14EI803 (A)

Hall Ticket Number:

IV/IV B.Tech (Regular/Supplementary) DEGREE EXAMINATION

March, 2019 **Electronics and Instrumentation Engineering**

Eighth Semester **PC BASED INSTRUMENTATION**

Time: Three Hours **Maximum :** 60 Marks

Answer Question No.1 compulsorily. (1X12 = 12 Marks)

Answer ONE question from each unit. (4X12=48 Marks)

1 Answer all questions (1X12=12 Marks)

a) Define VI & Sub VI.

b) What is meant by HMI and SCADA?

c) List the bus communication protocols

d) List the limitations of GPIB.

e) What is the use of Input Output port

f) When Add on cards are required

g) Discriminate local and global var iables

h) What is an embedded controller?

i) What is the purpose of VXI bus?

j) What is ISA bus standard?

k) What is meant by data acquisition?

l) What is VISA?

UNIT I

2 a) Explain the operation of a DMA controller with a neat diagram. 6M

b) What are the guide lines to be followed in selecting the data acquisition and control? 6M

(OR)

3 a) Explain the features of ISA and VME bus standards. 6M

b) Discuss the bus management approach among various micro computer buses. 6M

UNIT II

4 a) Explain data transfer scheme for add on cards. 6M

	b)	What is an add on card? Explain 8255 add on cards with a neat block diagram	6M

(OR)

5	a)	Explain briefly the need of different types of device drivers.	6M
	b)	What are the differences between static and loadable device drivers?	6M

UNIT III

6	a)	Explain the block diagram and architecture of virtual instrument.	6M
	b)	Explain different data flow techniques in VI.	6M

(OR)

7	a)	Explain the features of SCADA programming for automation.	6M
	b)	Discuss the software programming details of HMI/SCADA using OPC server.	6M

UNIT IV

8	a)	Explain the instrument drivers design techniques in VI.	6M
	b)	Write about arrays, clusters and graphs in VI programming.	6M

(OR)

9	a)	Discuss the (i) PC based control (ii) VXI based control	6M
	b)	Explain the GPIB bus interface clearly	6M

14EI803

Hall Ticket Number:

IV/IV B.Tech (Regular) Sixth Semester DEGREE EXAMINATION, March, 2019

Electronics & Instrumentation Engineering

PCBASED INSTRUMENTATION SCHEME AND SOLUTION

1.a.Define VI and sub VI.

A: Virtual instrumentation includes PC with flexible software and a wide variety of measurement and control hardware.

A SubVi is a stand VI that can be called by other VI. The SubVI is similar to an individual function. Using SunVI is an efficient programming skill in that it allows you to use the same code in different situations, and make your main VI program clear and compact.

1.B.what is meant by HMI and SCADA?

A:
An **HMI** is a software application that presents information to an operator or user about the state of a process, and to accept and implement the operators control instructions.
SCADA is an acronym that stands for Supervisory Control and Data Acquisition.
SCADA refers to a system that collects data from various sensors at a factory, plant or in other remote locations and then sends this data to a central computer which then manages and controls the data.
1.C.List the bus communication protocols?
A:
i.Hardware specifications

ii. Bus format specifications

iii. Data transfer control protocols

iv. The allocation control protocols

v.Device synchronization protocols.

1.d.List the limitations of GPIB?

A:

We can connect instruments to the controller in any arrangement with the following limitations:

Do not connect more than 15 devices on any GPIB system. This number can be extended with the use of a bus extension.

Do not exceed a total of 20 meters of total cable length or 2 meters per device, whichever is less.

Avoid stacking more than three connectors on the back panel of an instrument. This can cause unnecessary strain on the rear-panel connector.

1.e.What is the use of Input and Output port

A: Analog /digital signal can be measured using pc system from input port and Analog /digital signal can be generated using pc system and available at output port.

1.f.When Add on cards are required?

A: An add-on is either a hardware unit that can be added to a computer to increase its capabilities or a program utility that enhances a primary program . Examples of add-ons for a computer include card s for sound, graphics acceleration, modem capability, and memory.

1.g.Discriminate local and global variables?

A: Global variables are built in Labview objects.We can use these variables to access and pass data among several VIs that run simultaneously.   A local variable shares data with in a VI.

1.h.What is an embedded controller?

It consists of User Interface,Real-time processor,Digital I/O ,Analog I/O and peripherals.It can acquire data and analyse and control and present data.

1.i.What is the purpose of VXI bus?

A: The **VXI bus** architecture is an open standard platform for automated test based upon VMEbus.

VXI stands for **V**ME e**X**tensions for **I**nstrumentation, defining additional bus lines for timing and triggering as well as mechanical requirements and standard protocols for configuration, message-based communication, multi-chassis extension, and other features.

1.j.What is ISA bus standard?

A: An Industry Standard Architecture bus (ISA bus) is a computer bus that allows additional expansion cards to be connected to a computer's motherboard.

1.k.What is meant by data acquisition?

A: **Data acquisition** (DAQ) is the process of measuring an electrical or physical phenomenon such as voltage, current, temperature, pressure, or sound with a computer.

A DAQ system consists of sensors, DAQ measurement hardware, and a computer with programmable software.

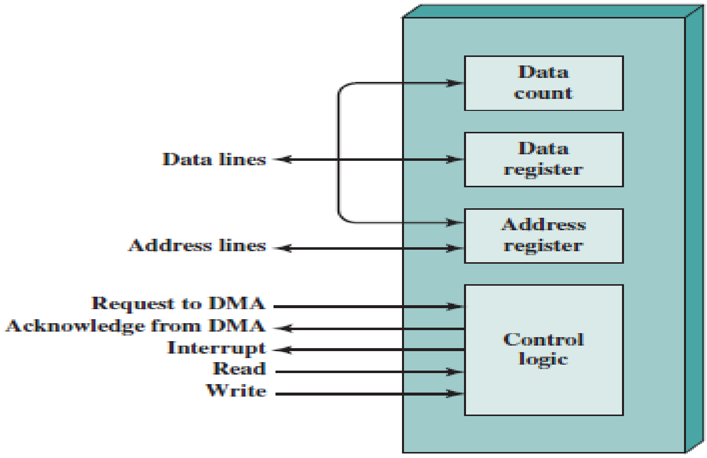
1.l.What is VISA?

A: The Virtual Instrument Software Architecture (VISA) is a standard for configuring, programming, and troubleshooting instrumentation systems comprising GPIB, VXI, PXI, serial (RS232/485), Ethernet, USB and/or IEEE 1394 interfaces.

UNIT-1

2.a . Explain the operation of DMA controller with a neat diagram. 6M

A:**Basic Operation of DMA**



Typical DMA Block Diagram

When the processor wishes read or send a block of data, it issues a command to the DMA module by sending some information to DMA module. The information includes:

1. read or write command, sending through read and write control lines.
2. number of words to be read or written, communicated on the data lines and stored in the data
3. starting location in memory to read from or write to, communicated on data lines and stored in address register.
4. address of the I/O device involved, communicated on the data lines

After the information are sent, the processor continues with other work. The DMA module then transfers the entire block of data directly to or from memory without going through the processor. When the transfer is complete, the DMA module sends an interrupt signal to the processor to inform that it has finish using the system bus.

2.b.What are the guide lines to be followed in selecting the data acquisition and control?

1. Types of signals needed to measure or generate.
2. Need of signal conditioning
3. Sample rate
4. Resolution
5. Accuracy
1. Types of signals needed to measure or generate.

Different types of signals need to be measured or generated in different ways. A sensor (or transducer) is a device that converts a physical phenomenon into a measurable electrical signal, such as voltage or current. You can also send a measurable electrical signal to your sensor to create a physical phenomenon. For this reason, it is important to understand the different types of signals and their corresponding attributes. Based on the signals in your application, you can start to consider which DAQ device to use.

2. Need of signal conditioning

A typical general-purpose DAQ device can measure or generate +/-5 V or +/-10 V. Some sensors generate signals too difficult or dangerous to measure directly with this type of DAQ device. Most sensors require signal conditioning, like amplification or filtering, before a DAQ device can effectively and accurately measure the signal.

3. Sample rate
One of the most important specifications of a DAQ device is the sampling rate, which is the speed at which the DAQ device's ADC takes samples of a signal. Typical sampling rates are either hardware- or software-timed and are up to rates of 2 MS/s. The sampling rate for your application depends on the maximum frequency component of the signal that you are trying to measure or generate.
4. Resolution
The smallest detectable change in the signal determines the resolution that is required of your DAQ device. Resolution refers to the number of binary levels an ADC can use to represent a signal. To illustrate this point, imagine how a sine wave would be represented if it were passed through an ADC with different resolutions. Figure 2 compares a 3-bit ADC and a 16-bit ADC. A 3-bit ADC can represent eight (2^3) discrete voltage levels. A 16-bit ADC can represent 65,536 (2^{16}) discrete voltage levels. The representation of the sine wave with a 3-bit resolution looks more like a step function than a sine wave where the 16-bit ADC provides a clean-looking sine wave.

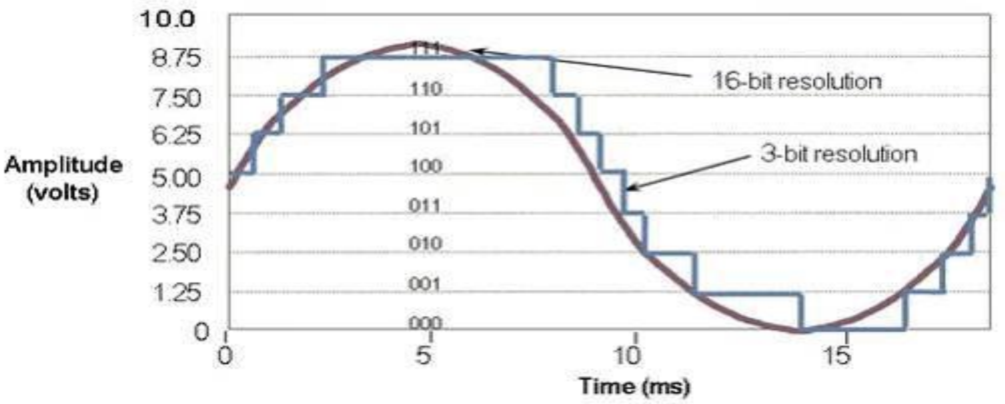


Figure 2. 16-Bit Resolution Versus 3-Bit Resolution Chart of a Sine Wave

5. Accuracy
Accuracy is defined as a measure of the capability of an instrument to *faithfully* indicate the value of a measured signal. This term is not related to resolution; however, accuracy can never be better than the resolution of the instrument.
An ideal instrument always measures the true value with 100 percent certainty, but in reality, instruments report a value with an uncertainty specified by the manufacturer.

3.a.Explain the features of ISA and VME bus standards.
6M

A:

ISA Features: It is 16-bit characteristic

It supported 16-bit peripheral devices.

Five devices with 16-bit interru pt request (IRQ) co ld be connected at the same time.

Three additional devices could be parallel to five devices with 16-bit IRQ.

16-bit direct memory access(DMA) channel

The CPU clock speed varied from 16 to 20MHz

VME features:

Larger, 64-bit data path for 6U boards.
Larger, 64-bit addressing range for 6U boards.
32-bit data and 40-bit addressing modes for 3U boards.
Twice the bandwidth (up to 80 Mbytes/sec).
Lower noise connector system.
Cycle retry capability.
Bus LOCK cycles.
First slot detector.

3.B.Discuss the bus management approach among various microcomputer buses. 6M

A: Consider a simple microcomputer system consisting of two devices that are linked by a bus. One Bus will initiate the data transfer another will supply or accept the data when requested to do so. The simplest parallel bus would consists of a single bi-directional line(Fig.1), along which the data are transferred in bit serial form and one transfer line to allow the master to tell the slave when data transfer is to occur other wise use two directional data buses i.e. shown in fig 2

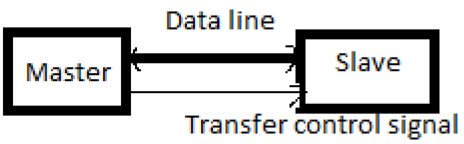


Fig.1

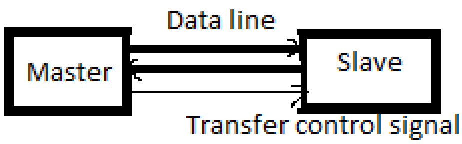


Fig.2

If the microcomputer has more than one slave device, additional information must be transmitted along the bus to allow the master to specify which of the slaves is to take part in the data transfer.

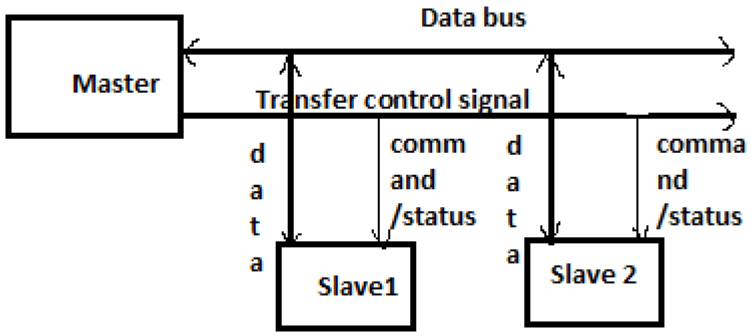


Fig.3

The simplest form for this command data is a single unique binary code. However, each bit of the command information would cause a specific action to be taken by the slave command information requires the slave to provide the master with status data indicating its current condition and ready for data transfer.

The other modification to the above method is shown below.

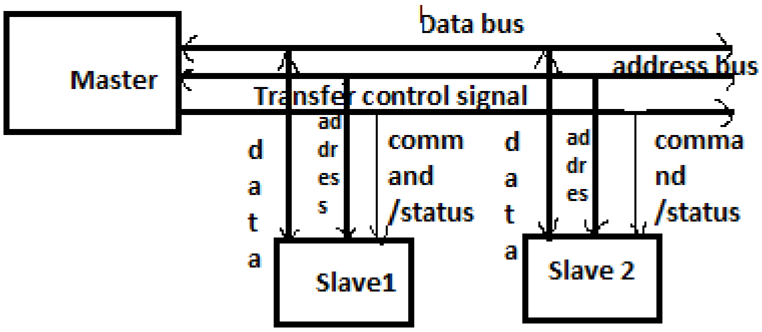


Fig. 4

The command information is sent out on a special bus (address bus).
The another approach is given below.

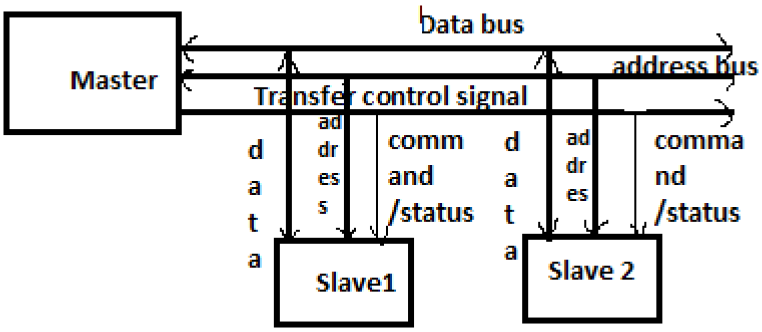


Fig. 4

In this approach, the command information is sent over a address bus and status

Information is sent over device synchronization control signal line. So, effectively we can transmit the data over the buses with some control lines.

Introduction of additional buses or the multiplexing of different types of information on the same bus will increase the no.of bus control lines and complexity of the logic of the bus interfaces in the master & slaves.

Bus Management:

One device, the current bus master, has control of the use the bus at any one time. The bus master initiates and organizations the transfer of data. In many microcomputer systems, the microprocessor will acts as the master and all other peripheral devices will act as slave devices.

The Below figure shows the bus management approach clearly.

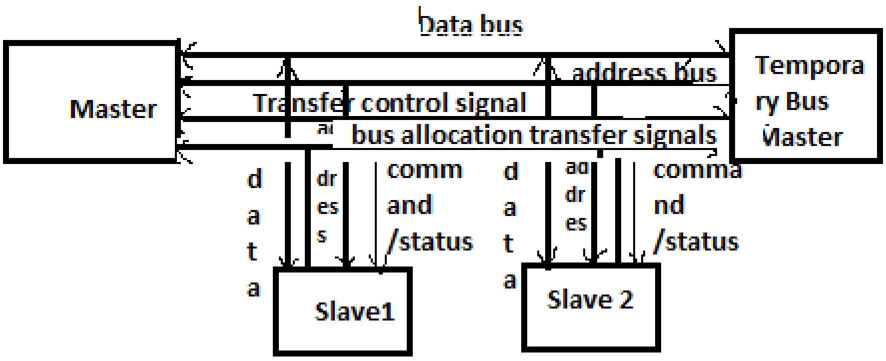


Fig. 6

Here, additional control signals as necessary to allocate the usage of bus for data transfer. These control lines are called as Bus allocation Control Signals.

Consider, one example to understand the bus management approach. It is now common for the microcomputer to use very fast data transfers between its main memory and magnetic storage devices. Here, Permanent bus master is the ‘microprocessor’ and temporary bus master will be the DMA comtroller.

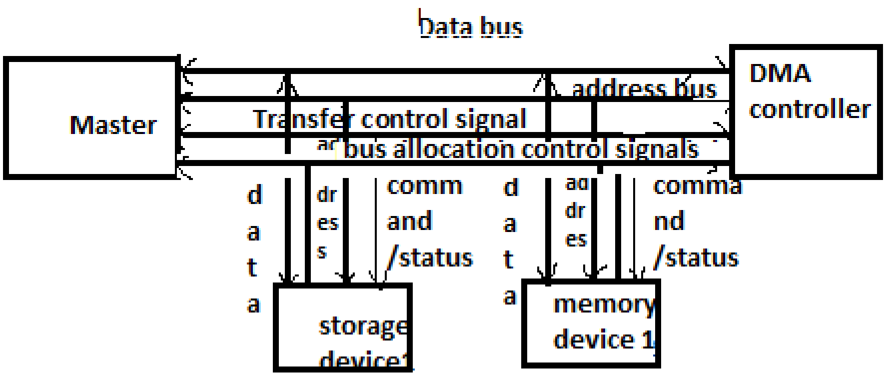


Fig. 7

The bus allocation control lines are useful for switching the control over a bus(data) between the permanent and temporary bus masters.

UNIT II

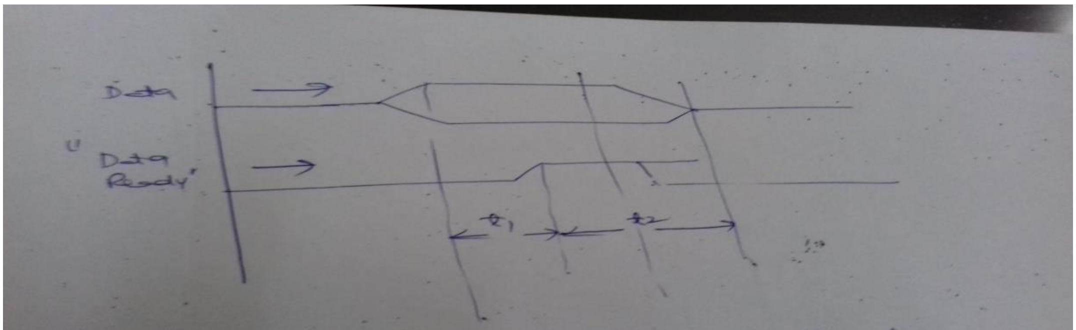
4.a. Explain data transfer scheme for add on cards. 6M

Data transfer control signals: Three basic functions of the data transfer control signals are

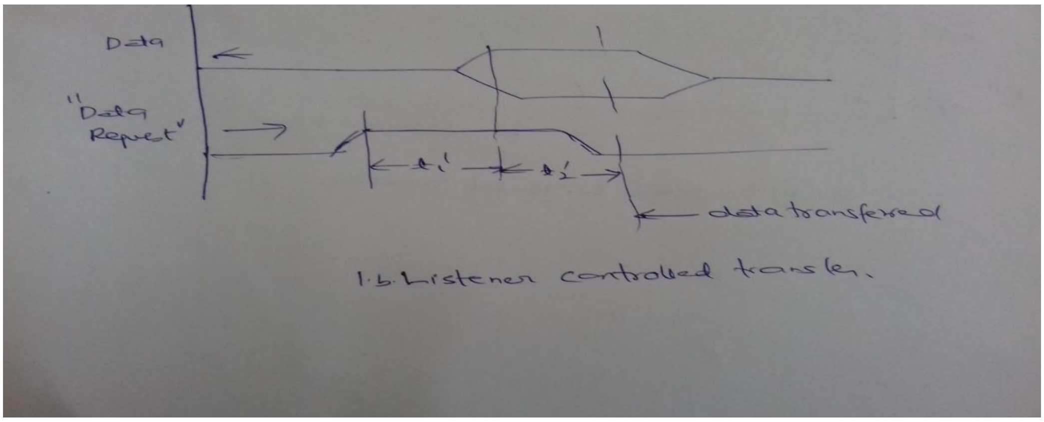
1. **To specify the source or destination of the data:**
   - The most important piece of information that must be sent out before the actual data transfer takes place is the slave device address
   - Most buses adopt the combined cycle method in which the slave address is transmitted over the data bus or over a separate address bus
   - In case of one way transfer control within the same bus transfer of address must complete before the data are transferred.
   - The time delay between the transfer of the address and the transfer of the data will be specified by the address and the transfer of the data will be specified by the bus protocol that define the decode and setup times required of devices interfaced to the bus.

SPLIT CYCLE METHOD:-

- Some buses adopt the split cycle method in which address and data sent separately.
- This scheme is widely used in multi master systems.



1.a. Talker Controlled Transfer.



(2)**To define the type of the data are being transferred:-**

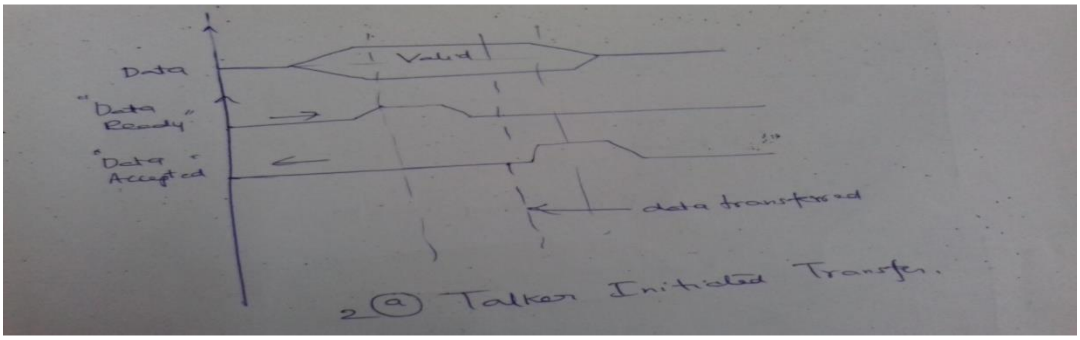
Before data transfer can take place , the bus control signal must indicate:

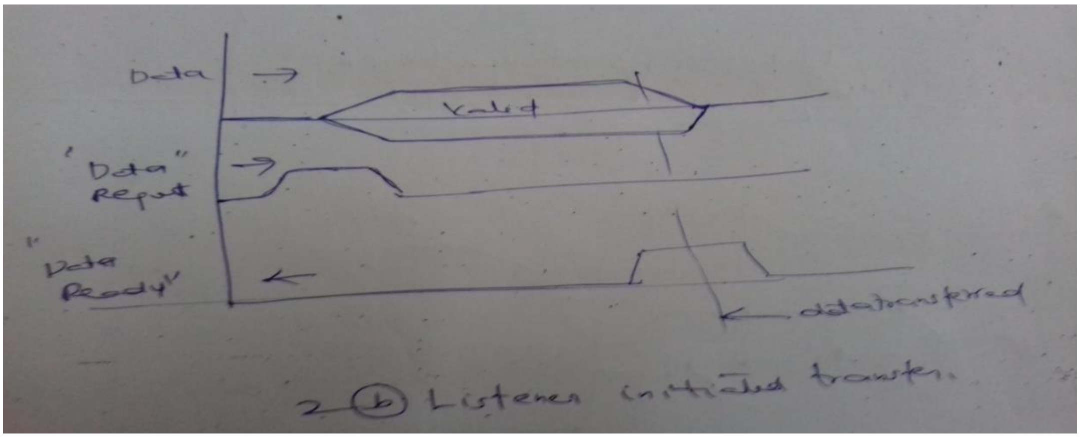
1.Direction of transfer w.r.t bus master.

2.Type of device that is involved in the transfer

3.The width of the data being transferred(1,2,3 or 4bytes)

4.Which bus lines are being used for the transfer i.e address bus ,lower byte of the data bus etc





(3)**To indicate when data are being transferred:-**

Since it is unusual for the various devices connected by a bus to operate synchronously.

1.So control signals must be transmitted along with the data,so that , both master and slave can co-ordinate their operations during the transfer.

There are three widely used approaches to contol of asynchronuous data transfers:

1.one way(synchronous conrol)

2.Request/acknowledge contol(hand shake control)

3.Semi synchronous control

4.Multi functional data transfer control

1.ONE –WAY (SYNCHRONOUS CONTROL):-

1.A single Data ready is sent out by the master after it has placed data on the data bus(if it is a talker) (or)

2.Reading the data bus (if it is listner), the signal is called Data request.

3.The slave must respond to the active transition of the control signal with in a prescribed time interval otherwise data will not be transferred properly.

4. Time intervals t1& t2 are highly dependent on implementation details like device operating speed , bus propagation delay etc

Advantages with this approach:-

1. simplicity.
2. speed of communication.(i.e it has single bus propagation delay).

Disadvantages:-

1.This scheme is sensitive to transmission speed.

communicating devices that have very different operating speed is difficult & will lead to inefficient use of the bus bandwidth.

(2)Request/acknowledge (handshake) control:-

Here two control are used.

1.Data Request:When it has successfully read the data if it is a listner.

2.Data Ready:when it has placed the data on the data bus if it is a talker.

Advantages:

1.Reduces the noise sensitivity.

2.Any speed is possible.

Disadvantages:

1.Each data transfer involves two bus propagation delays.

4.b. What is an add on card? Explain 8255 add on cards with a neat block diagram 6M

A: An add-on is either a hardware unit that can be added to a computer to increase its capabilities or a program utility that enhances a primary program . Examples of add-ons for a computer include card s for sound, graphics acceleration, modem capability, and memory. Software add-ons are common for games, word processors, and accounting programs.

PS-ADD-ON **8255 interface card** is designed to study the features of Intel **8255**, PS- **8255 Interface card** consist of 12 nos. of point LED's for displaying device logic output, 8nos of slide switches to give digital input and 4nos of tact switches for interrupt generating interrupt. PS-ADD-ON **8255** is user friendly facilitating the beginners to learn the operation of programmable peripheral interface. Address lines, Chip select lines and the Power lines are terminated with 50-pin connector.

Features:

- Programmable I/O control functions
- Up to 48 I/O lines TTL - 5V
- Three independent 16 bits counter
- Sixteen LED indicate when I/O is operating
- Provides DII device driver for PnP features
- ISA 8255N compatibly
- English manual with configuration and Pin Assignment.

5.a.Explain briefly the need of different types of device drivers. 6M

A:In computing, a **device driver** (commonly referred to simply as a *driver*) is a computer program that operates or controls a particular type of device that is attached to a computer. A driver provides a software interface to hardware devices, enabling operating systems and other computer programs to access hardware functions without needing to know precise details of the hardware being used.

A driver communicates with the device through the computer bus or communications subsystem to which the hardware connects. When a calling program invokes a routine in the driver, the driver issues commands to the device. Once the device sends data back to the driver, the driver may invoke routines in the original calling program. Drivers are hardware dependent and operating-system-specific. They usually provide the interrupt handling required for any necessary asynchronous time-dependent hardware interface.

List of Device Drivers: 1.Block Device Driver

2.Character Device Driver

3.Network Device Driver

4.Pseudodevice Driver

5.b.What are the differences between static and loadable device drivers? 6M

A:

SLNO	STATIC DEVICE DRIVERS	LOADBLE DEVICE DRIVERS
1	The static device driver has a plurality of handlers or functions , used to control a device	The loadable device drivers are loaded dynamically.
2	The control status register (CSR) determines the existence of static device drivers.	The bus configuration code determines the existence of loadable device drivers before calling.
3	It is not possible to install static device drivers from other vendors	It is easy for administrators to install loadable device drivers from other vendors.
4	These device drivers do not allow device drivers and other modules to be configured into kernel while the system is running.	These improve system availability by allowing device drivers and other modules to be configured into kernel , while the system is running.
5	Conservation of system resources in these device drivers is not possible when not in use.	Conservation of system resources in loadable device drivers can be done by unloading them frequently when not in use.
6	Static device drivers do not allow administrators to load and unload modules according to demand.	Loadable device drivers allow administrators to load and unload modules according to demand.
7	Auto loading of reqd. modules encountered by the kernel is not possible.	Auto loading occurs when the kernel detects a particular loadable module reqd. to accomplish some task.
8	Efficient use of kernel memory is not possible.	This efficiently saves kernel memory. [adv of loadable device drivers is that they can be loaded only when necessary & then unloaded]

UNIT III

6.a.Explain the block diagram and architecture of virtual instrument. 6M

The three-step approach has been one of the pillars of virtual instrumentation model as shown in Figure 6.a

Fig 6.a

A virtual instrument is not limited or confined to a stand-alone PC. In fact, with recent developments in networking technologies and the Internet, it is more common for instruments to use the power of connectivity for the purpose of task sharing. Typical examples include supercomputers, distributed monitoring and control devices, as well as data or result visualization from multiple locations. Every virtual instrument is built upon flexible, powerful software by an innovative engineer or scientist applying domain expertise to customize the measurement and control application. The result is a user-defined instrument specific to the application needs. Virtual instrumentation software can be divided into several different layers like the application software, test and data management software, measurement and control services software as shown in Figure 6.b.

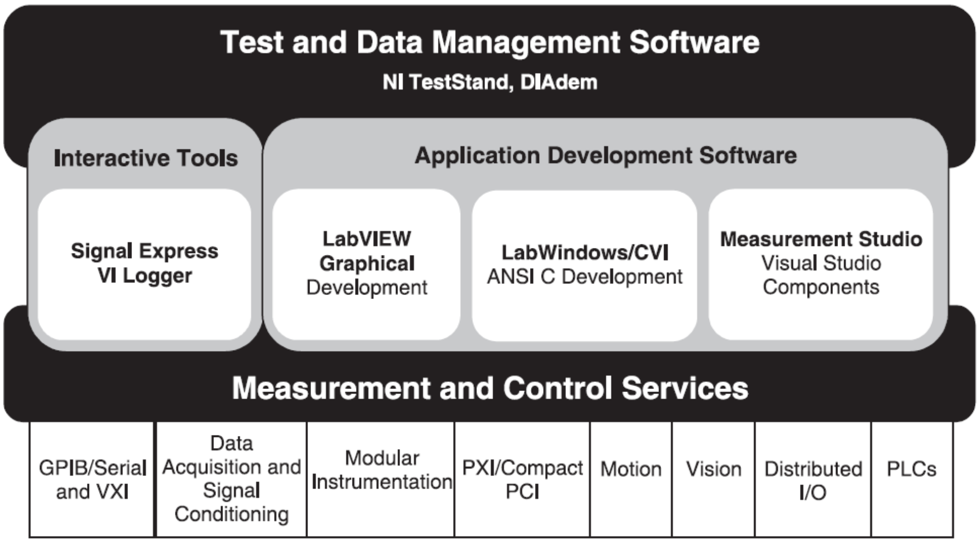


Figure 6.a.ii. Layers of virtual instrumentation software.

6.b.Explain different data flow techniques in VI.

DATA FLOW PROGRAM

LabVIEW follows a dataflow model for running VIs. A block diagram node executes when all its inputs are available. When a node completes execution, it supplies data to its output terminals and passes the output data to the next node in the dataflow path.

Visual Basic, C++, JAVA, and most other text-based programming languages follow a control flow model of program execution. In control flow, the sequential order of program elements determines the execution order of a program. For a data flow programming, consider a block diagram shown in Figure 6.b. that adds two numbers and then subtracts 50.00 from the result of the addition. In this case, the block diagram executes from left to right, not because the objects are placed in that order, but because the Subtract function cannot execute until the Add function finishes executing and passes the data to the Subtract function. A node executes only when data are available at all of its input terminals, and it supplies data to its output terminals only when it finishes execution.

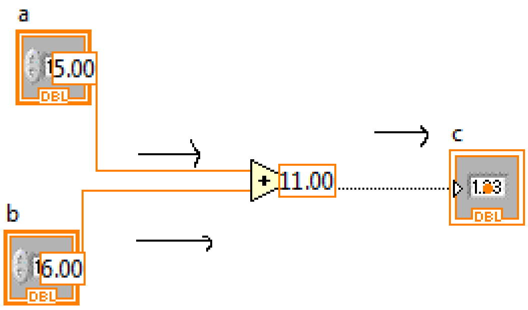


Figure 6.b Data flow program.

7.a.Explain the features of SCADA programming for automation. 6M

Features supported by SCADA:

1. System Management Features:

Device and System Control Features:

The SCADA system has defined algorithms and business logic to trigger a specific alert in case of any gateway or device failure.

There are various logics that can be customized to control the pattern of data flow. The SCADA on a large scale is able to do the parallel processing of a large data simultaneously.

This is an utmost important feature w.r.t industrial automation application where bulk processing is needed. Also the logics can be altered depending on the business need and the pattern of flow can be modifies if and when required.

Support for automated and customized reporting:

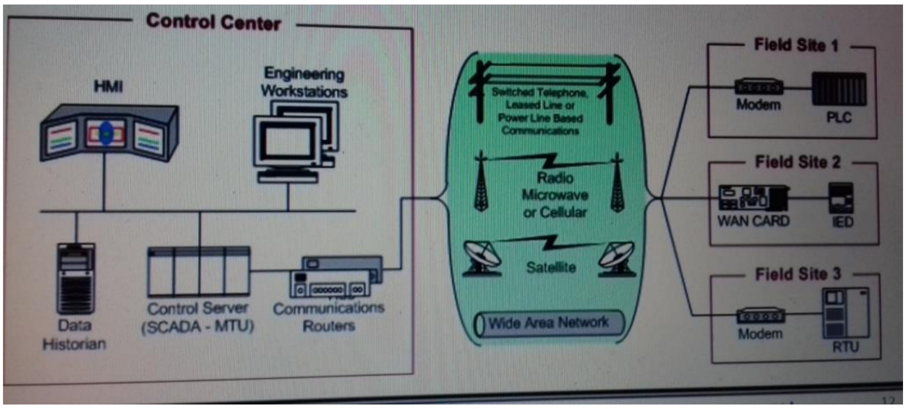
User defined reports can be synthesized using backend programming. Automated reports are mainly a health check of a particular functionality which can be generated in form of automated mailer alerts or specific admin alerts. This feature also helps in data acquisition and analysis in terms of user or industry admin operation and maintenance of a functional module.

7.b.Discuss the software programming details of HMI/SCADA using OPC server.

A: SCADA is an acronym that stands for Supervisory Control and Data Acquisition. SCADA refers to a system that collects data from various sensors at a factory, plant or in other remote locations and then sends this data to a central computer which then manages and controls the data.

SCADA systems are used not only in industrial processes: e.g. steel making, power generation (conventional and nuclear) and distribution, chemistry, but also in some experimental facilities such as nuclear fusion. The size of such plants range from a few 1000 to several 10 thousands input/output (I/O) channels.

A SCADA system usually includes signal hardware (input and output), controllers, networks, user interface (HMI), communications equipment and software. All together, the term SCADA refers to the entire central system. The central system usually monitors data from various sensors that are either in close proximity or off site (sometimes miles away).

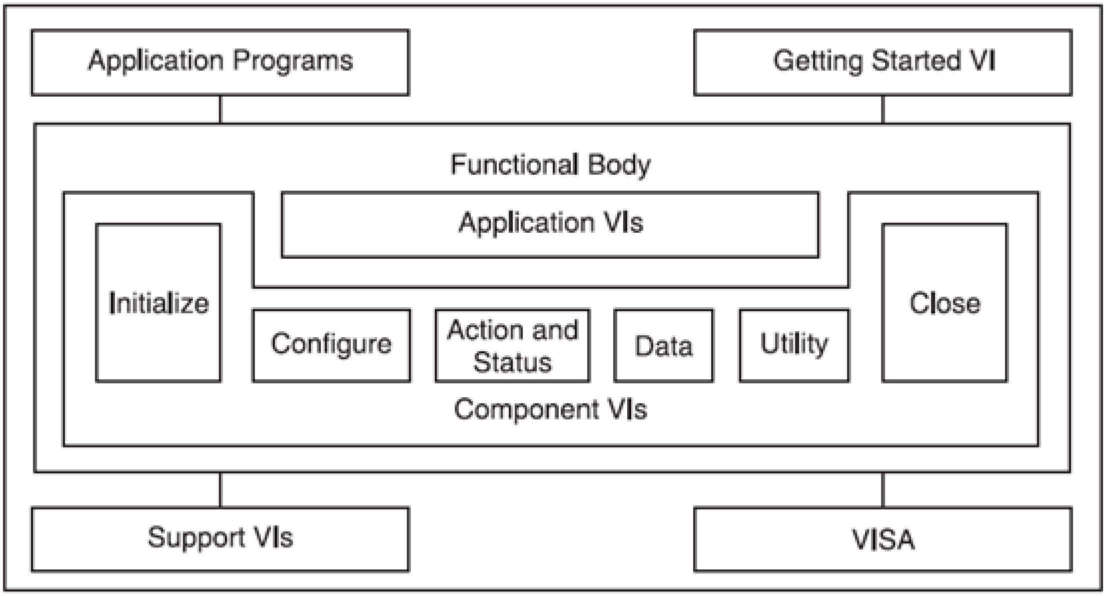


8.a. Explain the instrument drivers design techniques in VI. 6M

a: **INSTRUMENT I/O ASSISTANT**

1.The Instrument I/O Assistant is a LabVIEW Express VI which we can use to communicate with message-based instruments and convert the response from raw data to an ASCII representation.

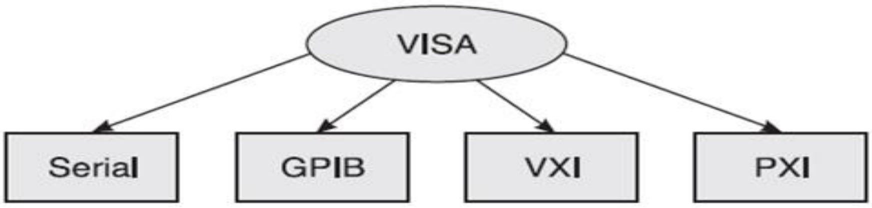
we can communicate with an instrument that uses a serial, Ethernet, or GPIB interface.



An instrument driver is a set of software routines that control a programmable instrument. Each routine corresponds to a programmatic operation such as configuring, reading from, writing to, and triggering the instrument. Instrument drivers simplify instrument control and reduce test program development time by eliminating the need to learn the programming protocol for each instrument.

National Instruments provides instrument drivers for a wide variety of instruments; these instrument drivers are written in LabVIEW and/or LabWindows/CVI and use either the Plug and Play architecture or the Interchangeable Virtual Instrument (IVI) architecture.

Both architectures use Virtual Instrumentation Software Architecture (VISA) to provide bus and platform independent instrument communication. The Instrument Driver Network has thousands of free instrument drivers for download.



.2VISA

Virtual Instrument Software Architecture (VISA) is the lower layer of functions in the LabVIEW instrument driver VIs that communicates with the driver software.

VISA by itself does not provide instrumentation programming capability. VISA is a high-level API that calls low-level drivers.

As shown in Figure 10.9 VISA can control VXI, GPIB, serial, or computer-based instruments and makes the appropriate driver calls depending on the type of instrument used.

When debugging VISA problems, remember that an apparent VISA problem could be an installation problem with one of the drivers that VISA calls.

8.b.Write about arrays, clusters and graphs in VI programming. 6M

A: LabVIEW includes the following types of graphs and charts:

● **Waveform graphs and charts**: Display data typically acquired at a constant rate.

● **XY Graphs:** Display data acquired at a non-constant rate and data for multivalued functions.

● **Intensity graphs and charts:** Display 3D data on a 2D plot by using color to display the values of the third dimension.

● **Digital waveform graphs:** Display data as pulses or groups of digital lines.

● **Windows 3D Graphs:** Display 3D data on a 3D plot in an ActiveX object on the front panel.

A. WAVEFORM GRAPHS -2M

LabVIEW includes the waveform graph and chart to display data typically acquired at a constant rate. The waveform graph displays one or more plots of evenly sampled measurements. The waveform graph plots only single-valued functions, as in y = f (x), with points evenly distributed along the x-axis, such as acquired time-varying waveforms. Figure 7.1 shows an example of a waveform graph. The waveform graph can display plots containing any number of points. The graph also accepts several data types, which minimizes the extent to which you must manipulate data before you display it.

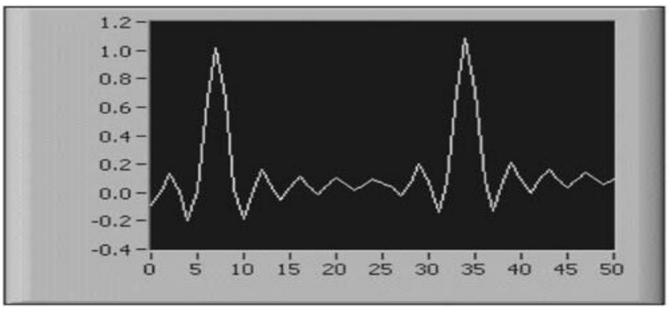


Figure 8.b.i Waveform Graph. 2M

3. Clusters :Clusters group data elements of mixed types. An example of a cluster is the LabVIEW error cluster, which combines a Boolean value, a numeric value and a string. A cluster is similar to a record or a struct in text-based programming languages.

Figure 8.b.ii show the error cluster control and the corresponding terminal created in the block diagram. This cluster consists of a Boolean control (status), a numeric control (code) and a string control (source).

Figure 8.b.ii

Bundling several data elements into clusters eliminates wire clutter on the block diagram. It also reduces the number of connector pane terminals that subVIs need by passing several values to one terminal.

The connector pane has, at most, 28 terminals. If your front panel contains more than want to pass to another VI, group some of them into a cluster and assign the cluster to a terminal on the connector pane.

9.a. Discuss the (i) PC based control (ii) VXI based control 6M

A: PC based control:

The computer (an ordinary PC or PC-compatible) controls each item of external instrumentation and automates the test and calibration procedure, increasing throughput, consistency, and reliability, freeing the test engineer for higher level tasks. A PC-based arrangement thus provides a flexible and highly cost effective alternative to traditional methods. Furthermore, systems can be easily configured to cope with the changing requirements of the user.

In general, PC-based instrumentation and control systems offer the following advantages:

• Flexible and adaptable: the system can be easily extended or reconfigured for a different application.

• The technology of the PC is well known and understood, and most companies already have such equipment installed in a variety of locations.
• Low-cost PC-based systems can be put together at a faction of the cost associated with dedicated controllers.
• Rugged embedded PC controllers are available for use in more demanding applications. Such systems can be configured for a wide range of instrumentation and control applications with the added advantage that they use the same familiar operating system environment and programming software that runs on a conventional PC.
• Availability of an extensive range of PC-compatible expansion cards from an increasingly wide range of suppliers.
• Ability to interface with standard bus systems (including the immensely popular IEEE-488 General Purpose Instrument Bus).

ii.VXI is used in many different applications ranging from test and measurement and ATE, to data acquisition and analysis in both research and industrial automation. Although some VXI systems today are purely VXI, many users are migrating to VXI by integrating it into existing systems consisting of GPIB instruments, VME cards, or plug-in data acquisition (DAQ) boards. We can control a VXI system with a remote general-purpose computer using the high-speed Multisystem eXtension Interface (MXI) bus interface or GPIB. We can also embed a computer into a VXI chassis and control the system directly. Whatever wer system configuration needs may be, VXI offers the flexibility and performance to take on today's most challenging applications.

VXIbus Mechanical Configuration

Physically, a VXIbus system consists of a mainframe chassis that has the physical mounting and backplane connections for plug-in modules, as shown in Figure 9A.. The VXIbus uses the industry-standard IEEE-1014 VMEbus as a base architecture to build upon. As shown in Figure 2, VXI uses the full 32-bit VME architecture, but adds two board sizes and one connector. The P1 connector and the center row of the P2 connector are retained exactly as defined by the VME specification. The VME user-definable pins on the P2 connector and the additional pins on P3, the third VXI connector, implement instrumentation signals between plug-in modules directly on the backplane.

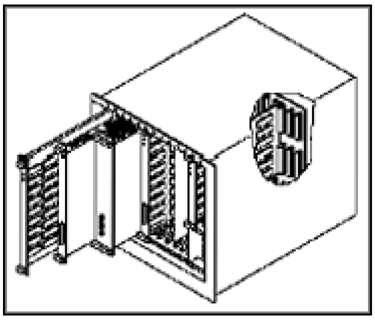


FIG 9a VXI BUS SYSTEM

9.b. Explain the GPIB bus interface clearly. 6M

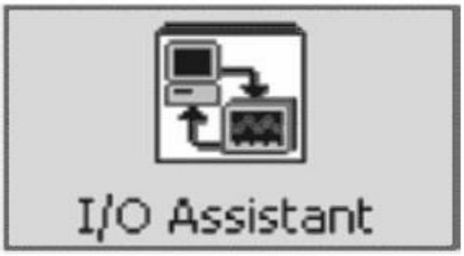
A: The Instrument I/O Assistant is a LabVIEW Express VI which you can use to communicate with message-based instruments and convert the response from raw data to an ASCII representation.

You can communicate with an instrument that uses a serial, Ethernet, or GPIB interface.

The Instrument I/O Assistant organizes instrument communication into ordered steps.

To use Instrument I/O Assistant, you place steps into a sequence. As you add steps to the sequence, they appear in the Step Sequence window.

Use the view associated with a step to configure instrument I/O.



If it does not appear, double-click the Instrument I/O Assistant icon. Complete the following steps to configure the Instrument I/O Assistant.

Step 1: Select an instrument. Instruments that have been configured in MAX appear in the *Select* *an instrument pull-down menu.*

Step 2: Choose a *Code generation type. VISA code generation allows for more flexibility and* modularity than GPIB code generation.

Step 3: Select from the following communication steps using the *Add Step button:*

- **Query and Parse—Sends a query to the instrument, such as *IDN? and parses the returned** string. This step combines the Write command and Read and Parse command.
- **Write—Sends a command to the instrument.**
- **Read and Parse—Reads and parses data from the instrument.**

Step 4: After adding the desired number of steps, click the *Run button to test the sequence of* communication that you have configured for the Express VI.

Step 5: Click the *OK button to exit the Instrument I/O Assistant configuration dialog box.*

In Figure 10.8, a GPIB instrument was set up, then a query (*IDN?) was sent to the instrument. The response was automatically parsed, resulting in a string output.

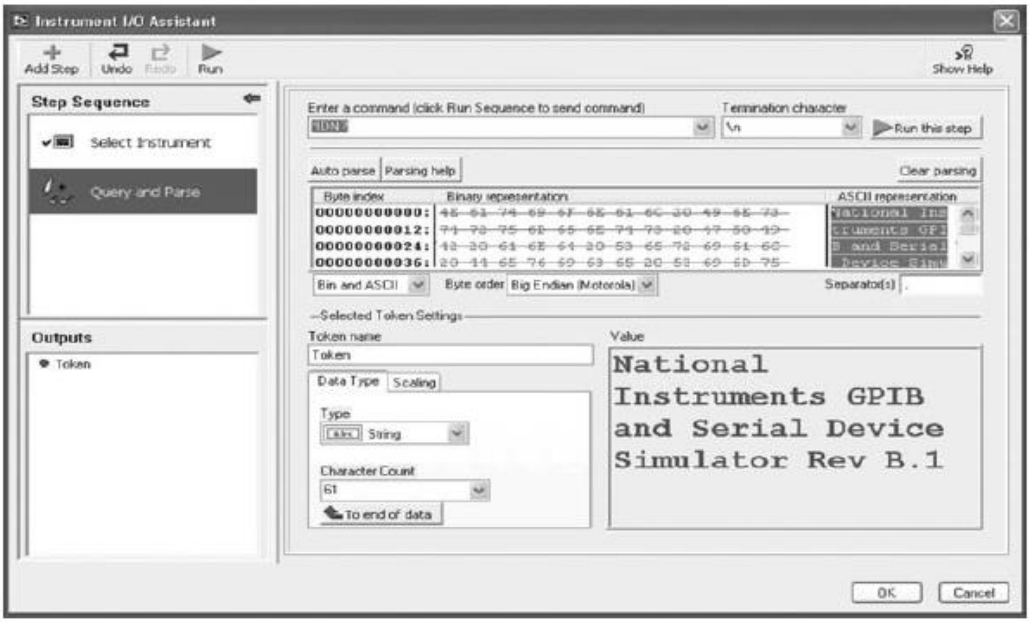


Figure 9.b. Serial Configuration of the Instrument I/O Assistant.