AMULET1 -
An Asynchronous ARM Processor

Steve Furber

Department of Computer Science,
University of Manchester,
Oxford Road,
Manchester M13 9PL, England.
sfurber@cs.man.ac.uk

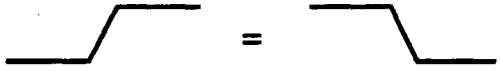
Motivation and background

- Low power applications are becoming more important
- Fast synchronous chips use a lot of power
- Asynchronous units dissipate only when doing useful work

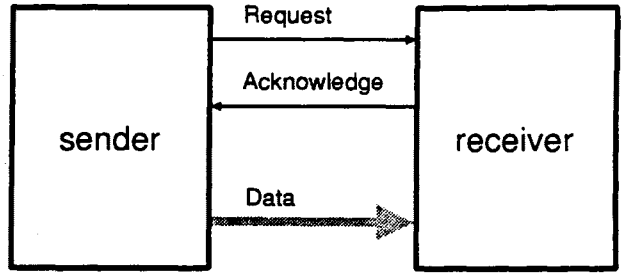
Therefore..

- Investigate the benefits of asynchronous logic by developing an asynchronous ARM chip
- Funded by ESPRIT in the OMI-MAP project

Micropipelines

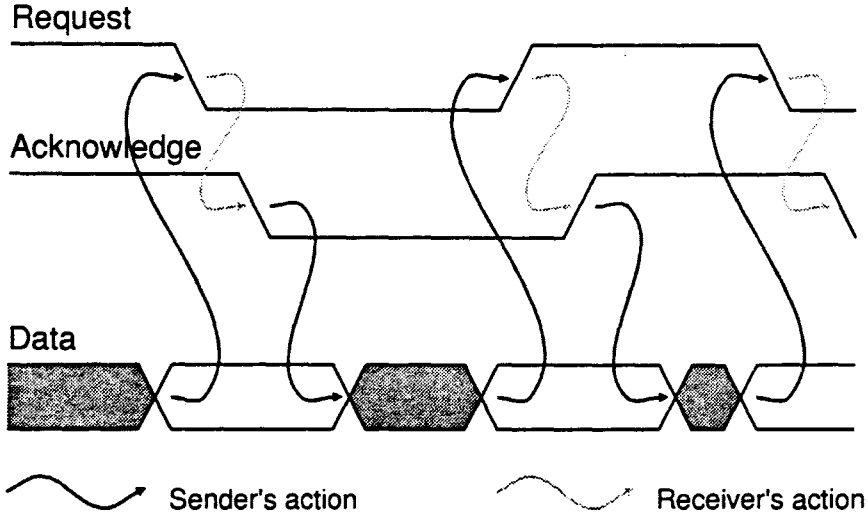
Transition signalling: 

Bundled data:

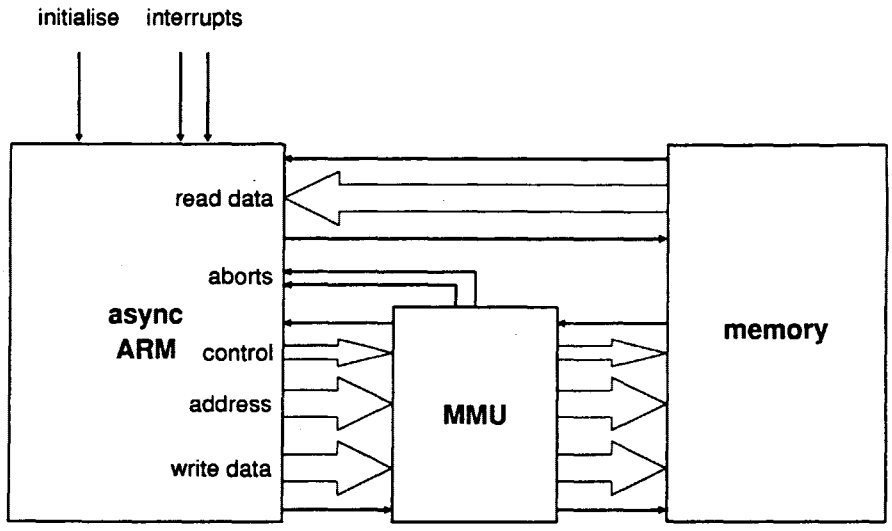


(I E Sutherland, Comm ACM, June 1989)

2-phase bundled data protocol

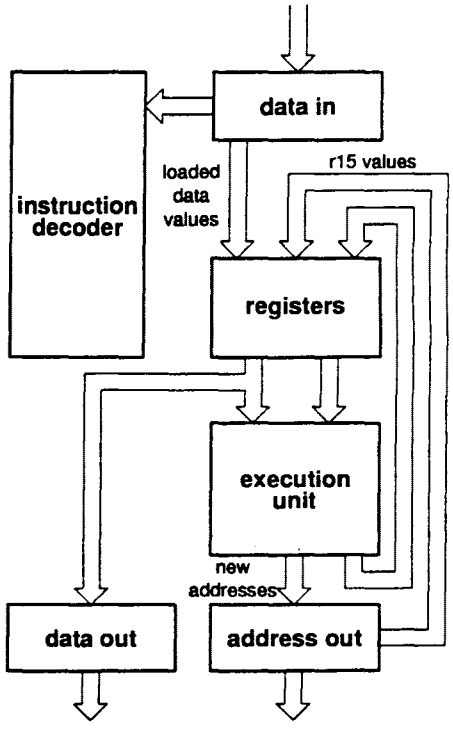


Top-level processor interface



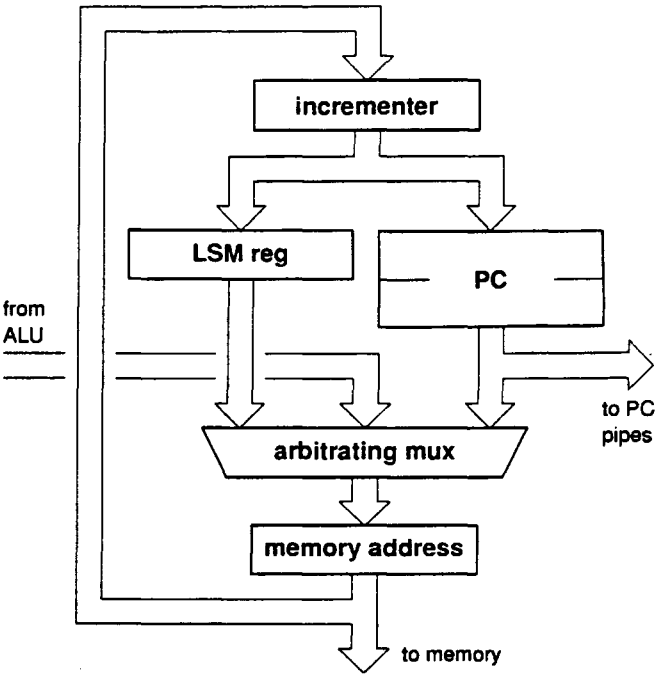
CPU organization

- independent units
- operate concurrently
- synchronize only to exchange data

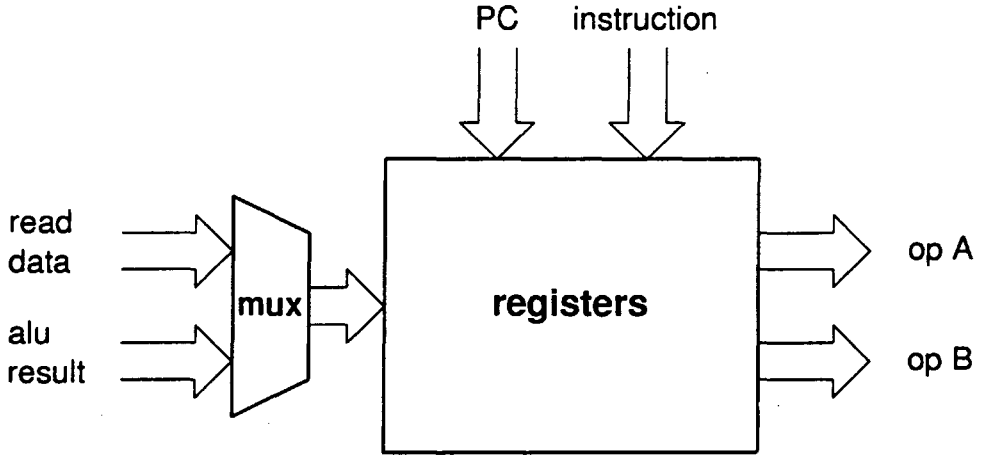


Address unit

- autonomous incrementing PC loop
- ALU breaks into loop
- load/store multiple also uses loop

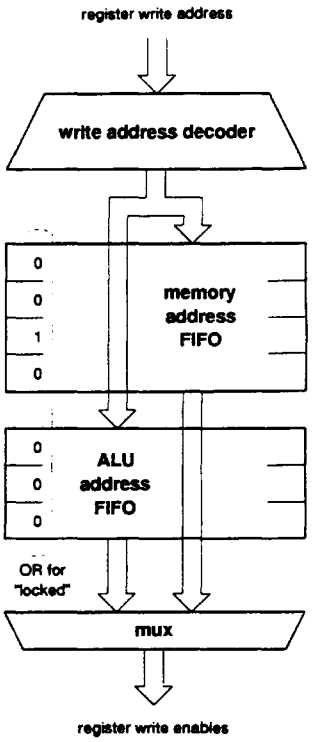


Register bank



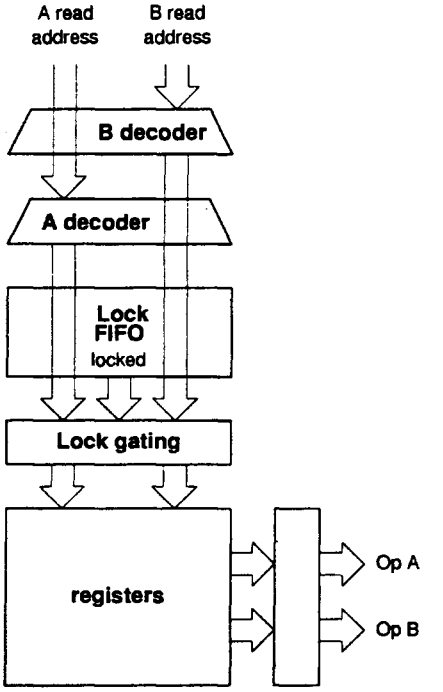
Locking mechanism

- FIFO stores decoded write addresses
- a '1' in a column signifies a pending write to a register
- logical OR of a column is used to lock register
- separate internal and external FIFOs remove some memory waits



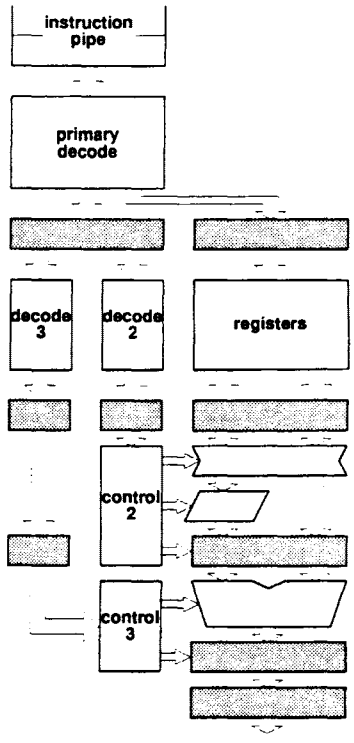
Read operation

- lock signal is gated with read word line
- read is stalled until lock clears
- self-timed data availability signal waits for read data

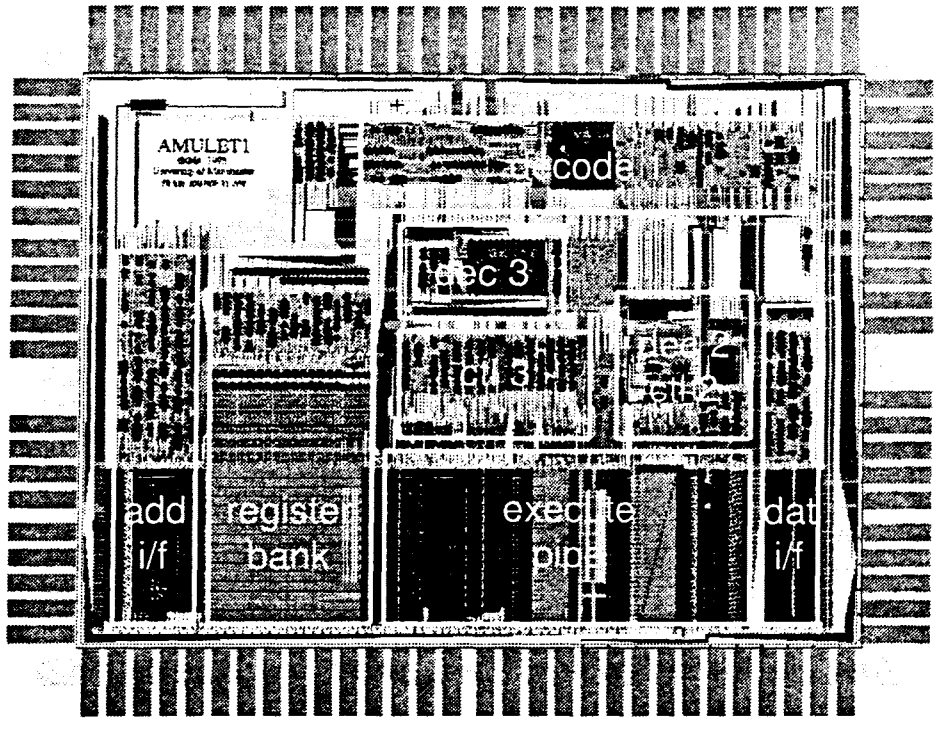


Execute pipe control

- parallel data and control pipelines
- synchronize only where they converge
- pipelines matched for optimum throughput



AMULET group
University of Manchester



AMULET group
University of Manchester

Results

Compared with the synchronous ARM6 on the same technology:

- chip area 2x, mainly due to more sophisticated organization
- similar performance, with very conservative engineering
- similar power consumption
- similar design effort

Conclusions

- Complex asynchronous designs are feasible - this design supports exact exceptions and backwards instruction set compatibility.
- The area cost of 'bundled data' asynchronous designs is probably around 10-20% over clocked styles. (Other asynchronous styles have an overhead of up to 100%.)
- The performance/power advantage still has to be demonstrated, though we are within a factor of 2.