US 20100290466A1

(19) **United States**

(12) **Patent Application Publication**

Morris

(10) **Pub. No.: US 2010/0290466 A1**

(43) **Pub. Date: Nov. 18, 2010**

(54) **ROUTING OF DATA STREAMS**

(75) Inventor: **Matt Morris**, Bristol (GB)

Correspondence Address:
GARDERE WYNNE SEWELL LLP
INTELLECTUAL PROPERTY SECTION
3000 THANKSGIVING TOWER, 1601 ELM ST
DALLAS, TX 75201-4761 (US)

(73) Assignee: **STMICROELECTRONICS LIMITED**, Bristol (GB)

(21) Appl. No.: **12/781,118**

(22) Filed: **May 17, 2010**

Related U.S. Application Data

(63) Continuation of application No. 10/779,466, filed on Feb. 16, 2004, now Pat. No. 7,720,112.

(30) **Foreign Application Priority Data**

Feb. 24, 2003 (EP) 03251091.9

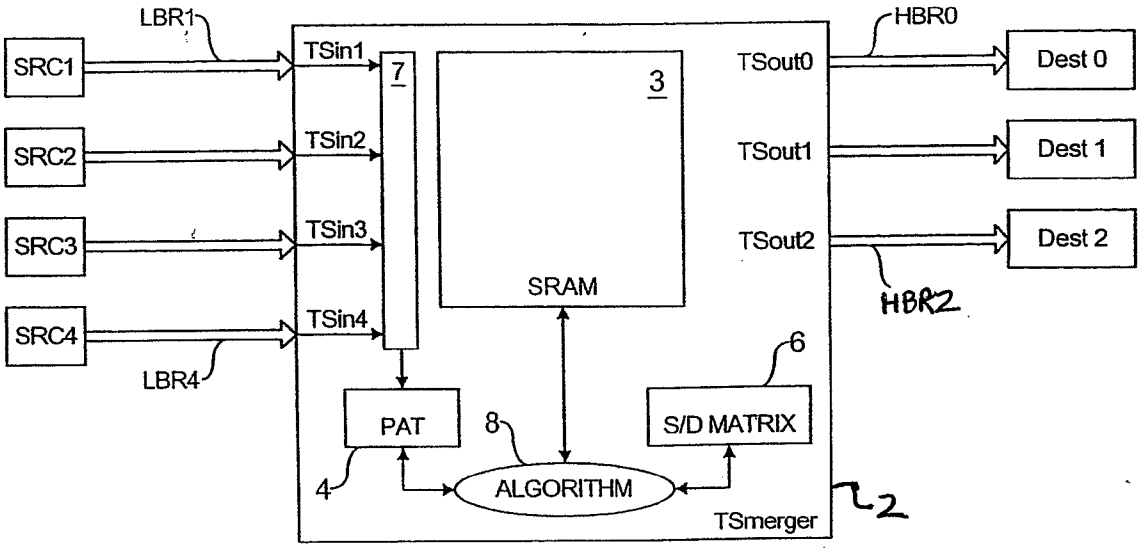
Publication Classification

(51) **Int. Cl.**
H04L 12/56 (2006.01)

(52) **U.S. Cl.** **370/391**

(57) **ABSTRACT**

The routing of data streams is discussed, and particularly routing one or more incoming streams to one or more output destination ports. The ability to merge incoming streams is discussed so that several low bit rate input packet streams can be merged into a higher bit rate output stream. An assignment data structure identifies for each input stream the or each destination to which it is to be routed, and a packet allocation data structure holds information about the packets and information about the destination of the packets to allow a memory holding the packets to be controlled accordingly.



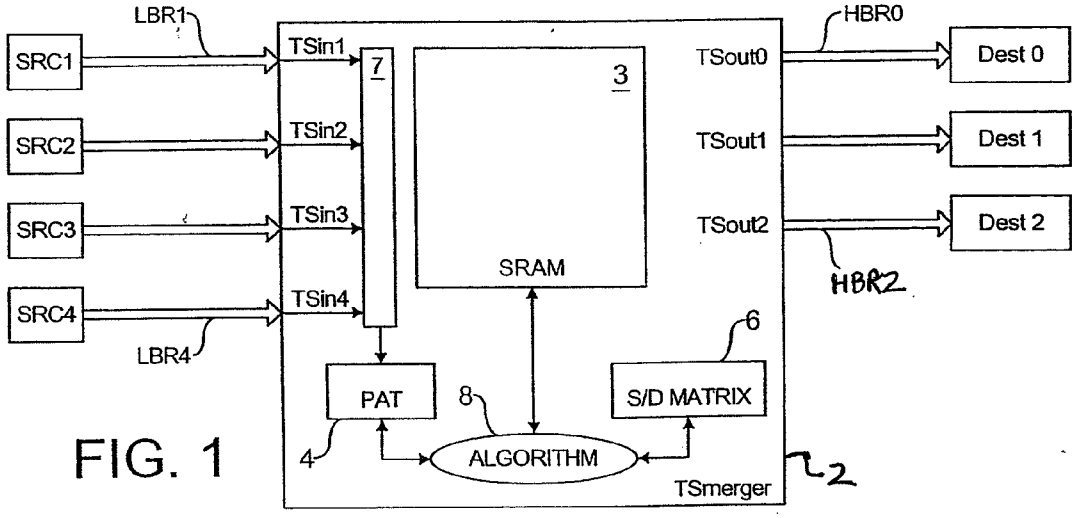


FIG. 1

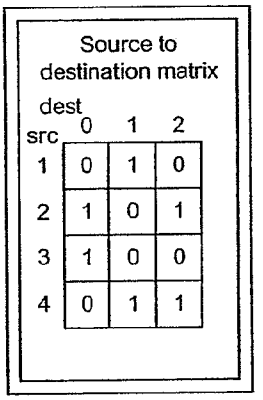


FIG. 2

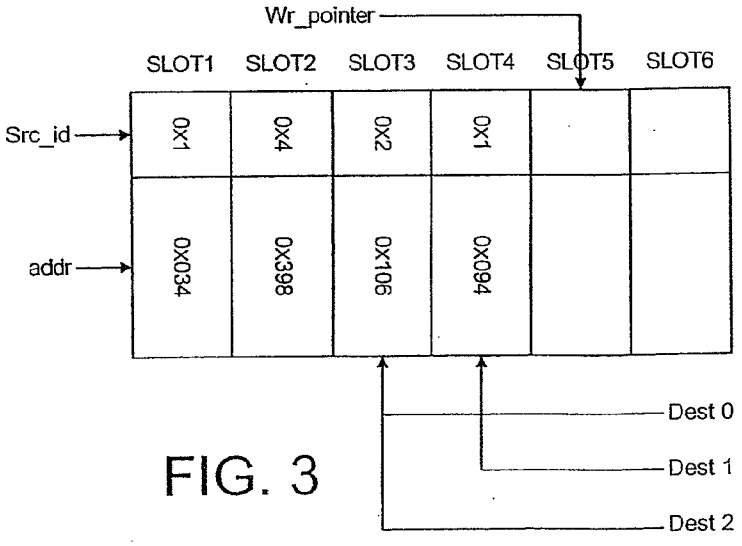


FIG. 3

At start: all destination pointers point to NULL

FIG. 4A

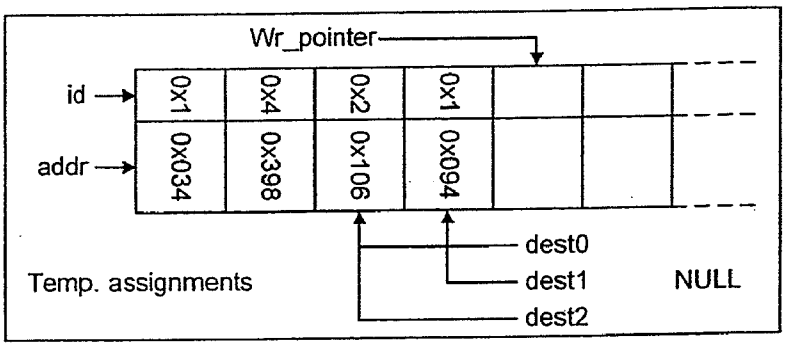


FIG. 4B

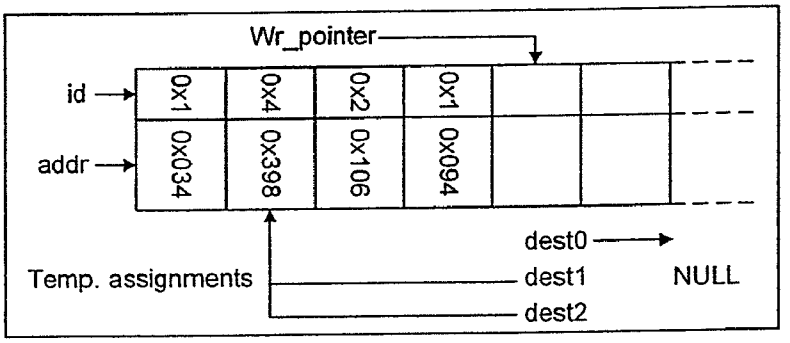


FIG. 4C

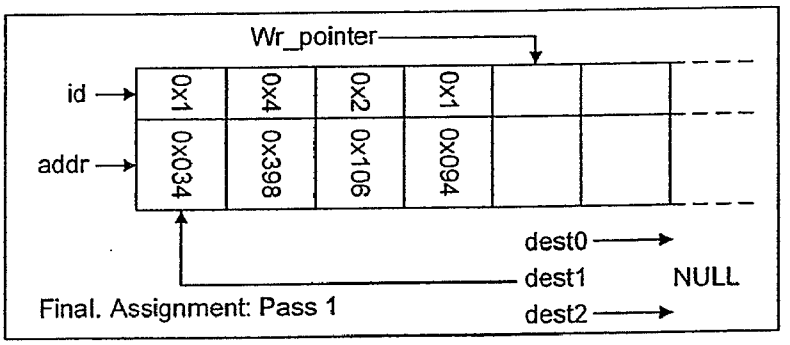
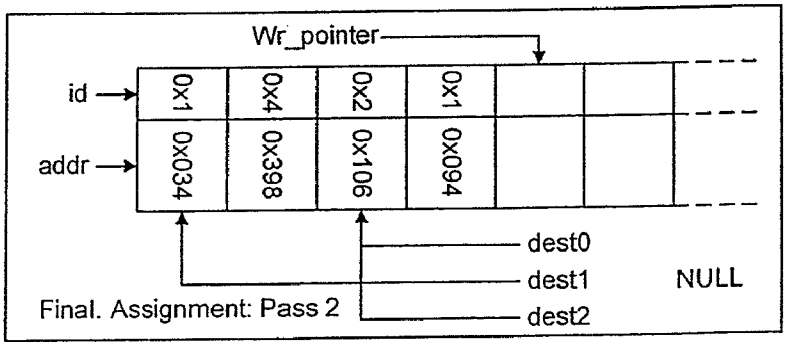


FIG. 4D



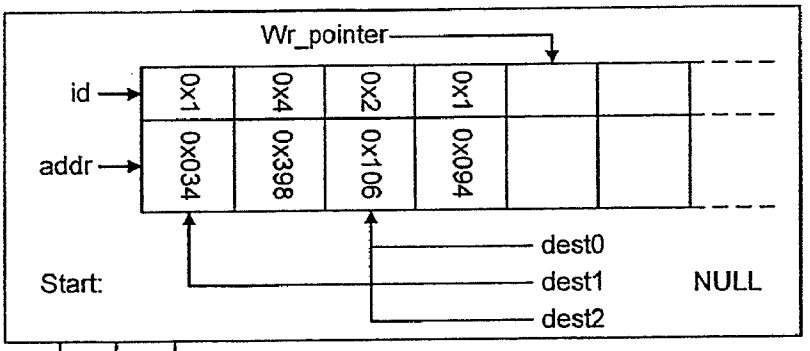


FIG. 5A

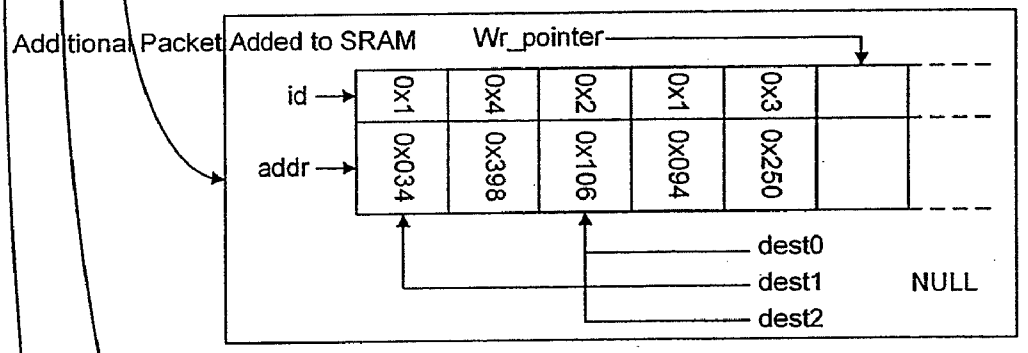


FIG. 5B

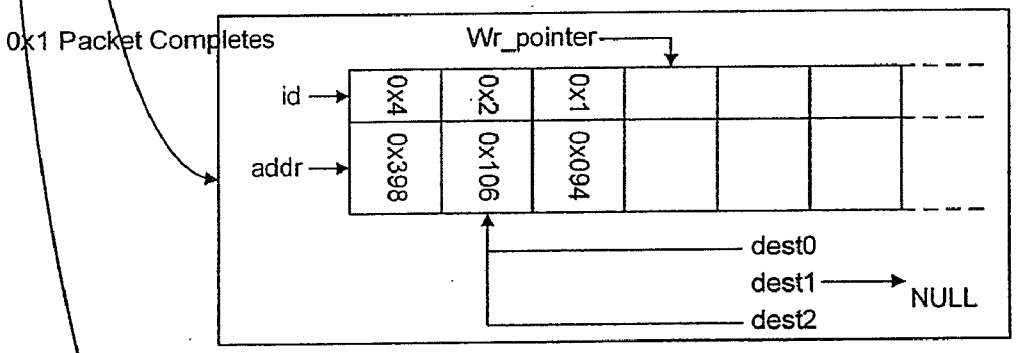


FIG. 5C

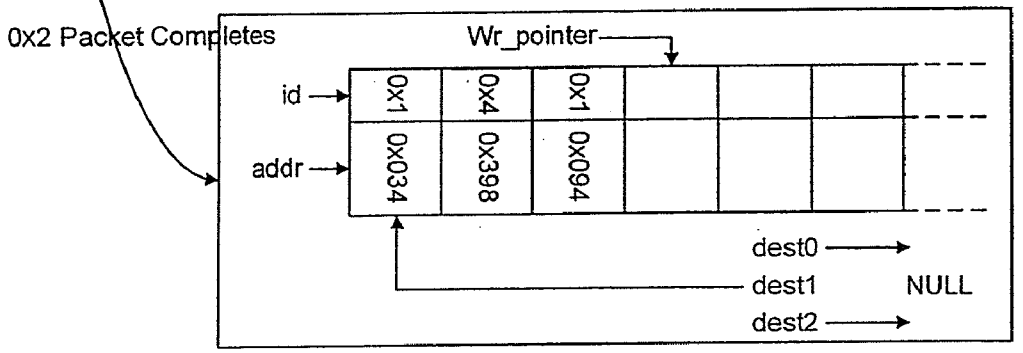


FIG. 5D

ROUTING OF DATA STREAMS

[0001] PRIORITY CLAIM

[0002] The present application is a continuation of U.S. patent application Ser. No. 10/779,466 filed Feb. 16, 2004, which claims priority from European Patent Application No. 03251091.9 filed Feb. 24, 2003, the disclosures of which are hereby incorporated by reference.

BACKGROUND OF THE INVENTION

[0003] 1. Technical Field of the Invention

[0004] The present invention relates to the routing of data streams, and in particular streams constituted by data packets.

[0005] 2. Description of Related Art

[0006] Networks today provide connectivity to a variety of devices such as servers, personal computers, memory storage systems, etc. Most networks provide a wide range of applications and technologies to allow for a multitude of communications to one or more destinations.

[0007] As is known in the art, illustrated by U.S. Pat. No. 6,081,522, a network may employ a multi-layer network element to forward received packets from an input port to one or more output ports. The received packets may be scanned for different types of forwarding information, i.e., layer 2 information, layer 3 information, layer 4 information, etc. Based on the results of the scan, a determination is made as to the most appropriate combination of layer 2 or layer 3 forwarding decisions for the received packet. Once the forwarding decisions are made, the received packet is transmitted to the destination.

[0008] U.S. Pat. No. 5,905,725 relates to a network utilizing a router to switch a packet between a source and a destination. The router may include a plurality of ports connected to various sources and destinations. The ports connected to sources are termed input ports and each input port includes a data handler. The router also includes output ports and a memory divided into a plurality of memory banks. The data handler divides each packet into one or more fixed length cells. The fixed length cells are transmitted to an input switch that writes a single cell into a cell slot time span to each memory bank. The input switch reads a key from the packet and, based on a destination indicated by a key, an output port associated with the destination is determined. An output switch is alerted to the determination and transfers the determination to the appropriate output port. The output port initiates transfer of the packet from the memory and the output switch.

SUMMARY OF THE INVENTION

[0009] A transport stream multiplexer (TSMUX) has been implemented which can route a data stream received as an input to any one of a number of outputs. An aspect of the present invention expands the capabilities of the multiplexer so that a number of input streams can be merged and can be directed to one or more of a plurality of output destinations.

[0010] According to one aspect of the present invention there is provided a stream routing unit for routing each of a plurality of input packet streams to any of a plurality of destinations. The stream routing unit includes a plurality of input ports for receiving respective input streams, a plurality of output ports associated with respective destinations to which the input packet streams can be routed, and storage means for holding packets of the input packet streams at addressable locations each identifiable by an address. The stream routing unit also includes an assignment data structure identifying for each input stream at least one destination to which each input packet stream is to be routed. The stream routing unit further includes a packet allocation data structure holding for each new incoming packet a source identifier identifying the origin of the packet and the address in the storage means where the packet is held. The packet allocation data structure also holds information identifying the intended destination of the packet derived from the assignment data structure.

[0011] Another aspect of the invention provides a method of routing packet streams from a plurality of sources to any of a plurality of destinations. The method includes receiving the packet streams, identifying for each input packet stream at least one destination to which each input packet stream is to be routed using an assignment data structure, holding each packet of the packet stream at an addressable location identifiable by an address in a storage means, and holding for each new incoming packet a source identifier identifying the origin of the packet and the address in the storage means where the packet is held. The method also includes holding information identifying the intended destination of the packet derived from the assignment data structure, and using the information identifying the intended destination to route the packet from the storage means to at least one output port associated with the respective identified destination(s).

[0012] A further aspect of the invention provides a communications system which utilizes a stream routing unit as hereinabove defined together with a plurality of sources for the input packet streams and destinations for receiving output streams.

[0013] As described below, the preferred embodiment of the invention provides an intellectual property (block of logic) known as TSmerger which merges multiple lower bit rate transport streams to a single higher bit rate transport stream for processing by a single programmable transport interface (PTI). For example, nine input streams can be merged into three output streams, with each input stream being able to be routed to any output stream, or to multiple destinations.

[0014] In the described embodiment, the TSmerger IP is implemented by storing packets of incoming streams in a single SRAM in a stream merger unit, which effectively behaves as a series of first-in first-out buffers (FIFOs).

BRIEF DESCRIPTION OF THE DRAWINGS

[0015] A more complete understanding of the method and apparatus of the present invention may be acquired by reference to the following Detailed Description when taken in conjunction with the accompanying Drawings wherein:

[0016] FIG. **1** is a schematic diagram of a communications system incorporating a stream routing unit;

[0017] FIG. **2** illustrates a source to destination matrix;

[0018] FIG. **3** illustrates a packet allocation table;

[0019] FIGS. **4**A to **4**D illustrate the assignment of destination pointers associated with the packet allocation table; and

[0020] FIGS. **5**A to **5**D illustrate the effect on the packet allocation table of adding packets into and removing packets from the memory.

DETAILED DESCRIPTION OF THE DRAWINGS

[0021] FIG. **1** is a schematic diagram of a stream routing unit referred to herein as a TSmerger unit. The TSmerger unit is denoted generally by reference numeral **2** and has a plurality of input ports TSin**1** . . . TSin**4** for receiving respectively each of a plurality of low bit rate input packet streams LBR**1** . . . LBR**4**. It will be appreciated that in practice any number of input streams may be present, four being illustrated by way of example only. Each input stream LBR**1** . . . LBR**4** is derived from a respective source SRC**1** . . . SRC**4**. The TSmerger unit **2** similarly has a plurality of output ports TSout**0** . . . TSout**3**, three of which are illustrated by way of example. These output ports are for the output of high bit rate output streams labeled HBR**0** . . . HBR**2**. Each output stream is supplied to a respective destination dest**0**, dest**1**, dest**2** respectively. These destinations can take any suitable form, for example they can be programmable transport interfaces (PTI) which process the stream data, or they could be audio or video systems or anything capable of handling a data stream.

[0022] The TSmerger unit **2** itself is capable of merging the lower bit rate transport streams to individual higher bit rate streams for processing by respective single PTIs. Each input stream can be routed to any output stream, and each input stream may be distributed to multiple output streams and thereby to multiple destinations.

[0023] The TSmerger unit **2** illustrated in FIG. **1** includes a memory (SRAM) **3** for holding packets of the input streams, a packet allocation table **4** (PAT), a source to destination matrix **6** and a processing means **8** which implements an algorithm to control removal of packets from the SRAM **3**.

[0024] The processing means **8** controls the removal of packets from the SRAM **3** to the destinations dest**0**, dest**1**, dest**2** in such a way as to maintain maximum bandwidth, while allowing any source stream to go to any destination. Packets from multiple sources are merged without breaking individual packets (i.e. streams are merged at the packet level and not at the byte level) and packets from a single stream are read from the SRAM in the correct order, that is in the order in which they arrived.

[0025] To maintain maximum bandwidth, in the preferred embodiment each packet is only read from the SRAM **3** once, so that if a packet from a particular input port is destined for more than one destination, the packet is only output from the SRAM **3** when the ports for both of those destinations are free.

[0026] The source to destination matrix **6** is a data structure illustrated in FIG. **2**. This matrix **6** maps sources to destinations and can be changed on the fly. The source numbers are indicated on the left hand side of the rows of the matrix **6**, and the destinations are illustrated at the top of the columns of the matrix **6**. A "1" in each square of the matrix indicates that a particular source is to be mapped to a particular destination. A "0" indicates that that source stream must not be mapped to that particular destination. As is clear from the matrix **6** in FIG. **2**, some source streams (i.e., source **2** and source **4**) may be mapped to more than one destination. Although sources **2** and **4** are shown as mapping to more than one destination, any of the sources may map to one or a plurality of destinations.

[0027] FIG. **3** illustrates the packet allocation table **4** which takes the form of a second data structure. This data structure takes the form of an array including a plurality of slots SLOT**1**, SLOT**2**, etc, each slot including a source identifier (src_id) of a particular packet in association with the address (addr) which is the start address of that packet in the SRAM **3**. The source identifier is inserted into the packet header of each packet at the respective input port of the TSmerger unit **2** at which the packet is received. The source identifier insertion circuitry is labeled **7** in FIG. **1**. The packet allocation table **4** includes a write pointer wr_pointer and three destination pointers, dest**0**, dest**1**, dest**2** each associated with a particular destination as illustrated in FIG. **1**. The pointers are implemented in any suitable known way. In FIG. **3** the write pointer is shown pointing to the next available empty slot (SLOT**5**) in the packet allocation table **4**. The destination pointers dest**0**, dest**1**, dest**2** are shown pointing to the two top full slots of the array (SLOTS **3**, **4**) illustrating the temporary assignment of those pointers when those slots of the array were just filled as will be described in more detail later.

[0028] FIGS. **4**A to **4**D illustrate how the algorithm assigns destination pointers. FIG. **4**A illustrates the state of the pointers as in FIG. **3**, that is with the dest**0** and dest**2** pointers directed at SLOT**3** holding the packet from source **2** and dest**1** pointer directed at SLOT**4** holding the most recently loaded packet from source **1**. This is the status when an incoming packet is newly loaded into the SRAM **3**.

[0029] The next temporary assignment of destination pointers is illustrated in FIG. **4**B. The source identifier in the next slot down of the array, SLOT**2**, is read which identifies source **4** SRC**4**. From the source to destination matrix **6** it is determined that packets from this source are destined for destinations **1** and **2** and therefore the destination pointers dest**1**, dest**2** are realigned to this slot. The destination pointer dest**0** is reassigned to null. The source to destination matrix **6** acts as an assignment data structure for identifying the destinations to which the input packet streams are routed.

[0030] The next assignment of destination pointers is shown in FIG. **4**C. This represents the first part of the final assignment, because the assignment algorithm has reached the end most slot, SLOT**1**, of the array. This slot holds the packet from source **1** which is destined for destination **1** and therefore the dest**1** pointer is assigned to this slot. The dest**0** and dest**2** pointers are assigned to null.

[0031] FIG. **4**D illustrates the next pass in the final assignment. The source identifier in SLOT**2** of the array identifies SRC**4** as the source which is destined for destinations **1** and **2**. There is no point assigning the destination pointer dest**2** to this packet because the destination pointer dest**1** has already been assigned to the packet which is identified in the SLOT**1** of the array and, for bandwidth reasons, the packet should be removed to both destination ports simultaneously. Therefore, no destination pointers are assigned to this slot. In the next slot is a packet from source **2** which is destined for destinations **0** and **2**, and so these destination pointers are set for that slot.

[0032] After the assignment of destination pointers has been completed, the algorithm controls the SRAM **3** to output the identified packets according to the status of the destination pointers in the packet allocation table.

[0033] FIGS. **5**A to **5**D illustrate the effect of moving and adding packets into the SRAM **3**. FIG. **5**A shows the status of the packet allocation table **4** in FIG. **4**D, that is with four slots full, representing four packets in the SRAM **3** and the destination pointers having been finally assigned. FIG. **5**B shows the effect of adding an additional packet to the SRAM **3**. This

packet has come in from source **3** and data identifying the packet is added into the vacant slot, SLOT**5**, of the packet allocation table pointed to by the write pointer wr_pointer. The write pointer is incremented to point to the next vacant slot, SLOT**6**. The destination pointers remain in place.

[0034] FIG. **5**C shows the state of the packet allocation table when the packet identified in the first slot of the table has been completely transmitted from the SRAM **3** out of its allocated destination port dest**1**. Note that the write pointer wr_pointer has been decremented to point to the next available slot, SLOT**4**, and that the assignment of the destination pointer dest**1** has been allocated back to null. The next packet to be transmitted from the SRAM **3** is that identified by the data in SLOT**2** and this packet is transmitted out of the destination ports dest**0** and dest**2** as identified by the destination pointers.

[0035] FIG. **5**D shows the effect of removing this packet, i.e. the one identified by the data in SLOT**3** in FIG. **5**A, from the SRAM **3**, before the one identified by the data in SLOT**1**. In this case the data defining the subsequent packet moves up one slot in the array and the destination pointers dest**0** and dest**2** are reassigned to null.

[0036] It will be appreciated that whether or not packets are removed from the SRAM depends on the capability of destinations to receive them. When destinations are capable of accepting data, they return a signal to the TSmerger unit **2** indicating that they can accept data and then a packet is transmitted. The speed at which packets are removed from the SRAM may also depend on the length of the packets. Generally, each packet stream will contain packets of a common length, although the packet length can differ between different packet streams.

[0037] Although preferred embodiments of the method and apparatus of the present invention have been illustrated in the accompanying Drawings and described in the foregoing Detailed Description, it will be understood that the invention is not limited to the embodiments disclosed, but is capable of numerous rearrangements, modifications and substitutions without departing from the spirit of the invention as set forth and defined by the following claims.

What is claimed is:

1. A method for delivering incoming packets to at least one destination, comprising:
- mapping at least one source to at least one destination;
- associating a source and at least one destination for a particular packet with a memory location at which the particular packet is stored; and
- controlling removal of the incoming packets from a memory to at least on destination wherein the incoming packets have a lower bit-rate than packets delivered to the at least one destination.

2. A method for controlling removal of packets from a memory to at least one destination, comprising:
- reading each packet from a memory;
- determining a number of particular destinations to which each read packet is destined for delivery;
- checking whether a port for each particular destination is available;
- if the packet is destined for delivery to more than one particular destination, outputting the packet for multi-destination delivery only if a port for each of the particular destinations is simultaneously available; and
- if the packet is destined for delivery to only one particular destination, outputting the packet only when a port for ht at particular destination is available.

3. An apparatus for controlling removal of packets from a memory at least one destination, comprising:
- a processor for reading each packet from a memory and determining a number of particular destinations to which each read packet is destined for delivery; and
- a plurality of outputs for receiving the read packet, wherein if the packet is destined for delivery to more than one particular destination, each output in the plurality of outputs receiving the packet only if a port for each of the particular destinations is simultaneously available.

* * * * *