

TRANSPUTER ARCHITECTURE



What is Transputer ?

- The first single chip computer designed for message-passing parallel systems, in 1980s, by the company INMOS
- *Transistor Computer*. Goal was produce low cost low power chips to form a complete processor, just as transistors had earlier
- It has a RISC type of instruction set

Transputer Versions

- First generation

16 bit transputers: T212 → T222 → T225

32 bit transputers without a floating unit: T400 → T414 → T425 → T426

32 bit transputers with a floating unit : T800 → T801 → T805

All have the same architecture, similar instruction sets and fully compatible communications links.

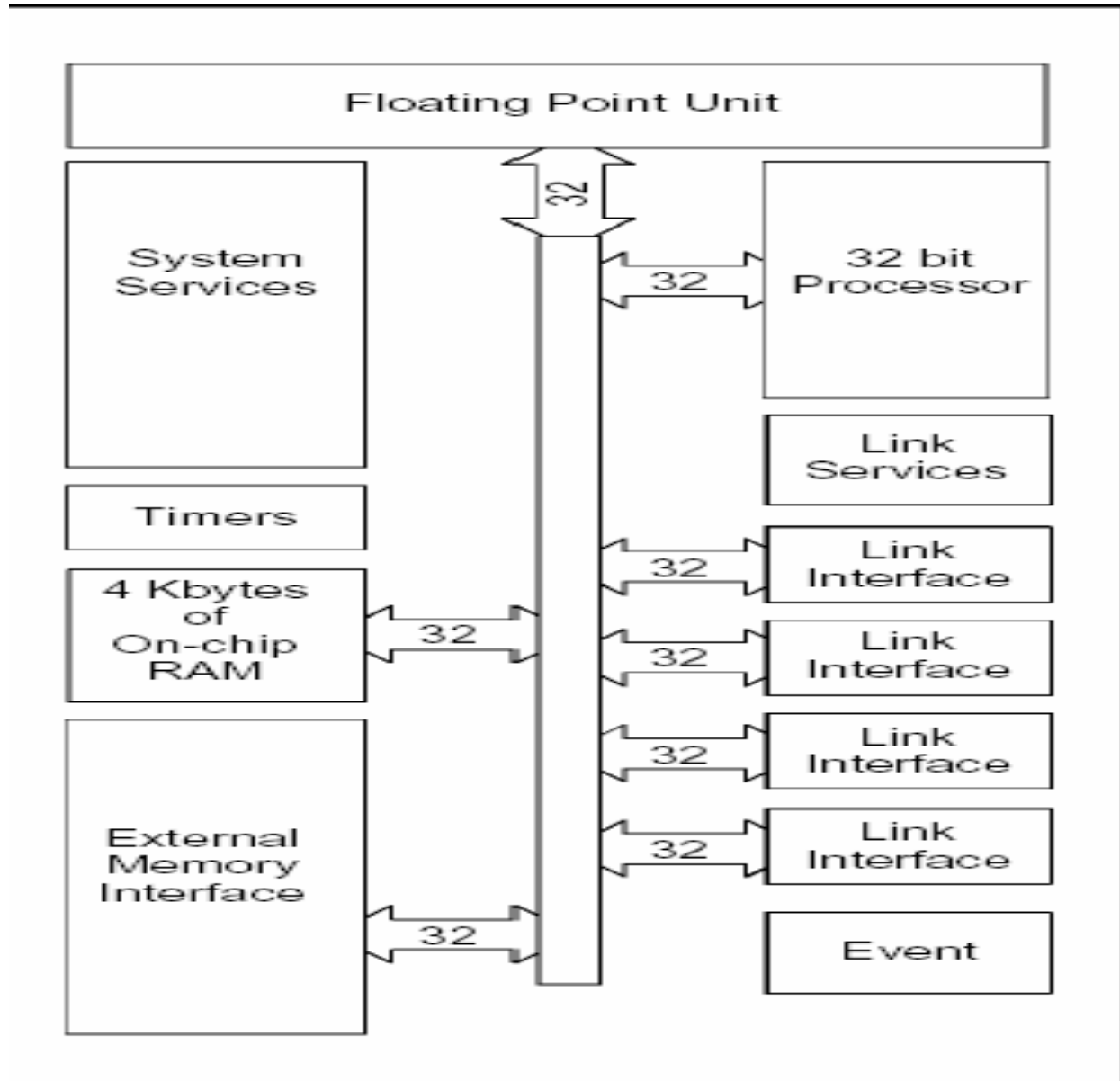
Second Generation

64 bit transputer with a floating unit : T9000

Although general architecture much the same, it was a new design and is much more complex chip than its predecessors.

And it never saw the light of day

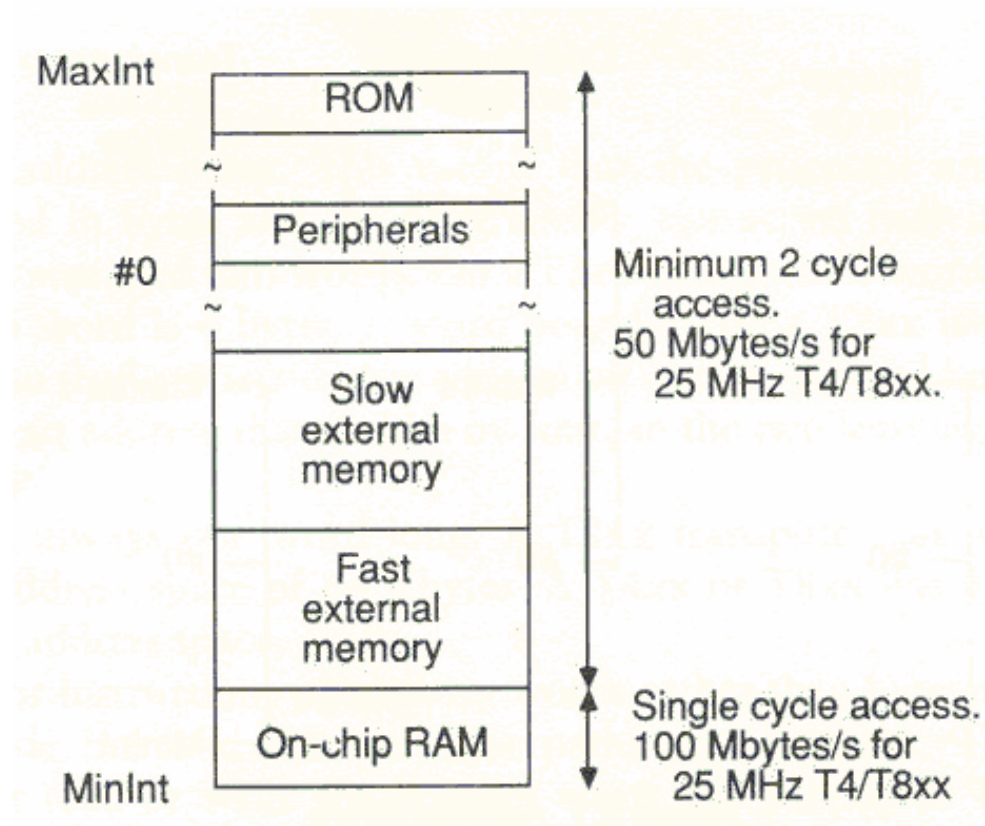
TRANSPUTER ARCHITECTURE



Memory

- All transputers address bytes
- 32 bit addresses gives 4 Gigabytes address space
- The range of addresses is unusual : starts at #800000 and goes up to #7FFFFFF, with #000000 lying in the middle. Do not need to calculate unsigned arithmetic so it simplifies the ALU and reduce the number of instructions.

Transputer Memory Map



External Memory Interface

- There are several designs of emi used on the various transputers for different memory systems. Divided into two types:
- Fast EMI : used in 16 bit transputers, allows mixed systems of SRAM, ROM and other devices.
- Programmable EMI : provided on 32 bit transputers to support dynamic memories, mixed memory systems and memory mapped devices.

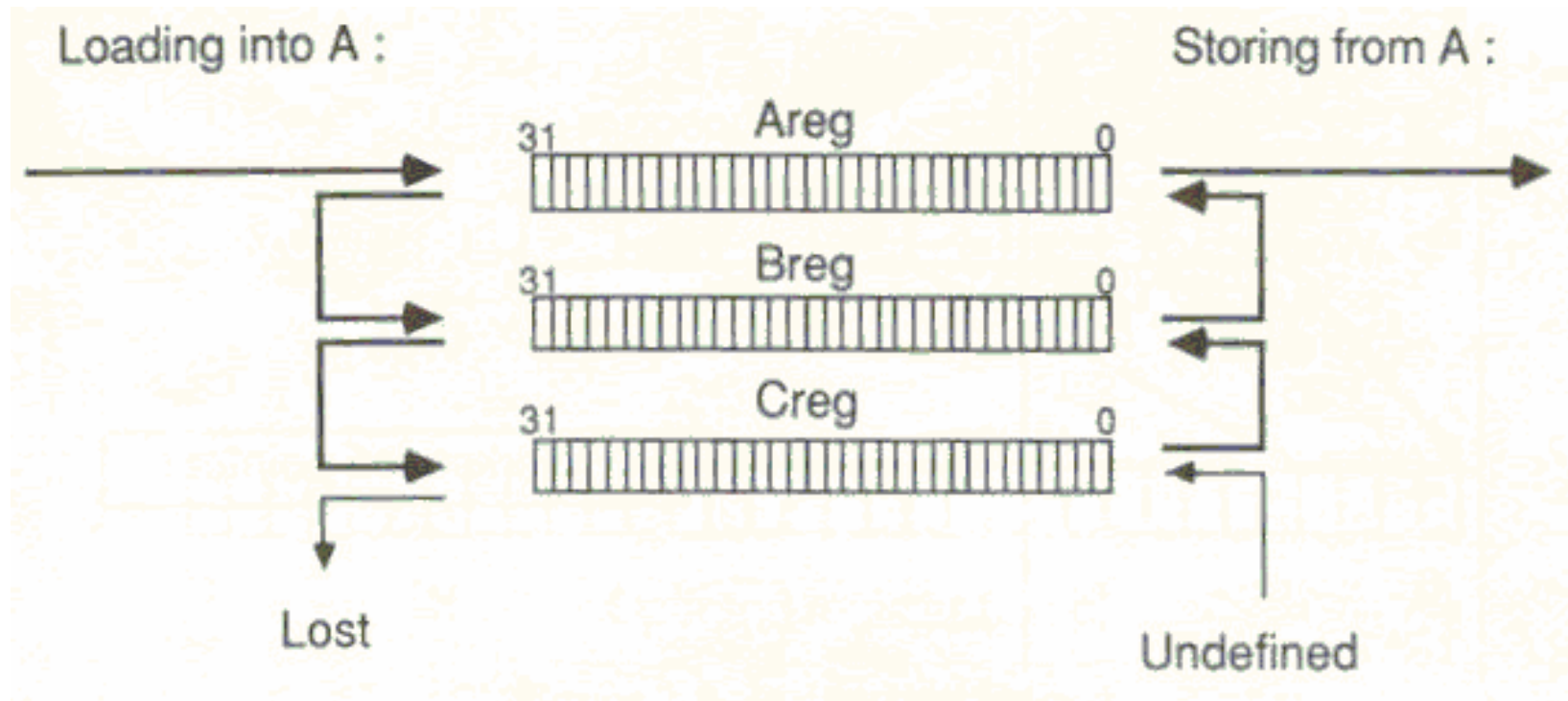
Processor

- The transputer processor is in some ways a conventional microprocessor. It executes one instruction at a time
- It has a small number of world-wide registers mostly dedicated to a particular purpose

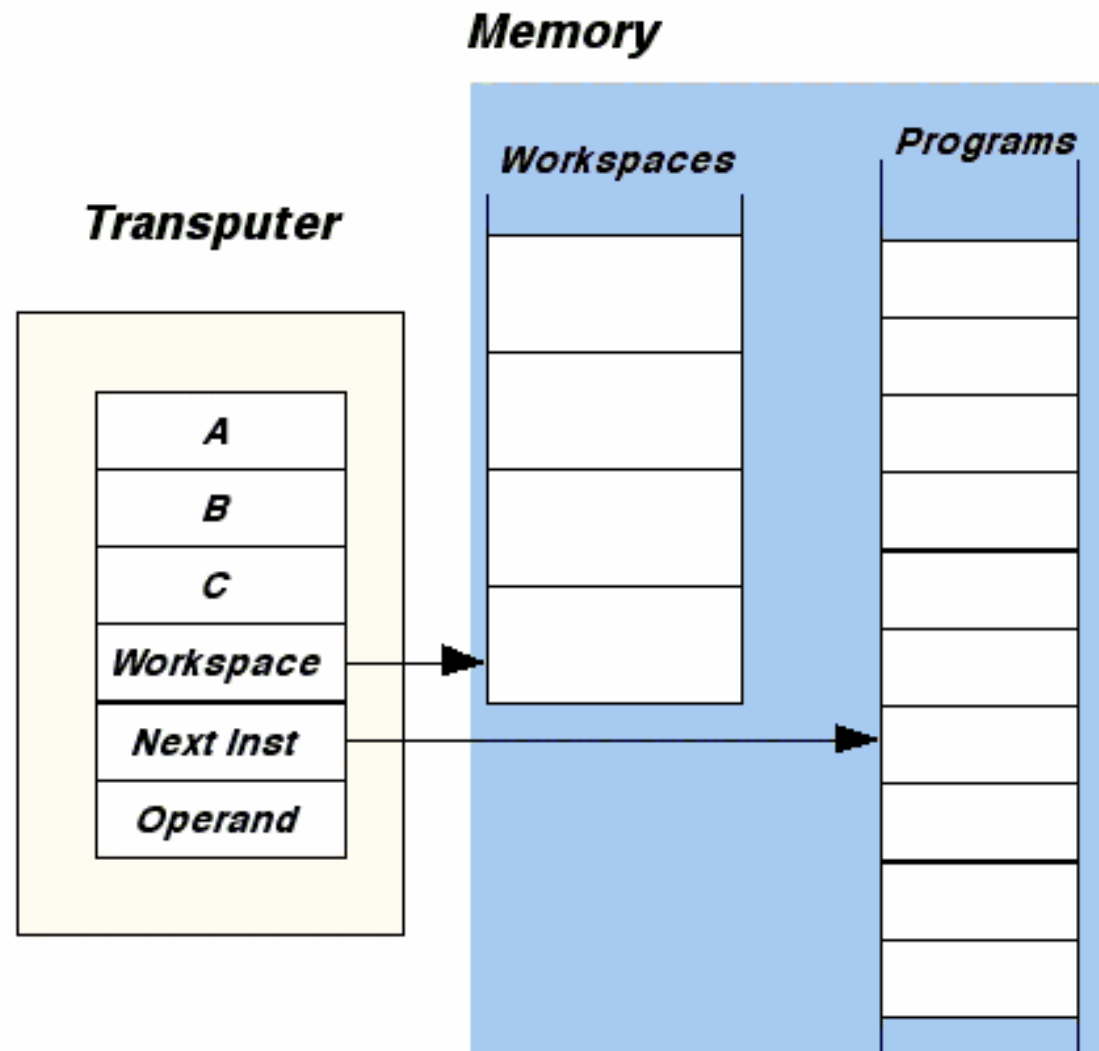
Areg, Breg and Creg: They are used to evaluate expressions and hold instruction operands and results. These are called as evaluation registers and arranged into a stack. Only the Areg is connected to internal buses, so only the Areg can be read or written to.

Iptr, Oreg, Wreg: These are called sequential control registers: Instruction pointer (Iptr), holds the address of the next instruction. Operand register (Oreg), holds the operand for the current instruction. Workspace register (Wreg), holds the workspace pointer (Wptr) which is the address an area of memory called the *local workspace*.

Stack Organization Of Registers



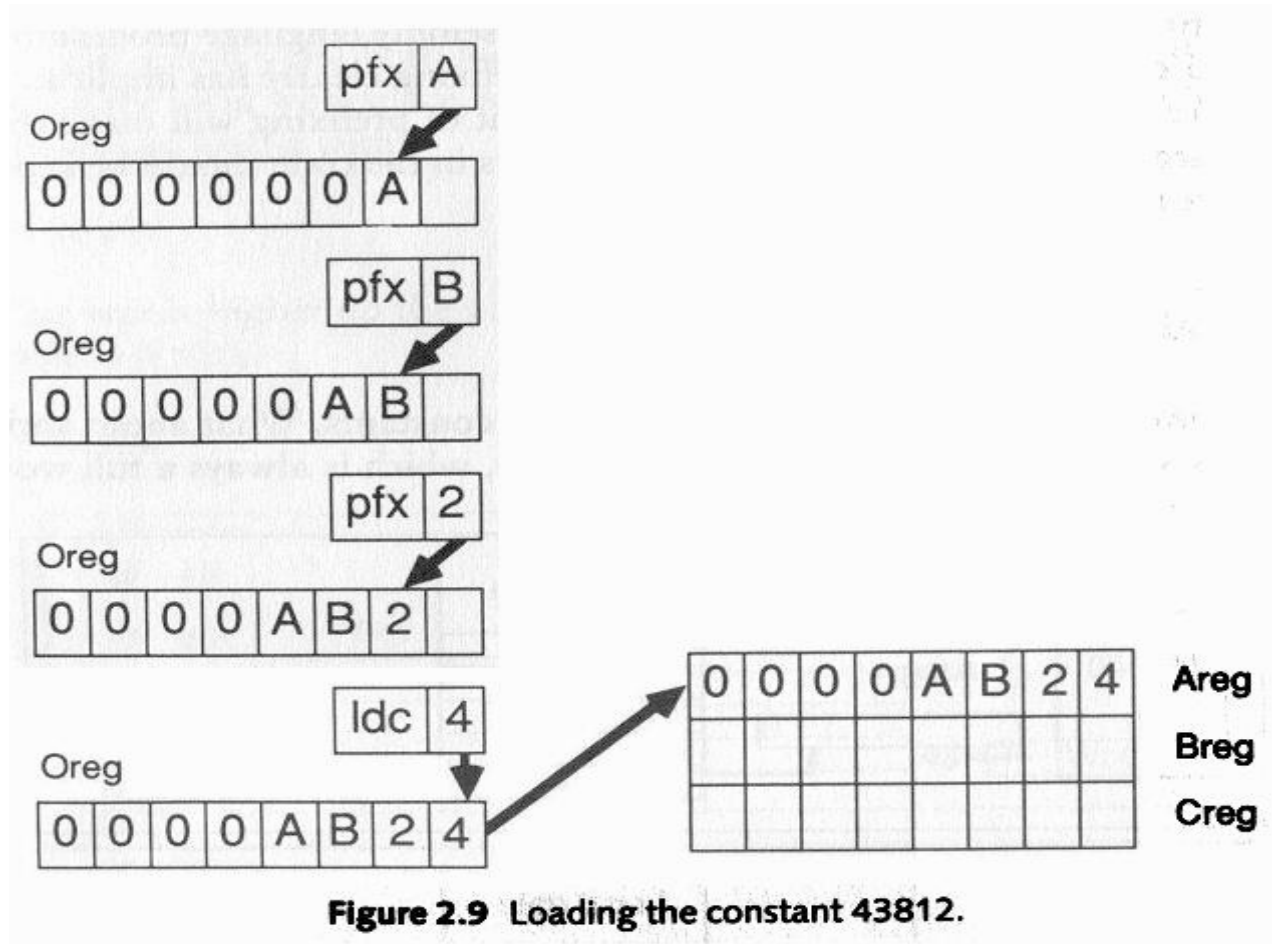
Sequential Control Registers



Processor cont...

- All the transputers have the same instruction format. Each instruction is 8 bits(1 byte) long. The 4 most significant part is gives the opcode and the 4 least significant part is used for data(operand). Execution of every instruction has the same sequence. First the Iptr is incremented. Next, the four data bits are copied into the four least significant bits of the Oreg. Then the function given by the opcode is executed. Finally the Oreg is set back to zero

Prefix



Processor cont...

- Transputer supported two level priority levels, high-priority and low priority
- There is on chip microcode to support, automatically controls timesharing and queues between processes
- At any time a process may be either active or inactive. Transputer has such an architecture that inactive process do not use any processor time

Floating Point Unit

- A separate coprocessor, slave of the CPU, can run at the same time as the CPU but can not run a different parallel process
- There are 53 floating point instructions
- High level programming language to program is strongly advised rather than assembly
- It bases IEEE standards for the floating point format, operations and results

Timers

- The transputer has two timers which can be accessed by the programmer.

High Resolution Timer: increments every five periods of ClockIn (one microsecond resolution with the normal 5 Mhz clock)

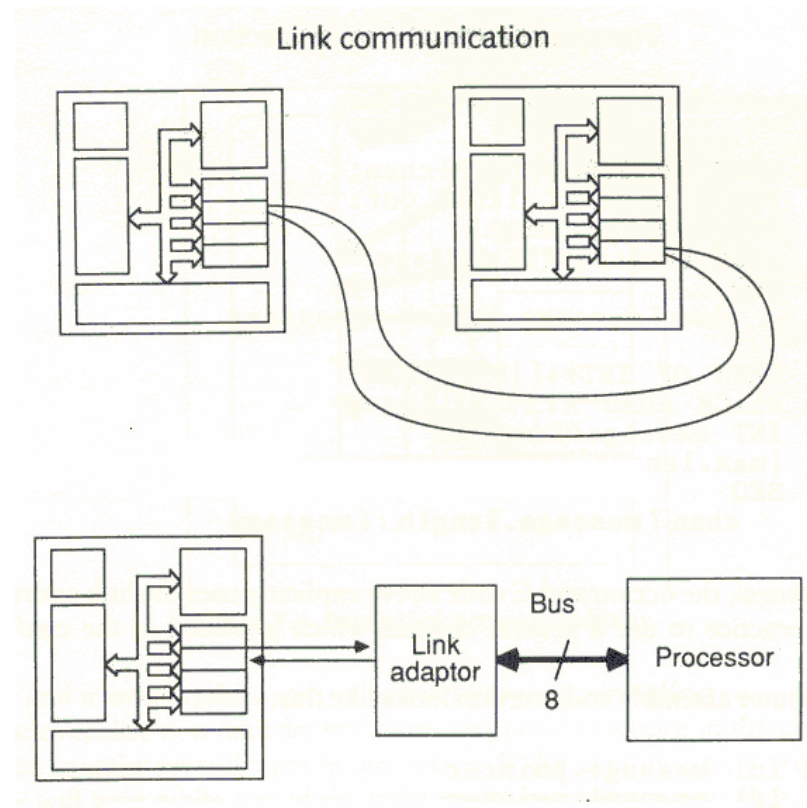
Low Resolution Timer: This is 64 times slower, so increments every 64 microseconds

System Services

- In multi-processor systems it should be convenient to have a hierarchy of control. For example, a host should be able to boot up a network of transputers, detect when an error occurs and debug the network. These are achieved by means of *reset*, *analyze*, and *error* pins that called system service pins

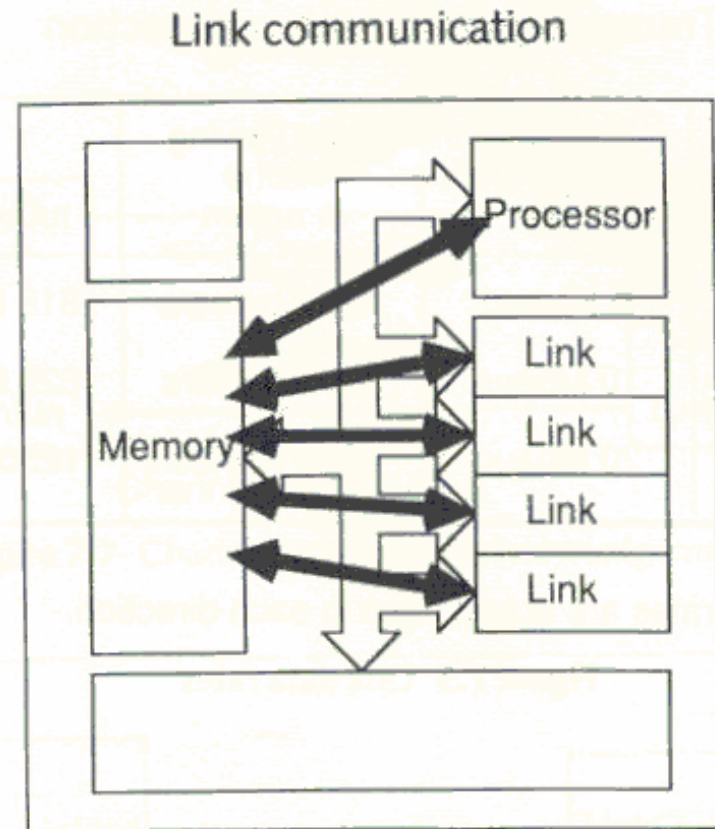
Link Interface

- INMOS link is effectively a serial DMA port.
- It can be used for, interfacing with peripherals using a link adaptor, an ASIC can use a link to read and write directly into a transputer memory at high speed, most common to talk to another processor, usually another transputer.



Link Interface cont..

- The four links and processor have independent access to the memory
- The links designed so that transputers do not need to be synchronized in order to talk each other. However, need to agree nominal bit rate. So this means that transputers in a network may be driven either from a common clock or from separate clocks.



Link Interface cont..

Link Protocols

- Transputers buffer only a single incoming data such that every data packet is “acknowledged”
- First processor executes an input or output instruction and convey it to link. Then starts another parallel process
- When it is output it sends a data packed, and waits till an acknowledge arrives, then next package(chance to wait forever)
- When it is input, checks whether a data has come, if not waits, and sends an acknowledge as soon as get the package
- Synchronized but unbuffered

Link Connection

- Point to point and one way
- Each link carries two channels, one directional, communicates through pins *LinkIn* and *Linkout*. Every data packet is acknowledged so that each channel is synchronized at the programming level.
- Links are designed for short distance communication, max length 30 cm, for more than this distance critical design needed

