An interface system for asynchronous data

A. G. ANNAN

University of Stirling, Stirling FK9 4LA, Scotland

and

M. SAVVIDES

Signal Processing Section, Polytechnic of Central London, London W1M 8JS, England

This article describes a system for interfacing items of computing equipment in circumstances where the short-term data rate of the source exceeds that of the receiver device. Outline circuitry is given for an application permitting high-speed input to an event-driven time and event data logger.

In the design of data collection systems for psychology, a problem may arise when an interface is required between system components that operate at differing data rates. Standard asynchronous control may provide an appropriate solution in situations where the rate of input is not critical and where the data source may wait until the receiver is ready to accept the next item. However, data rates can be constrained by factors such as sampling criteria, manual response speeds, mechanical punch limits, and so on, and thus be incompatible with the variable timing delays and single level of buffering inherent in normal asynchronous communication. A new element in large-scale integration, first-in first-out (Fi-Fo) memory, provides an elegant solution to this problem. This type of memory is ideally suited to inclusion as a component in an interface, and allows multiple-level memory blocks to be used as serial data stacks to connect unsynchronized data streams. An interface may be designed with Fi-Fo memory to provide an asynchronous channel with a large buffering capacity.

First-in first-out memories are read-write storage systems which automatically retain the sequence of input data, have separate input and output ports, and, in particular, enable input and output to occur independently at different rates. The memory is organized in parallel sections to process data in word form and will have multiple internal storage levels. Any data clocked in at the entry port will automatically descend to the lowest unoccupied level within the store, eventually passing out to the receiver device at the latter's clock rate. (See Figure 1.)

The Fairchild 3341 MOS integrated circuit is a 4-bit 64-level Fi-Fo store of the type described. The device includes TTL-compatible interface circuitry and is controlled at its two ports by simple OUTPUT READY/SHIFT OUT and INPUT READY/SHIFT IN signals. The

Figure 1. FI-FO concept.

data available on the output registers may be read when output ready is "true," but is only replaced by the next word after a SHIFT OUT signal is given.

The storage interface described below was designed to link event category and time check units to a paper-tape data logger. The logger is a Solartron data transfer unit (DTU), modified for event-driven operation. On command, the DTU reads event and time data from two 6-decade parallel BCD inputs, serializes, recodes, and inserts format characters, outputting the final record in bit-parallel character-serial ASCII code to a paper-tape punch.

The event register unit consists of six channels, each comprising a DEC/BCD conversion, ac coupled to a 4-bit store giving a buffered BCD output. Each channel has a selection switch which, when closed, allows any stored bit in that channel to access a common DATA READY (DR) line. Thus, information may be held in certain channels of the event registers and passed to the Fi-Fo interface later with other event data and the time at which an input was made on a channel whose selection switch was closed. This feature reduces throughput in the system and is useful, for example, in experimental trials, where stimuli and other parameters can be logged without separate timing and can be subsequently output with a response category and response time. The event registers are cleared by the interface return signal DATA LOADED (DL).

Figure 2. Data logging system.

The time data is cumulative and is sampled by a method which insures that all BCD levels are effectively at rest during the transfer.

The relation of system units and signals is as in Figure 2. It will be apparent that the storage interface could have been designed for operation between the logger and its punch, using a serial rather than a parallel organization. This option was not selected, as the amount of Fi-Fo for a given buffering capability would have been the same as in parallel organization, and a 6-decade BCD buffer would still have been necessary between the clock and DTU to hold the time data stationary during serialization.

System specification: Event inputs—54 lines arranged in six decimal decades; signal input by switch closure or current sink. Time input—6-decade BCD, seconds $\times 10^{-4}$. Output—via Solartron DTU to paper tape at 80 characters/sec, ASCII.

Interface specification: Input—12-decade BCD; data ready and data loaded signals. Output—12-decade BCD; alert and transfer complete signals; 64 levels of internal storage. Input and output may operate simultaneously under control of independent external clocks. Input resolution: switched .1 or 5 msec. The maximum short-term input capacity of the interface is therefore 12 parallel decades of BCD at 10 kHz. This provides a resolution which will allow several experiments to simultaneously access a single logger.

The operation of the interface circuitry, starting with the input control loop (Figure 3), is as follows: Input data to the event register unit raises the signal "DATA READY" (DR), setting bistable B1 and thus enabling gate G1. Time data, in BCD, is continuously available and a signal from the least significant decade (level 1), via gate G1, will start the settle delay. This delay in the signal path insures that all signal levels will be at rest during transfer, even if BCD ripple counters are used as a clock source. When the settle delay finishes, pulse former PF1, via gate G2, enabled by the Fi-Fo "INPUT READY," triggers pulse former PF2, giving a SHIFT IN pulse to the Fi-Fo. The 12 decades of time and event data will load into the Fi-Fo store and automatically descend to the lowest unoccupied internal position.

INPUT READY will have been dropped by the Fi-Fo at the start of SHIFT IN and will not be raised again, enabling the next input cycle, until the current data block transfers out of the top internal location.

The trailing edge of SHIFT IN resets bistable B1 and triggers pulse former P3, which resets the external data registers. Thus, the duration of SHIFT IN and the clock timebase together control the input data rate. SHIFT IN may be set to a 50-microsec or 5-msec duration. The latter position insures that the input cycle does not respond to switch contact noise which would otherwise cause repeated assertions of the same input data.

The output control logic is as shown in Figure 4. Data transferring down the internal locations of the Fi-Fo appears on the output registers and raises the signal OUTPUT READY, setting bistable B2, whose output holds the ALERT line on the data logger. Transfer of the data through the logger's serializing logic to the paper-tape punch then occurs, and at the end of the data output sequence the logger returns the signal ALERT COMPLETE. This signal resets bistable B2, clears the logger ALERT, and sends a SHIFT OUT to the Fi-Fo, causing the last store location to be available for the next internal transfer and output cycle. Transfer complete also triggers pulse former PF4, which inhibits the next output cycle for 25 msec, thus allowing time for the logger to insert format characters in the output record.

The design principles of this storage device can be readily adapted to interface many other digital devices,

Figure 3. I/P control of interface.

Figure 4. O/P control of interface.

when operating in an asynchronous mode. In particular, this type of interface allows loggers, printers, and other output devices to be used in circumstances where the short-term input data rates exceed the normal operating speed of the unit.

The system can resolve near-simultaneous event calls and may thus be accessed concurrently by several experiments. The event numbers and decades appearing in the

output record may be preassigned between the various experimenters and a simple computer program used to sort the data into blocks relating to the individual experiments.

(Received for publication October 29, 1976;
revision accepted January 15, 1977.)