**S.D.G.P Barkot**

**Comouter Organisation and microprocessor Date 28/4/2020**

**Asynchronous Data Transfer**

We know that, the internal operations in individual unit of digital system are synchronized by means of clock pulse, means clock pulse is given to all registers within a unit, and all data transfer among internal registers occur simultaneously during occurrence of clock pulse. Now, suppose any two units of digital system are designed independently such as CPU and I/O interface.

And if the registers in the interface (I/O interface) share a common clock with CPU registers, then transfer between the two units is said to be synchronous. But in most cases, the internal timing in each unit is independent from each other in such a way that each uses its own private clock for its internal registers .In that case, the two units are said to be asynchronous to each other, and if data transfer occur between them this data transfer is said to be **Asynchronous Data Transfer**.

But, the Asynchronous Data Transfer between two independent units requires that control signals be transmitted between the communicating units so that the time can be indicated at which they send data.

This asynchronous way of data transfer can be achieved by two methods:

1. One way is by means of strobe pulse which is supplied by one of the units to other unit. When transfer has to occur. This method is known as “**Strobe Control**”.
2. Another method commonly used is to accompany each data item being transferred with a control signal that indicates the presence of data in the bus. The unit receiving the data item responds with another signal to acknowledge receipt of the data. This method of data transfer between two independent units is said to be “**Handshaking**”.

The strobe pulse and handshaking method of asynchronous data transfer are not restricted to I/O transfer. In fact, they are used extensively on numerous occasion requiring transfer of data between two independent units. So, here we consider the transmitting unit as source and receiving unit as destination.

As an example: The CPU, is the source during an output or write transfer and is the destination unit during input or read transfer.

And thus, the sequence of control during an asynchronous transfer depends on whether the transfer is initiated by the source or by the destination.

So, while discussing each way of data transfer asynchronously we see the sequence of control in both terms when it is initiated by source or when it is initiated by destination. In this way, each way of data transfer, can be further divided into parts, source initiated and destination initiated.

We can also specify, asynchronous transfer between two independent units by means of a timing diagram that shows the timing relationship that exists between the control and the data buses.

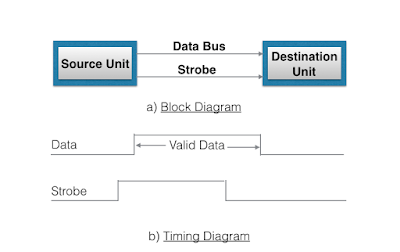
Now, we will discuss each method of asynchronous data transfer in detail one by one.

**1. Strobe Control:**

     The Strobe Control method of asynchronous data transfer employs a single control line to time each        transfer .This control line is also known as strobe and it may be achieved either by source or     destination, depending on which initiate transfer.

**Source initiated strobe for data transfer:**

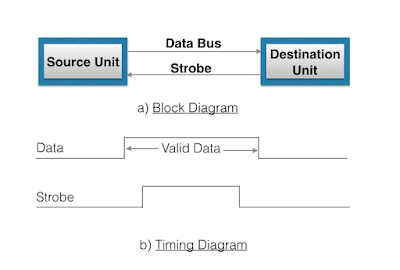
     The block diagram and timing diagram of strobe initiated by source unit is shown in figure below:

[](https://1.bp.blogspot.com/-HsusyfVCiBw/V8-PN4sD4KI/AAAAAAAAAOw/J1Hq9blUGS0HBLJ0JPuM5GYo6Yfe-i9jwCLcB/s1600/Screen%2BShot%2B2016-09-07%2Bat%2B09.22.32.png)

     In block diagram we see that strobe is initiated by source, and as shown in timing diagram, the                source unit first places the data on the data bus. After a brief delay to ensure that the data settle to a        steady value, the source activates a strobe pulse. The information on data bus and strobe control            signal remain in the active state for a sufficient period of time to allow the destination unit to receive        the data. Actually, the destination unit, uses a falling edge of strobe control to transfer the contents of        data bus to one of its internal registers. The source removes the data from the data bus after it        disables its strobe pulse. New valid data will be available only after the strobe is enabled again.

**Destination-initiated strobe for data transfer:**

     The block diagram and timing diagram of strobe initiated by destination is shown in figure below:

[](https://3.bp.blogspot.com/-ch3GPPYxY6g/V8-PnxBgGlI/AAAAAAAAAO0/vrwrYk8AeIwm4rHFzuq_vSt6KmsGjivQQCLcB/s1600/Screen%2BShot%2B2016-09-07%2Bat%2B09.24.17.png)

     In block diagram, we see that, the strobe initiated by destination, and as shown in timing diagram, the      destination unit first activates the strobe pulse, informing the source to provide the data. The source          unit responds by placing the requested binary information on the data bus. The data must be valid            and remain in the bus long enough for the destination unit to accept it. The falling edge of strobe              pulse can be used again to trigger a destination register. The destination unit then disables the                  strobe. And source removes the data from data bus after a per determine time interval.

     Now, actually in computer, in the first case means in strobe initiated by source - the strobe may be a        memory-write control signal from the CPU to a memory unit. The source, CPU, places the word on          the data bus and informs the memory unit, which is the destination, that this is a write operation.

     And in the second case i.e, in the strobe initiated by destination - the strobe may be a memory read        control from the CPU to a memory unit. The destination, the CPU, initiates the read operation to                inform the memory, which is a source unit, to place selected word into the data bus.

**2. Handshaking:**

     The disadvantage of strobe method is that source unit that initiates the transfer has no way of                  knowing whether the destination has actually received the data that was placed in the bus. Similarly, a      destination unit that initiates the transfer has no way of knowing whether the source unit, has actually      placed data on the bus.

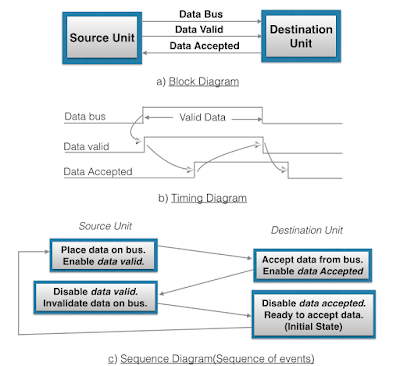
     This problem can be solved by handshaking method.

     Hand shaking method introduce a second control signal line that provides a replay to the unit that            initiates the transfer.

     In it, one control line is in the same direction as the data flow in the bus from the source to                        destination. It is used by source unit to inform the destination unit whether there are valid data in the        bus.The other control line is in the other direction from destination to the source.It is used by the              destination unit to inform the source whether it can accept data. And in it also, sequence of control            depends on unit that initiate transfer. Means sequence of control depends whether transfer is initiated      by source and destination. Sequence of control in both of them are described below:

**Source initiated Handshaking:**

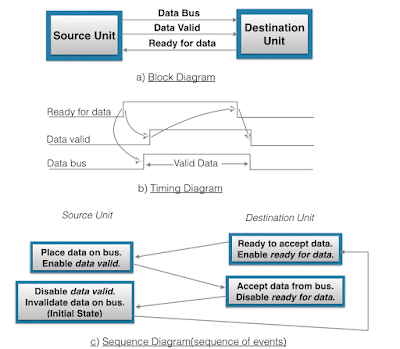
     The source initiated transfer using handshaking lines is shown in figure  below:

[](https://1.bp.blogspot.com/-mK55LGTjugA/V9FY7CoXDtI/AAAAAAAAAPs/uMNIQaCBrVg7e8R2HpL1Gc72vFzyErx3ACLcB/s1600/Screen%2BShot%2B2016-09-08%2Bat%2B14.59.30.png)

     In its block diagram, we se that two handshaking lines are "data valid", which is generated by the            source unit, and "data accepted", generated by the destination unit.  
  
     The timing diagram shows the timing relationship of exchange of signals between the two                        units. Means as shown in its timing diagram, the source initiates a transfer by placing data on the bus      and enabling its data valid signal. The data accepted signal is then activated by destination unit after it      accepts the data from the bus. The source unit then disable its data valid signal which invalidates the        data on the bus. After this, the destination unit disables its data accepted signal and the system goes        into initial state.The source unit does not send the next data item until after the destination unit shows      its readiness to accept new data by disabling the data accepted signal.  
  
     This sequence of events described in its sequence diagram, which shows the above sequence in            which the system is present, at any given time.

**Destination initiated handshaking:**

     The destination initiated transfer using handshaking lines is shown in figure  below:

[](https://1.bp.blogspot.com/-G10r82h91tU/V9FaG0KQcKI/AAAAAAAAAPw/Bn25tXqutUUfDD9ZwEAnO5R1a7RgC8OhQCLcB/s1600/Screen%2BShot%2B2016-09-08%2Bat%2B15.02.22.png)

     In its block diagram, we see that the two handshaking lines are "data valid", generated by the source      unit, and "ready for data" generated by destination unit. Note that the name of signal data accepted          generated by destination unit has been changed to ready for data to reflect its new meaning.

     In it, transfer is initiated by destination, so source unit does not place data on data bus until it                    receives ready for data signal from destination unit. After that, hand shaking process is some as that        of source initiated.

     The sequence of event in it are shown in its sequence diagram and timing relationship between              signals is shown in its timing diagram.

     Thus, here we can say that, sequence of events in both cases would be identical. If we consider              ready for data signal as the complement of data accept. Means, the only difference between source          and destination initiated transfer is in their choice of initial state.