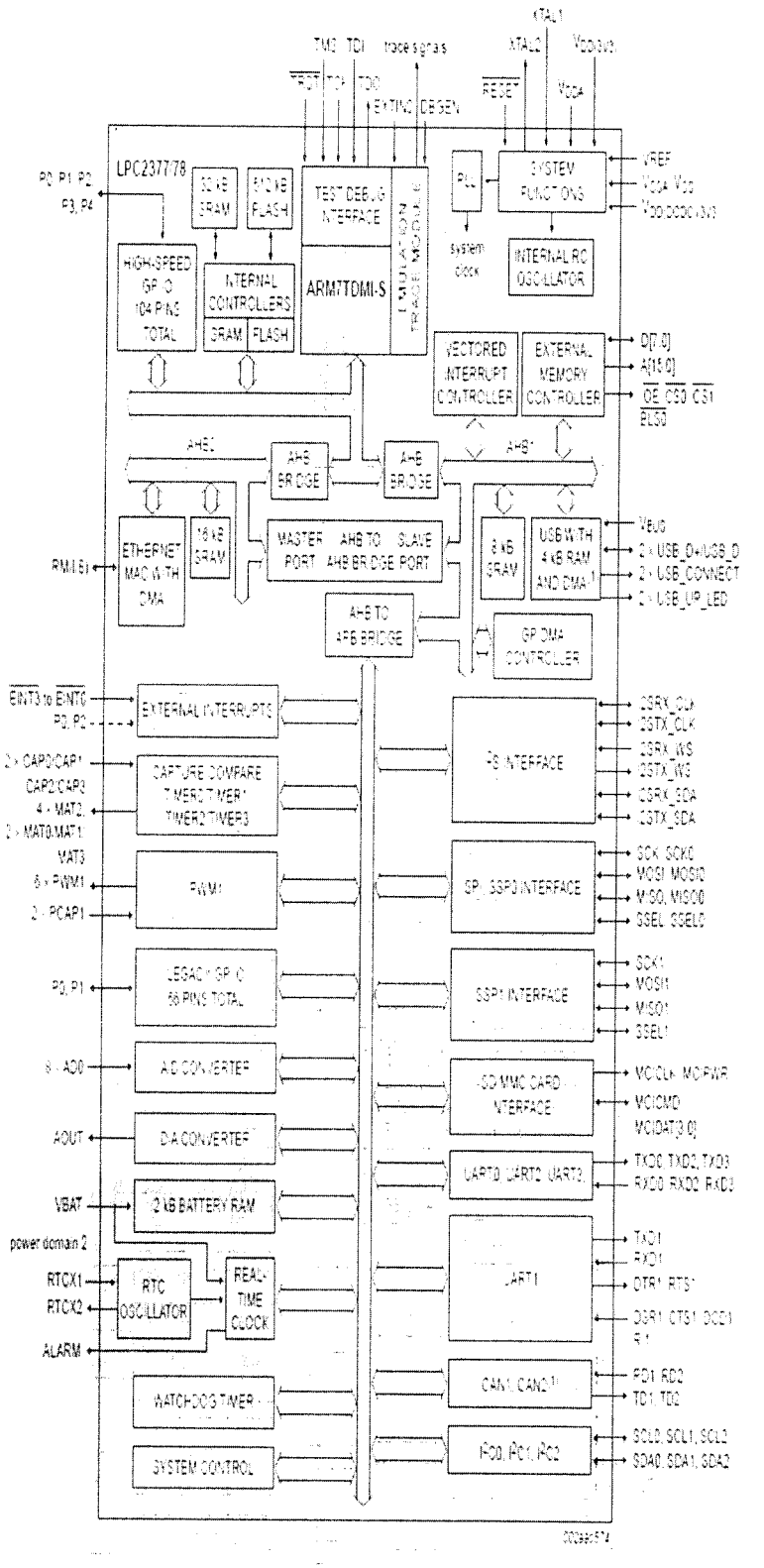
Part C (1Q x 10M = 10Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
10	ARM7TDMI-S processor, running at up to 72 MHz Up to 512 kB on-chip flash program memory with In-System Programming (ISP) and In-Application Programming (IAP) capabilities. Flash program memory is on the ARM local bus for high performance CPU access. 32 kB of SRAM on the ARM local bus for high performance CPU access. 16 kB SRAM for Ethernet interface. Can also be used as general purpose SRAM. 8 kB SRAM for general purpose DMA use; also accessible by the USB.	7+3=10	30

	Dual Advanced High-performance Bus (AHB) system that provides for simultaneous Ethernet DMA, USB DMA, and program execution from on-chip flash with no contention between those functions. A bus bridge allows the Ethernet DMA to access the other AHB subsystem. EMC provides support for static devices such as flash and SRAM as well as off-chip memory mapped peripherals. Advanced Vectored Interrupt Controller (VIC), supporting up to 32 vectored interrupts. General Purpose DMA controller (GPDMA) on AHB that can be used with the SSP serial interfaces, the I2S port, and the Secure Digital/Multi Media Card (SD/MMC) card port, as well as for memory-to-memory transfers. Serial Interfaces: Ethernet MAC with associated DMA controller. These functions reside on an independent AHB. USB 2.0 full-speed device with on-chip PHY and associated DMA controller (LPC2378 only). Four UARTs with fractional baud rate generation, one with modem control I/O, and one with IRDA support, all with FIFO. CAN controller with two channels (LPC2378 only). SPI controller. Two SSP controllers with FIFO and multi-protocol capabilities. One is an alternate for the SPI port, sharing its interrupt and pins. These controllers can be used with the GPDMA controller. Three I2C-bus interfaces (one with open-drain and two with standard port pins). I2S (Inter-IC Sound) interface for digital audio input or output. It can be used with the GPDMA.		

	Other peripherals: SD/MMC memory card interface. 104 General purpose I/O pins with configurable pull-up/down resistors. 10-bit ADC with input multiplexing among 8 pins. 10-bit DAC. Four general purpose timers/counters with 8 capture inputs and 10 compare outputs. Each timer block has an external count input. One PWM/timer block with support for three-phase motor control. The PWM has two external count inputs. Real-Time Clock (RTC) with separate power pin, clock source can be the RTC oscillator or the APB clock. 2 kB SRAM powered from the RTC power pin, allowing data to be stored when the rest of the chip is powered off. Watch Dog Timer (WDT). The WDT can be clocked from the internal RC oscillator, the RTC oscillator, or the APB clock. Standard ARM test/debug interface for compatibility with existing tools. Emulation trace module supports real-time trace. Single 3.3 V power supply (3.0 V to 3.6 V). Four reduced power modes: idle, sleep, power-down, and deep power-down. Four external interrupt inputs configurable as edge/level sensitive. All pins on Port 0 and Port 2 can be used as edge sensitive interrupt sources. Processor wake-up from Power-down mode via any interrupt able to operate during Power-down mode (includes external interrupts, RTC interrupt, USB activity, Ethernet Wake-up interrupt). Two independent power domains allow fine-tuning of power consumption based on needed features.		

	Each peripheral has its own clock divider for further power saving. Brownout detect with separate thresholds for interrupt and forced reset. On-chip power-on reset		



(1) LPC2378 only

Fig 1. LPC2377/78 block diagram

	Features of ARM controller: 1. 32bit data processor 2. RISC architecture 3. Instructions –one Clock cycle 4. Thumb instruction set 5. Load and store instruction 6. Low cost and size		

Roll No												

PRESIDENCY UNIVERSITY
BENGALURU

SCHOOL OF ENGINEERING

END TERM FINAL EXAMINATION

Semester: Odd Semester: 2019 - 20 | **Date**: 24 December 2019

Course Code: ECE 308 | **Time**: 9.30 AM to 12.30 PM

Course Name: EMBEDDED SYSTEM DESIGN USING ARM | **Max Marks**: 80

Program & Sem: B.Tech (ECE) & VII (DE - II) | **Weightage**: 40%

Instructions:

(i) Read Questions carefully and answer accordingly

(ii) This question paper contains two pages

Part A [Memory Recall Questions]

Answer all the Questions. Each Question carries 2 marks. (10Qx2M=20M)

1.	List the Types of processors in Embedded System	(C.O.No.1) [Knowledge]
2.	Give the significance of Embedded C programming	(C.O.No.1) [Knowledge]
3.	What is CPSR?	(C.O.No.2) [Knowledge]
4.	Difference between ARM processor and ARM core	(C.O.No.2)[Knowledge]
5.	Define Thumb instruction set	(C.O.No.3)[Knowledge]
6.	Illustrate the advantages of AMBA Bus	(C.O.No.3)[Knowledge]
7.	What is meant by UART?	(C.O.No.4)[Knowledge]
8.	Define full duplex communication	(C.O.No.4)[Knowledge]
9.	State the special features on I2C?	(C.O.No.4)[Knowledge]
10.	List out some applications of timer devices	(C.O.No.4)[Knowledge]

Part B [Thought Provoking Questions]

Answer all the Questions. Each Question carries 6 marks. (5Qx6M=30M)

11. Discuss in detail about classification of embedded system (C.O.No.1) [Comprehension]

12. Difference between ARM Cortex-M3 and Arm 7 DMI processors. (C.O. No.2) [Comprehension]

13. Write an ALP program to find the larger of two 32-bit variables. (C.O.No.3) [Comprehension]

14. Write an Embedded C program of Stepper motor interfacing using LPC2148 Arm Microcontroller

(C.O.No.4) [Comprehension]

15. Discuss in detail about Software and Hardware Design issues of Embedded System

(C.O.No.5) [Comprehension]

Part C [Problem Solving Questions]

Answer all the Questions. Each Question carries 10 marks. (3Qx10M=30M)

16. Describe in Detail about ARM core architecture with the neat block diagram. Conclude the Significance of the ARM processor compared to other processors

(C.O.No.2) [Comprehension]

17. Explain LCD interfacing of ARM LPC2148 with its schematic diagram

(C.O.No.4) [Comprehension]

18. Explain the features of UART interface for serial communication. Give a comparison table for UART, SPI and I2C interfaces. (C.O.No.4) [Comprehension]

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

SCHOOL OF ENGINEERING

END TERM FINAL EXAMINATION

Extract of question distribution [outcome wise & level wise]

Q.NO.	C.O.NO (% age of CO)	Unit/Module Number/Unit /Module Title	Memory recall type [Marks allotted] Bloom's Levels	Thought provoking type [Marks allotted] Bloom's Levels	Problem Solving type [Marks allotted]	Total Marks
			K	C	A	
1	1	1	2			2
2	1	1	2			2
3	2	2	2			2
4	2	2	2			2
5	3	3	2			2
6	3	3	2			2
7	4	4	2			2
8	4	4	2			2
9	4	4	2			2
10	4	4	2			2
11	1	1		6		6
12	2	2		6		6
13	3	3		6		6
14	4	4		6		6
15	5	4		6		6
16	2	2		10		10
17	4	4		10		10

18	4	4		10		10
	Total Marks		20	60		80

K =Knowledge Level C = Comprehension Level, A = Application Level

Note: While setting all types of questions the general guideline is that about 60%

Of the questions must be such that even a below average students must be able to attempt, About 20% of the questions must be such that only above average students must be able to attempt and finally 20% of the questions must be such that only the bright students must be able to attempt.

I hereby certify that all the questions are set as per the above guidelines.

Faculty Signature: J.B.Pj [Dr. J. Beulah Para]

Reviewer Commend: Faculty was asked to revise and briefed about thought provoking questions, which should be done as per guidelines.

Format of Answer Scheme Q17 is application level.

17/12/19

SCHOOL OF ENGINEERING

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

SOLUTION

Semester: Odd Sem. 2019-20
Course Code: ECE 308
Course Name: Embedded System Design using ARM
Program & Sem:B.Tech(ECE) & 7Th Sem

Date: 24.12.2019
Time: 9.30AM to 12.30PM
Max Marks: 80
Weightage: 40%

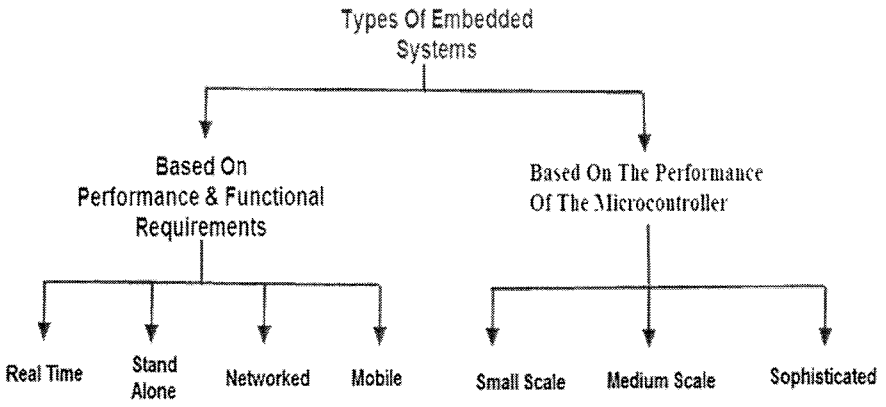
Part A (10Q x 2M = 20Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
1	a)Microprocessor or Embedded processor b) Application Specific Instruction set processor C) Single purpose processor d)SOC	2	2
2	Time consuming is less. Memory usage is reduced	2	2
3	The current processor status register (CPSR) contains the current status of the processor. This includes various condition code flags, interrupt status, processor mode and other status and control information	2	2

4	ARM core is the part of the Arm processor. Inside ARM core- memory, ALU, Barrel shifter and instruction register and control unit.	**2**	**2**
5	Thumb instruction set: Subset of 16 bit- Arm instruction set Reduced Memory usage	**2**	**2**
6	Higher bandwidth and data rate	**2**	**2**
7	UART stands for universal Asynchronous Receiver/Transmitter. UART is a hardware component for translating the data between parallel and serial interfaces. UART does convert bytes of data to and from asynchronous start stop bit. UART is normally used in MODEM	**2**	**2**
8	Transmission occurs in both the direction, simultaneously	**2**	**2**
9	• Low cost • Easy implementation • Moderate speed (upto 100 kbps).	**2**	**2**
10	• Real Time clock • Watchdog timer • Input pulse counting • TDM • Scheduling of various tasks	**2**	**2**

Part B (5Q x 6M = 30 Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
11	**Real-time embedded system:** Some examples of real-time embedded systems include:- •Controlling heat, elevators, lights, and doors in buildings •Robots •Traffic control system including railway tracks, airspace, shipping lines, highways **Standalone embedded system:** Examples of standalone embedded systems include:- •Microwave ovens •Digital cameras •Mp3 players •Video game consoles •Temperature measurement systems **Networked embedded system:** The network can be either a local area network (LAN) or a wide area network (WAN). **Mobile embedded system:** Mobile embedded systems are limited in resources including memory. Examples of mobile embedded systems include:- • Personal digital assistants (PDA) • Cellular phones • Mp3 players • Digital cameras, **Small-scale embedded system:** Small-scale embedded systems consist of 8-16 bit microcontroller. **Medium scale embedded system:**	6	20

	This embedded system has 16-32 bit microprocessor or microcontroller with external RAM and ROM They can perform medium to complex level works. **Sophisticated embedded system:** The embedded system which can do large-scale works with multiple 32-64 bit chips is known as sophisticated embedded systems.(CAN controller)  		
12	Program code size significantly reduced Low cost, size and power Switching from ARM state to Thumb state is removed	6	20

13	Program 7.6: bigger.s *Find the larger of two numbers*
	`; Find the larger of two numbers`
	`TTL Ch4Ex7 - bigger`
	`AREA Program, CODE, READONLY`
	`ENTRY`
	`Main`
	`LDR R1, Value1 ; Load the first value to be compared`
	`LDR R2, Value2 ; Load the second value to be compared`
	`CMP R1, R2 ; Compare them`
	`BGT Done ; If R1 contains the highest`
	`MOV R1, R2 ; otherwise overwrite R1`
	`Done`
	`STR R1, Result ; Store the result`
	`SWI &11`
	`Value1 DCD &12345678 ; Value to be compared`
	`Value2 DCD &87654321 ; Value to be compared`
	`Result DCD 0 ; Space to store result`
	`END`
		6	

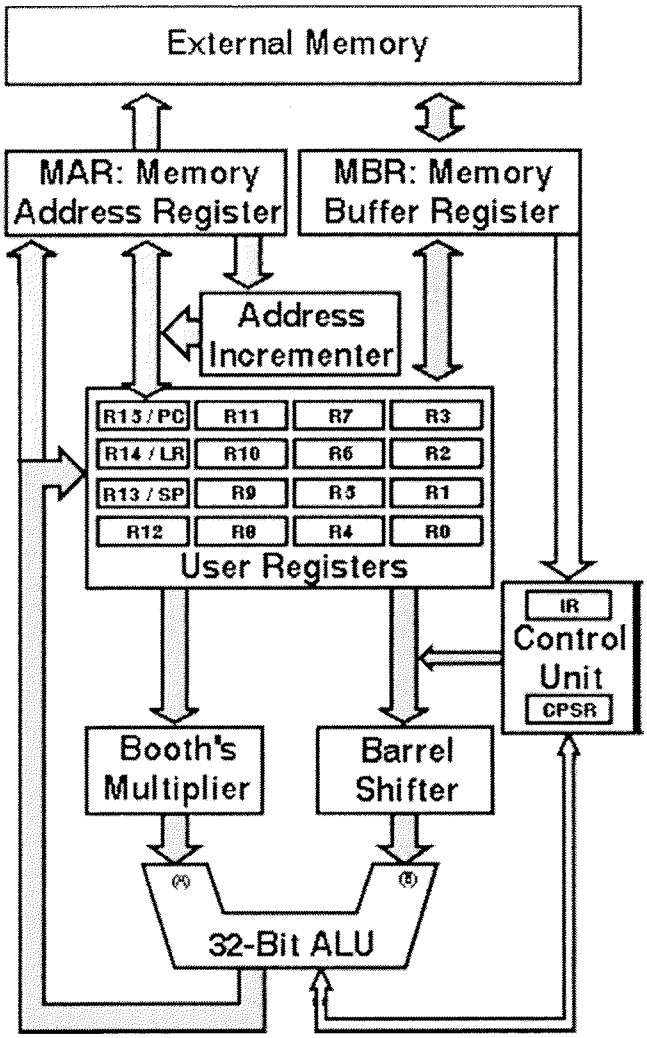
14	#include<lpc214x.h>
	
	void delay(unsigned int value);
	
	unsigned char clockwise[4] = {0x1,0x2,0x4,0x8}; //Commands for clockwise rotation
	unsigned char anticlockwise[4] = {0x8,0x4,0x2,0x1}; //Commands for anticlockwise rotation
	
	int no_of_steps = 550; //Change this value for required number of steps rotation (550 gives one complete rotation)
	
	int main()
	{
	
	PINSEL0 = 0x00000000; //Setting PORT0 pins
		6	20

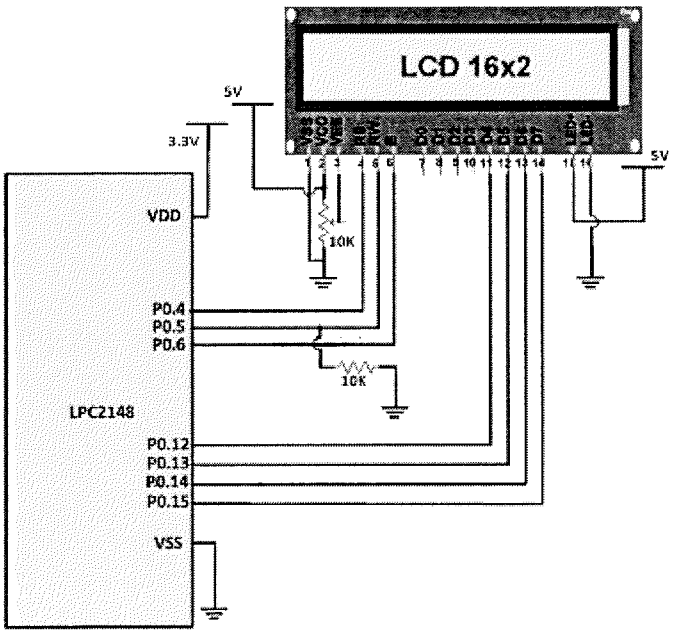
	IO0DIR |= 0x00000780; //Setting pins P0.7, P0.8, P0.9, P0.10 as OUTPUT
	IO0CLR = 0x00000780; //Setting P0.7, P0.8, P0.9, P0.10 pins OUTPUT as LOW
	
	while(1) // While loop for continueous operation
	{
	for (int j=0; j<no_of_steps;j++)
	{
	for(int i=0; i<4;i++)
	{
	IOPIN0 =clockwise[i]<<7; // Settting the pin value HIGH one by one after shifting bit to left
	delay(0x10000); //Change this value to change the speed of rotation
	}
	}
	for (int z=0;z<no_of_steps;z++)
	{
	for(int i=0; i<4;i++)
	{
	IOPIN0 =anticlockwise[i]<<7;
	delay(0x10000); //Change this value to change the speed of rotation
	}
	}
	
	}
	}
	
	void delay(unsigned int value) //Function to generate delay
	{
	unsigned int z;
	for(z=0;z<value;z++);
	}		

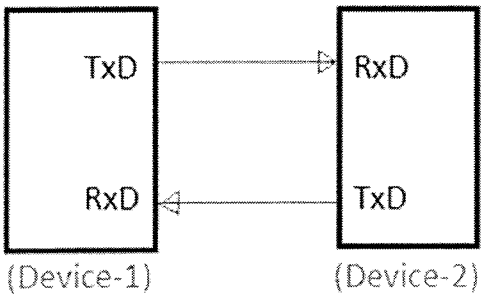
15	**Hardware issues:** a) Type and amount of Hardware-SOC b) optimizing the power Dissipation and energy consumption-Phone C) Process deadlines-Video conferencing System d) flexibility-USB e) ability to upgrate –mobile with camera f) Reliability- Validation and verification **Software issues:** a)connectivity-IOT b)change in phase c)debugging d)Era of Embedded Software	6	20

Part C (3Q x 10M = 30Marks)

Q No	Solution	Scheme of Marking	Max. Time required for each Question
16	The ARM is a Reduced Instruction Set Computer (RISC) system and includes the attributes typical to that type of system: _ A large array of uniform registers. _ A load/store model of data-processing where operations can only operate on registers and not directly on memory. This requires that all data be loaded into registers before an operation can be performed, the result can then be used for further processing or stored back into memory. _ A small number of addressing modes with all load/store addresses begin determined from registers and instruction fields only. _ A uniform fixed length instruction (32-bit). In addition to these traditional features of a RISC system the ARM provides a number of additional features: _ Separate Arithmetic Logic Unit (ALU) and shifter giving additional control over data processing to maximize execution speed. _ Auto-increment and Auto-decrement addressing modes to improve the operation of program loops. _ Conditional execution of instructions to reduce pipeline flushing and thus increase execution speed. **Significance of ARM processor** 1. 32bit data processor 2. RISC architecture 3. Instructions –one Clock cycle 4. Thumb instruction set 5. Load and store instruction 6. Low cost and size	7+3=10	30



17	Interfacing LCD with ARM7-LPC2148  The LCD can work in two different modes, namely the 4-bit mode and the 8-bit mode. In 4 bit mode we send the data nibble by nibble, first upper nibble and then lower nibble. For those of you who don't know what a nibble is: a nibble is a group of four bits, so the lower four bits (D0-D3) of a byte form the lower nibble	10	30

	while the upper four bits (D4-D7) of a byte form the higher nibble. This enables us to send 8 bit data Whereas in 8 bit mode we can send the 8-bit data directly in one stroke since we use all the 8 data lines. **NAME FUNCTION** VSS Ground Pin VDD +5V Input Pin VEE Contrast Adjust Pin RS Register Select R/W Read/Write Pin E Enable Pin D0-D7 Data Pins (8 Pins) LED A Anode Pin (+5V) LED K Cathode Pin (GND)		
18	**UART Interface** • The full name is Universal Asynchronous Receiver/Transmitter. • UART supports lower data rate. • Receiver need to know baud rate of the transmitter before initiation of reception i.e. before communication to be established. • UART is simple protocol, it uses start bit (before data word), stop bits (one or two, after data word), parity bit (even or odd) in its base format for data formatting. Parity bit helps in one bit error detection. ➡UART Packet = 1 start bit(low level), 8 data bits including parity bit, 1 or 2 stop bit(high level) • Data is transmitted byte by byte. • UART generates clock internally and synchronizes it with data stream with the help of transition of start bit. • It is also referred by RS232 . • For long distance communication, 5V UART is converted to higher voltages viz. +12V for logic 0 and -12V for logic 1. • The figure-1 depicts UART interface between two devices.  UART Interface Diagram	4+6=10	30

	UART	I2C	SPI
Complexity	Simple	Easy to chain many devices.	Complex as device increases
Speed	Slowest	Faster than UART	Fastest
Number of devices	Up to 2 devices	Up to 127 but may get complex as devices increases	Many, but there are practical limits and may get complicated
Number of wires	1	2	4
Duplex	Full Duplex	Half Duplex	Full Duplex
Number of masters and slaves	No multiple slaves and masters	Multiple slaves and masters	Only 1 master but can have multiple slaves.