Module 3F5: Computer and Network Systems
## Solutions to 2008 Tripos Paper

1. Datapaths and pipelining

(a) A pipelined datapath features extra registers between the principal datapath stages. In each clock cycle, instructions advance through just one stage of the datapath, writing their interim results into the pipeline registers. In this way, several instructions can be in the pipeline at the same time, one at each stage. Pipelining therefore increases instruction throughput.

The term *hazard* is used to describe dependencies between instructions which disrupt the operation of a pipelined datapath. *Data hazards* occur when an instruction requires data before a previous instruction has written it to the register file. *Branch hazards* occur when the address of the next instruction is required (for instruction fetching) before an earlier conditional branch instruction has been evaluated.   [20%]

(b) Data forwarding avoids the need for values to be available in the register file: instead, they are extracted from the downstream pipeline registers. Forwarding can be implemented by adding multiplexors to the inputs of the ALU and supplying suitable control lines, as shown in the diagram.

The forwarding unit compares the registers which have just been read ("read reg 1" and "read reg 2") with any registers about to be written by the two downstream instructions. If they are the same, it sets MUXA and/or MUXB to ignore the value read from the register file and instead use the appropriate forwarded value.
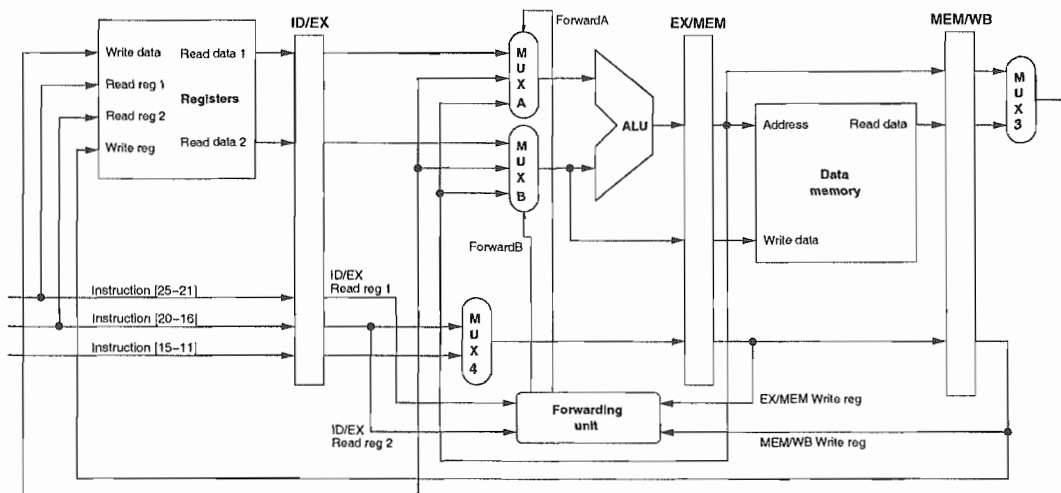
This sort of data forwarding will not work when the instruction following a `lw` requires the result of the load. The data is read from memory at clock cycle 4, and required at the ALU for the following instruction (assuming it is not a `sw`) at the same time. This sort of data hazard must be resolved by hardware stalling or specifying *delayed loads* in the instruction set architecture. [40%]

(c) (i) Without pipelining, each instruction takes $(2 + 3k)T$, so n instructions take $n(2 + 3k)T$. (ii) With pipelining, all the stages need to run at the speed of the slowest stage, so we need to distinguish between the cases $k \leq 1$ and $k > 1$. For $k \leq 1$, the slowest stage requires T, so all stages must run at this speed. The first instruction takes $5T$ to execute, with each subsequent adding a further T. The time to execute n instructions is therefore $5T + (n-1)T = (n+4)T$. For $k > 1$, the slowest stage requires kT, so all stages must run at this speed. The first instruction takes $5kT$ to execute, with each subsequent adding a further kT. The time to execute n instructions is therefore $5kT + (n - 1)kT = (n + 4)kT$.

The speed-up is the time without pipelining divided by the time with pipelining:

$$\text{speed-up} = \begin{cases} \dfrac{n(2 + 3k)T}{(n + 4)T} \rightarrow 2 + 3k \text{ as } n \rightarrow \infty, & k \leq 1 \\[4mm] \dfrac{n(2 + 3k)T}{(n + 4)kT} \rightarrow \dfrac{(2 + 3k)}{k} \text{ as } n \rightarrow \infty, & k > 1 \end{cases}$$
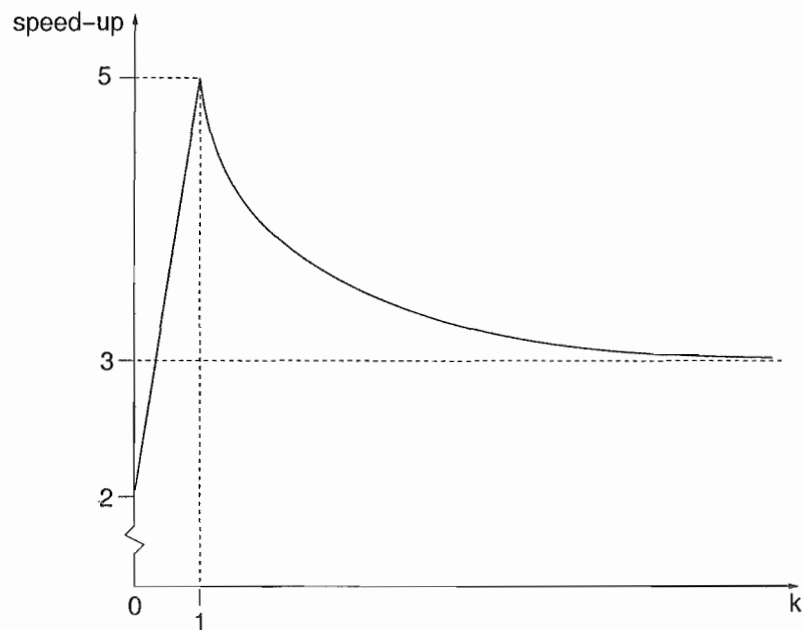
As expected, the speed-up peaks when the pipeline is balanced at $k = 1$. [40%]

2. Instruction set architectures and ALUs

(a) An accumulator instruction set architecture uses a special register, the accumulator, as an operand in many instructions. For example, consider adding two numbers at memory locations MEM1 and MEM2, storing the result in MEM3:

```
ldaa MEM1    # accumulator loaded with number at MEM1
adda MEM2    # add number at MEM2, result put back in accumulator
staa MEM3    # accumulator stored at MEM3
```

Note how the arithmetic instruction adda gets one of its operands from memory. In contrast, a general purpose register, load-store architecture does not have a special purpose accumulator but a large number of general purpose registers. Arithmetic operations can only operate on numbers in registers: only load and store instructions can access memory. The addition would look like this:

```
lw $8,MEM1($0)  # register $8 gets number at MEM1
lw $9,MEM2($0)  # register $9 gets number at MEM2
add $10,$8,$9   # register $10 gets register $8 plus register $9
sw $10,MEM3($0) # register $10 stored at MEM3
```

Note how there are no implicit operands (compare with the adda instruction for the accumulator instruction set architecture).   [20%]

(b) The distinguishing features of RISC architectures are:

- Fixed length instructions
- Just a few instruction formats
- General purpose register, load-store instruction set architecture
- Limited operations
- Simple addressing modes
- Complete one instruction per datapath cycle (with pipelining)
- No microcode to control the datapath   [20%]

(c) Amdahl's Law says that it is of paramount importance to optimise the speed of those components that are most frequently used in a computer system ("make the common case fast"). For most instruction set architectures, ALUs are used at least once in the execution of just about every instruction. Faster ALUs would therefore have a significant effect on the speed of the overall system.

In a pipelined datapath, all stages must run to the same clock, the speed of which is dictated by the slowest stage. It is conceivable that a stage (eg. instruction decode and register fetch) does not use an ALU and happens to be the rate-limiting stage,

in which case further optimisation of the ALU would be pointless, since we would not be able to increase the clock speed. However, we do need to optimise the ALU at least to the point where it is not contributing to a rate-limiting pipeline stage.　　　[20%]

(d) Carry lookahead can be used to determine the carry inputs to each full adder without using ripple carry. For each bit i of the adder, we define two signals, generate gi and propagate pi. Bit i generates a carry if the two bits it's adding are both 1, and propagates a carry if either of the two bits it's adding are 1:

$$gi = ai.bi \qquad\qquad pi = ai + bi$$

c1, the carry into bit 1, will be 1 if either bit 0 generates a carry or c0 is 1 and bit 0 propagates a carry:

$$c1 = g0 + p0.c0$$

Likewise for c2 and c3:

$$c2 = g1 + p1.g0 + p1.p0.c0$$
$$c3 = g2 + p2.g1 + p2.p1.g0 + p2.p1.p0.c0$$

These expressions show how the carry-in signals can be obtained without waiting for them to ripple through a 4-bit adder.　　　[20%]

(e) For an n-bit ripple-carry adder, the asymptotic time and space requirements are both clearly $O(n)$.

For an n-bit carry-lookahead adder, if we use carry-lookahead to predict m carry-in signals at each level of the hierarchy, then we will need $\log_m n$ levels, and gates with a fan-in of m. Since each level takes a constant amount of time to operate, the asymptotic time requirement is $O(\log_m n)$. In any reasonably straightforward silicon layout, the asymptotic space requirement is $O(n \log_m n)$.

Note that we should take asymptotic behaviours with a pinch of salt. They provide a useful design criterion but should not be relied on too much, since asymptotic behaviour is only really important when n becomes large. For smaller n, the constant of proportionality might make an $O(n)$ adder faster than an $O(\log_m n)$ one. Real adders in real CPUs deal with 64-bit numbers at most.

The choice of the optimal value for m (and hence the number of levels) will depend on the particular implementation technology. The asymptotic complexities suggest that the larger the value of m, the fewer the number of levels and hence the faster the adder. However, this will require higher fan-ins to gates, which may be infeasible in certain technologies and will certainly result in longer propagation delays and hence slower operation.　　　[20%]

Q3 Crib (longer than expected from candidates)
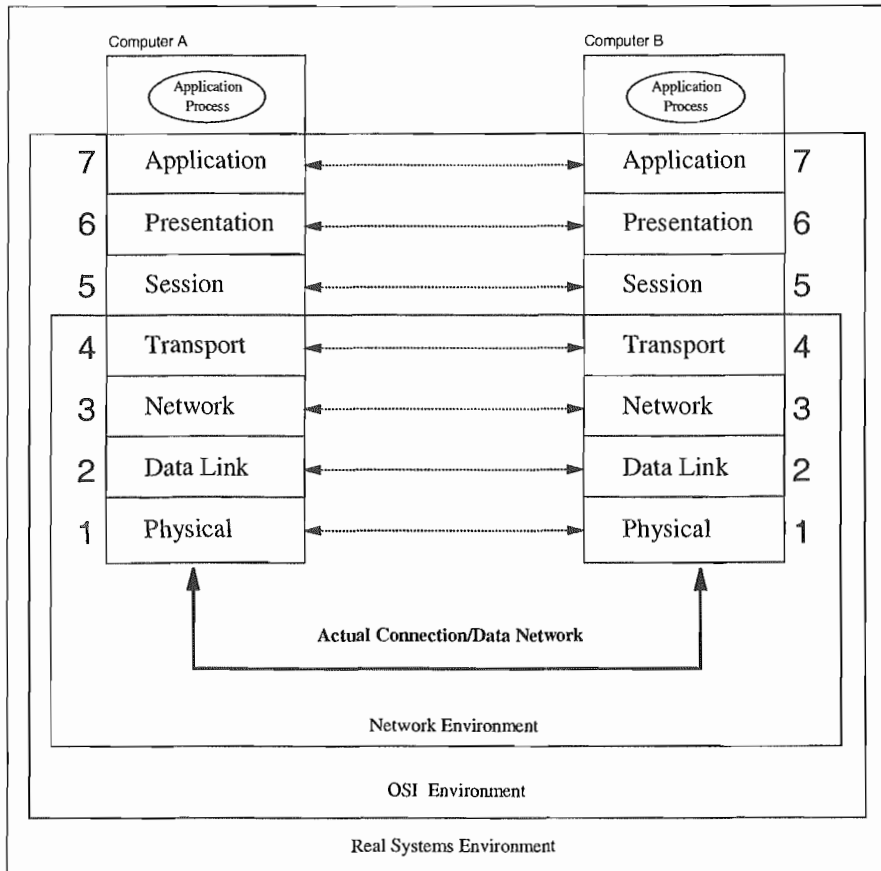
(1) The network environment. The protocols and standards relating to the different types of underlying data communications networks.
(2) The OSI environment. Adds additional application oriented protocols and standards to allow end systems to communicate with one another in an open way.
(3) The real systems environment. Builds on the OSI environment and is the manufacturers own proprietary software and services.

Both the network dependent and application oriented (network independent) components of the OSI model are in turn implemented in the form of a number of protocol layers. The boundaries and processes for each layer have been selected based on experience gained from other standards in the past.

b)
i)   Manchester coding – Physical layer – purely a bit encoding function
ii)   Frame check sequence – Data-link layer – process for the reliable transmission of data
iii)   Remote login – Session layer – setting up and pulling down a comms session
iv)   JPEG to TIFF conversion – Presentation layer – conversion to find a common graphics format
v)   MAC address lookup – data-link layer as it is a hardware address, however the lookup process is really a network layer process.

vi)   Address resolution protocol – network layer, although the address is often a physical one which is a data-link entity.
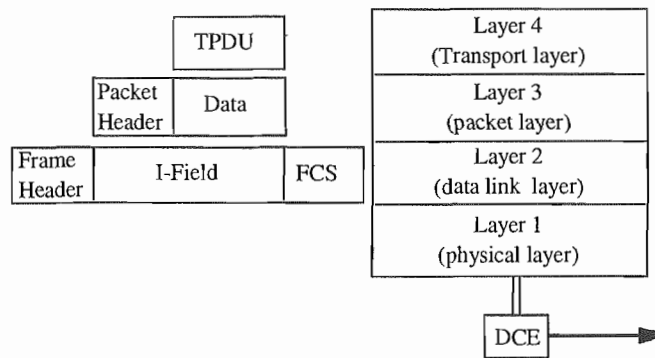
c)
The OSI model is broken into 7 layers, each with a defined purpose and protocol.
• Each layer is a service user to the layer below and a service provider for the layer above.
• Each layer is specified independently of the other layers;
• Each layer has a defined interface with the layer above and below.
The lowest three layers (1-3) are network dependent and are concerned with the protocols associated with the data communication network being used to link the two computers. In contrast, the upper three layers (5-7) are application oriented and are concerned with the protocols that allow the two end user application processes to interact with each other.

As far as users are concerned, communication appears to take place across each layer. Each data exchange passes down to the bottom layer (the physical layer) at the sending terminal, crosses the network to the receiving terminal and then passes up again. Data communication between layers is done through the addition and reading of headers on the data

TPDU		Layer 4 (Transport layer)

| Packet Header | Data | Layer 3 (packet layer) |

| Frame Header | I-Field | FCS | Layer 2 (data link layer) |

| | Layer 1 (physical layer) |

DCE →

d) In ATM, the cell is limited to 53bytes, a 48 byte payload and a 5 byte header. ATM is defined at a transmission rate of 155.52Mbits/sec to be compliant with SDH and SONET. This very short packet length and simple header make it difficult to map onto the OSI model. There is a lot of fragmentation of data and also whole cells can be used as a single header to set up connection oriented services. The mapping of ATM onto the OSI model is also difficult, as there are layer splits. This is because a lot ATM services are physical media depedant.

OSI	ATM	ATM Sublayer	
3/4	AAL	SAR	CS
2/3	ATM	ATM	
2	Physical	TC	
1		PMD	

The ATM layer deals with cells and cell transport. It defines the header layout and functionality. The ATM adaptation layer (AAL) takes packets that are larger than cells and segments them. The AAL is split into the segmentation and reassembly (SAR) sublayer and the convergence sublayer (CS). The SAR/CS split allows different network services like file transfer and video on demand that have different error correction requirements and quality of service. The physical layer has more functionality than in the OSI model and is split into the physical medium dependent (PMD) sublayer and the transmission convergence (TC).

Q4 Crib (longer than expected from candidates)

Coaxial cable - It is possible to use Maxwell's equations to calculate the electric and magnetic fields within the coaxial cable to give the attenuation in the cable due to the inner conductor and the solid insulator which shows that coaxial cables have:
- A high frequency cut-off
- A high immunity to external interference and crosstalk.

Coaxial cable is a good transmission medium for high data rates over relatively short distances, however it is rather expensive and is not ideally suited to long distance transmission and has been largely replaced by optical fibre. It was the basis for Ethernet and gave us the 1500byte packet limit still used today

Optical fibre - Singlemode (SM) fibre is extensively used for long haul telecommunications. Almost all of the digital telephone networks in the UK uses silica based singlemode optical fibre due to its very low attenuation and high data bandwidth. It can be expanded greatly using WDM.

Multimode fibre is used in local area networks (LANs) in a few special cases. For instance, a length of multimode fibre is used to connect together LANs in two separate buildings. Use is restricted due to modal dispersion leading to inter-symbol interference over long distances.

Multimode fibre is still difficult to connect and terminate and is relatively expensive. New networks and LANs are being designed with plastic optical fibres which is the basis for home network standards, including Firewire and IEEE 1392.

Twisted pair cable - Normally used with a telephone connection. The bandwidth of this cabling is severely limited by its length and the way in which it is used within the telecoms networks. Twisted pair cables are classified in two types
- Unshielded twisted pair (UTP)
- Shielded twisted pair (STP).

Initially TP gave us the modem to allow higher data rates within the existing telecoms networks however recently we have seen the opening of the telecoms sector allowing the creation of digital subscriber lines (DSL) which avoid the telecoms networks for computer data and generate much higher data rates within a certain distance from the exchange.

Wireless – wireless is perhaps the perfect physical media as it has totally unrestricted physicality, however this is also its limitation as it is shared by all giving limited bandwidth, interference and security problems. Wireless has lead to several great

protocols most of which originated with Ethernet and use some form of collision based access mechanism.

b) Aside from being small and fragile, optical fibre is the perfect transmission medium for a telecoms network offering an attenuation of less than 1dB/km over a wavelength range of 700nm.

This means that we could modulate an optical carrier at data rates beyond 100Tbits/sec (100, 000Gbits/sec) and still have attenuation less than 1dB/km. Optical fibres are virtually immune to external interference and do not produce any radiation that could cause interference.
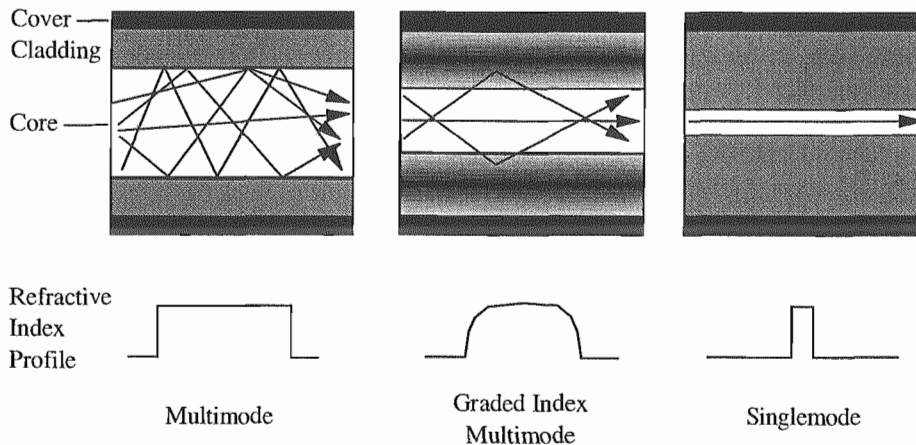
Multimode   Graded Index Multimode   Singlemode

Singlemode (SM) fibre is extensively used for long haul telecommunications. Almost all of the digital telephone networks in the UK uses silica based singlemode optical fibre. Silica SM fibre has very low attenuation per unit length, especially at the two main comms wavelengths, 1310 and 1550nm. Singlemode optical fibre takes advantage of the high bandwidth of the optical fibre by using multiple wavelengths down the fibre and wavelength division multiplexing (WDM).

It also allows the use of all optical (Erbium doped) fibre amplifiers instead of regenerators. The modulation rate is limited by frequency chirp in the laser and dispersion in the optical fibre leading to inter-symbol interference. Singlemode fibre is not used over short distances in LANs, due to the expense of the optical modulators (lasers) and amplifiers and the expense and difficulty of connecting and terminating the fibre.

Multimode fibre is used in local area networks (LANs) in a few special cases (see FDDI later). Use is restricted in telecoms networks due to modal dispersion leading to inter-symbol interference over long distances. This dispersion can be greatly reduced by using graded index multimode fibre to reduce the number of modes in the fibre. MMF has become increasingly popular in MANs as it offers high data rates at relatively low cost over metropolitan distances.

c) SM optical has a enormous bandwidth, hence it is ideal for SDH where the process of multiplexing allows this bandwidth to be utilised. STM is defined to 40Gbit/sec and with WDM, terabits of data can be sent over a single SM fibre. The timeslot mechanism

of SDH means that it is easy to access data at any data rate within the Mux, creating a very low latency (as required by voice) telecoms network.

Below is an example a tier 1 ring structure. This is capable of carrying 96 2Mbit/sec channels. The ring head mux can groom the 2Mbit/sec channels to provide connection to the higher tiers and sort out traffic within the ring. The ring is self healing for fault tolerance using a pair of optical fibres in East/West mode .
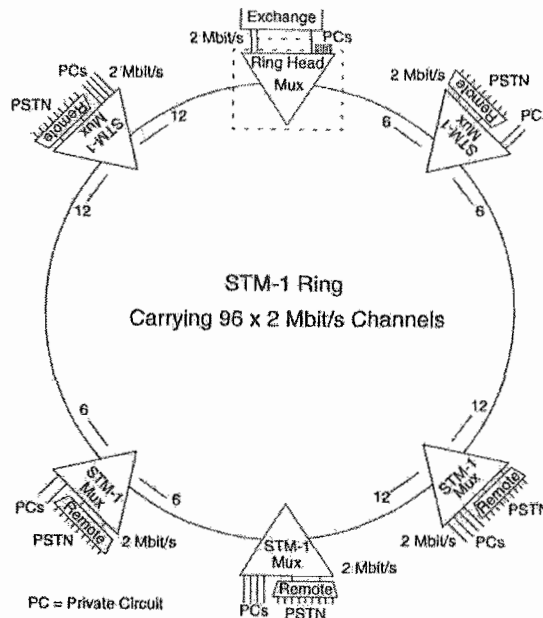
The synchronous mux (basic function only), ITU-T defined.
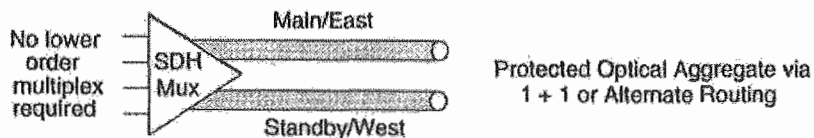
Allows optical integration at the STM-N level. Optical interfaces are often repeated for redundancy as main/standby or East/West pairs. Often in a ring topology. Must offer digital mux to STM-1 with flexible inputs.

Assessors' remarks for question 1: A popular question testing candidates' understanding of pipelined datapaths. Most produced excellent textbook answers to (a), though a common mistake in (b) was to sketch instruction flow schematics rather than the datapath layout showing the forwarding unit's connections. The speedup calculations in (c) were performed flawlessly by several candidates, though many others analysed only the $k \geq 1$ case.

Assessors' remarks for question 2: This essay-style question tested the candidates' understanding of instruction set architectures and carry-lookahead adders. Most candidates knew the basics, especially when it came to the detail of the carry-lookahead adder, though there was some confusion about the differences between the various instruction set architecture types. As usual with essay questions, the best answers were distinguished by sound editorial judgement: they highlighted the key points and, just as importantly, omitted irrelevancies and trivia.

Assessors' remarks for question 3: Overall well answered with most candidates being able to recite the OSI model accurately. Quite a few missed the real systems environment and its role in the model. The layer allocations were well discussed but many candidates did not appear to know about Manchester encoding. The final section was more speculative with few candidates getting the role of the OSI model in the X.25 protocol stack.

Assessors' remarks for question 4: Mostly book work and well answered. Quite a few candidates could only come up with three physical layer systems and tried to include two types of optical fibre as separate layers. The section on SMF versus MMF was well answered but most lost the significance of silica and WDM in the popularity of fibres. The final section was quite varied with a few candidates producing the correct diagram, but most missing the low loss aspects of SMF.

<div align="right">

Andrew Gee & Tim Wilkinson
May 2008

</div>