UK Asynchronous Forum, Edinburgh

Multi-View Design of Asynchronous Micropipeline Systems using Rainbow

Howard Barringer Donal Fellows Graham Gough
Pete Jinks Brian Marsden Alan Williams

Department of Computer Science,
University of Manchester

December 1996

Supported via EPSRC Grant GR/K42073

Contents

- Background
- Rainbow Design Framework
- Example: Simple Processor
- Formal Semantics
- Simulation
- Summary

Background

Asynchronous hardware design:

- offers: increased speed, reduced power, easier design (compositional)
- example: AMULET microprocessor (Manchester)

Need:

- fast prototyping
- simulation
- early design analysis

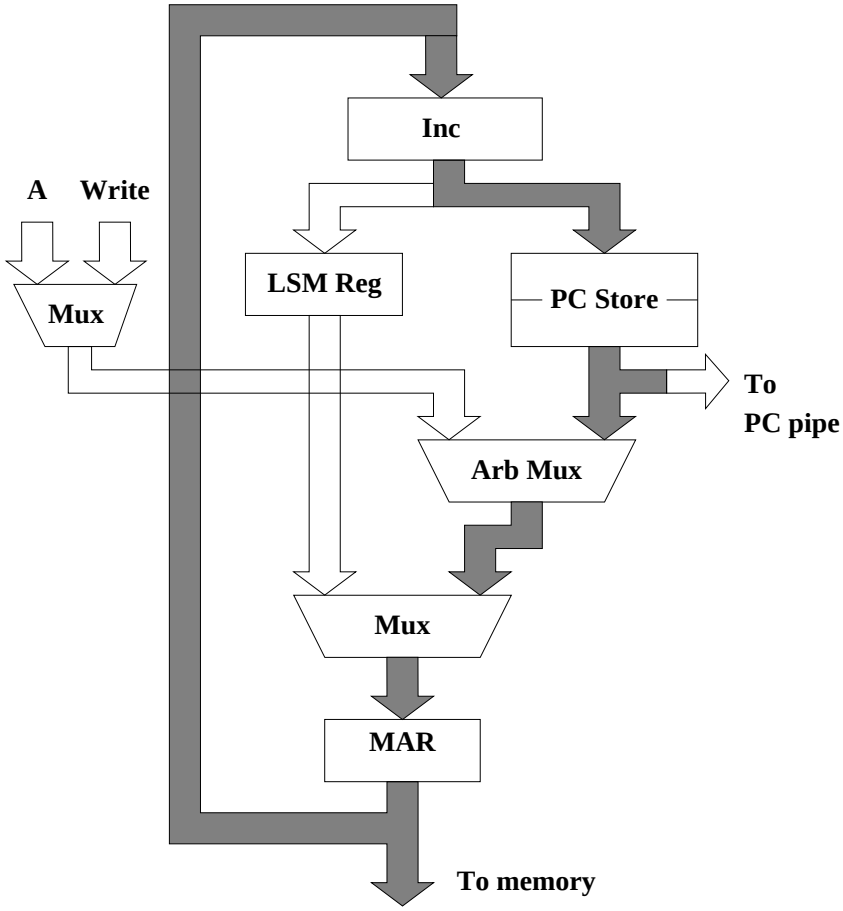
But...problems facing asynchronous design:

- lack of specialised design tools
- design analysis

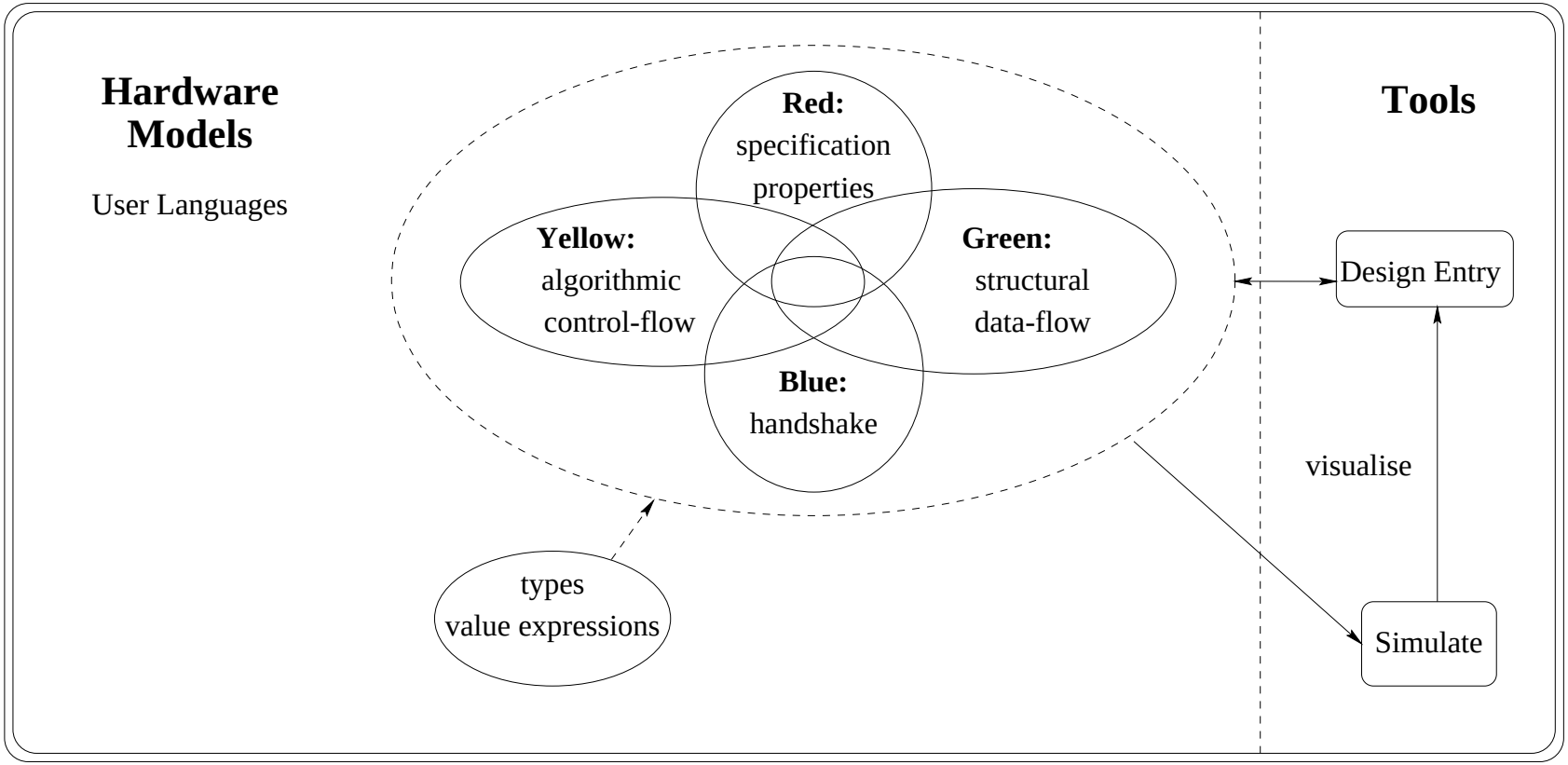
Require:

- design methodology: Micropipelines
- analysis tools
  - deadlock detection
  - high level power, speed estimation
  - functional verification

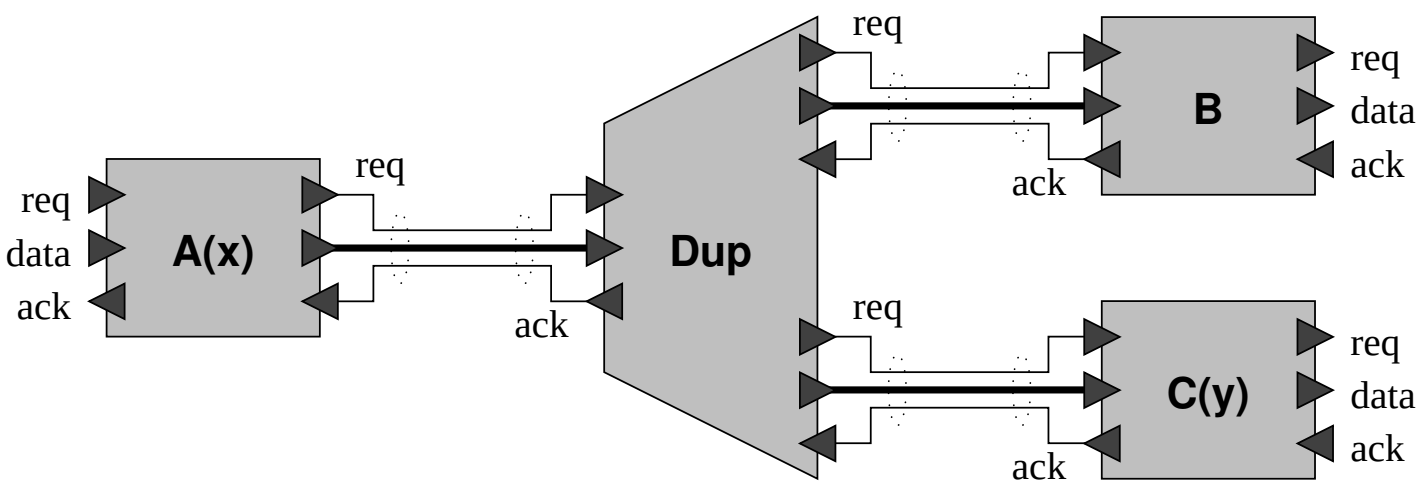
...and a suitable design representation

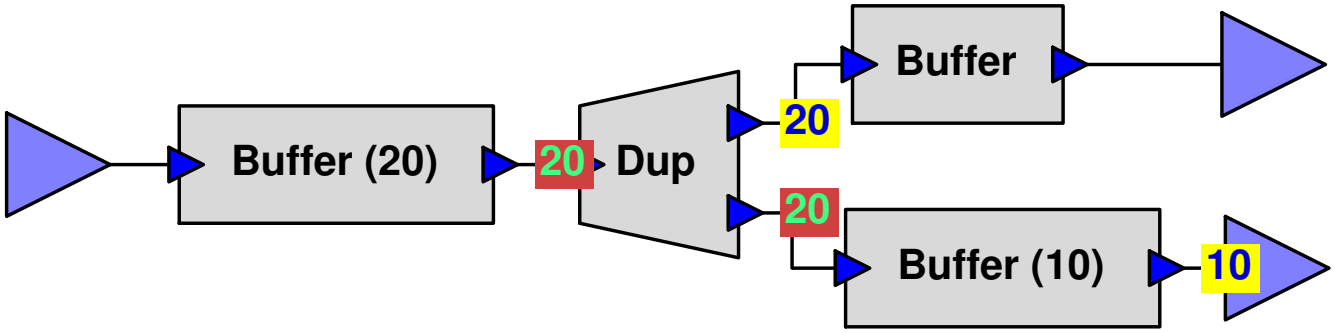

Inc
A
Write
LSM Reg
PC Store
Mux
To
PC pipe
Arb Mux
Mux
MAR
To memory

Rainbow Design Framework

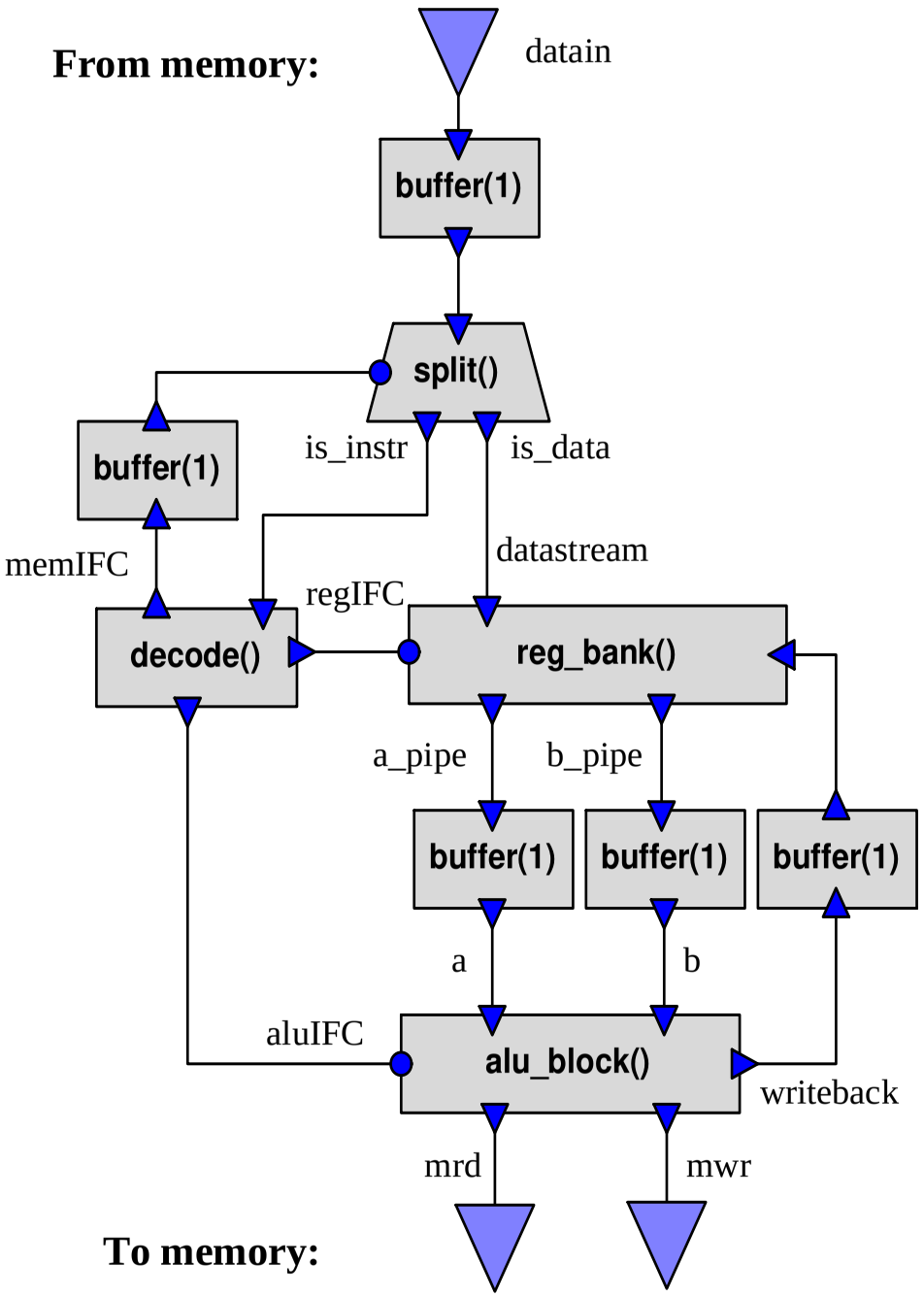


- Application-specific
- User-oriented language design (AMULET)
- Abstract representation of bundled data
- Mixed-view modelling
- Hierarchical design


req
data
ack
A(x)
req
ack
Dup
req
ack
B
req
data
ack
req
ack
C(y)
req
data
ack


Buffer (20)
20
Dup
20
Buffer
20
Buffer (10)
10

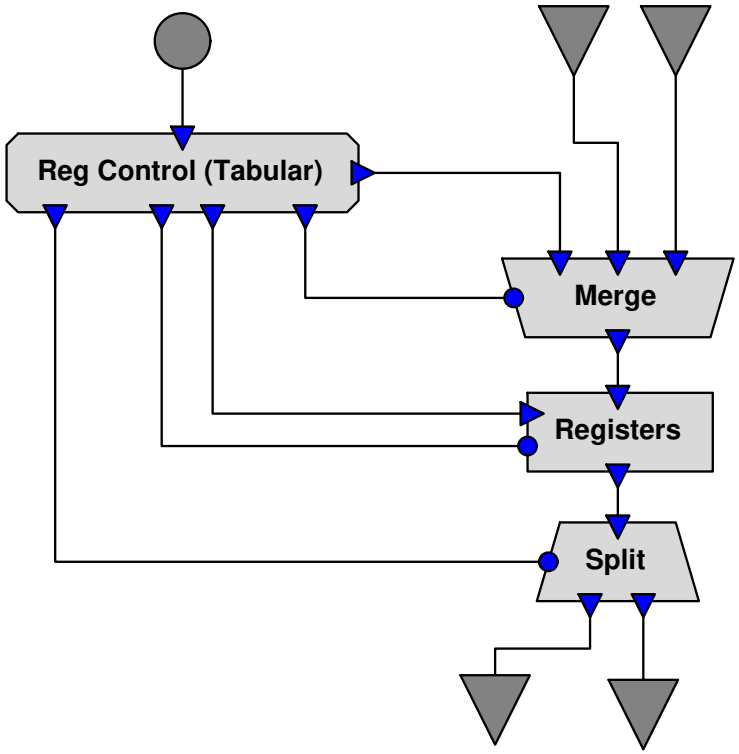
Example: Simple Processor



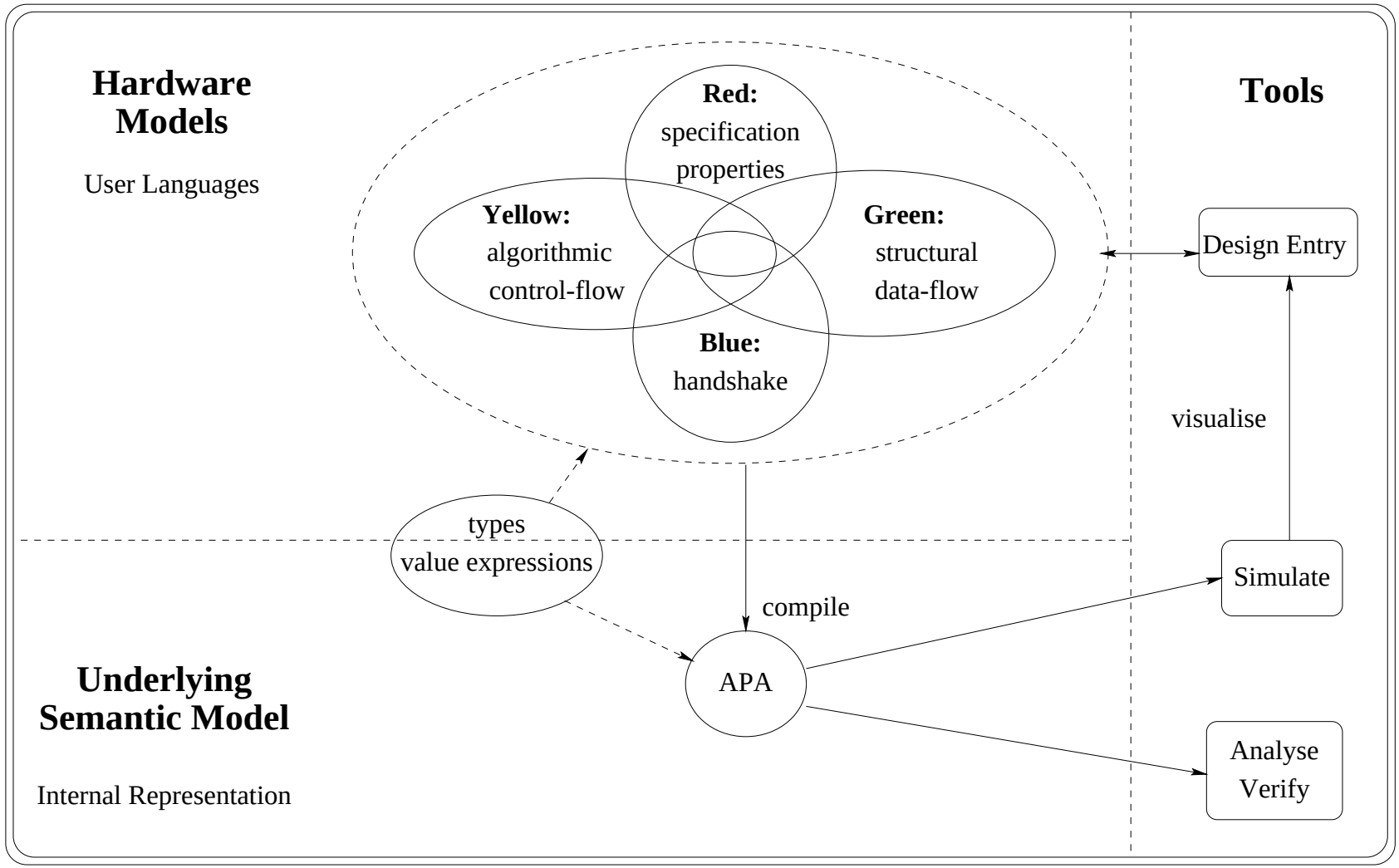
Green Table — ALU:

```
table alu_block (input  aluIFC: aluctrl;
                 input  a,b: val;
                 output mrd: memread;
                 output mwr: memwrite;
                 output writeback: val) = {
    FETCHINC, a,   -  =>  a,    -,      a+1
 or FROMMEM,  a,   b  =>  a+b,  -,      -
 or TOMEM,    a,   b  =>  -,    (a,b),  -

 or ADD,      a,   b  =>  -,    -,      a+b
 or SUB,      a,   b  =>  -,    -,      a-b
 or AND,      a,   b  =>  -,    -,      a/\b
 or CMP,      a,   b  =>  -,    -,      a>b

 or MOV,      a,   -  =>  -,    -,      a
 or NOT,      a,   -  =>  -,    -,      not a
}.
```

- Yellow Decoder: `fetch(); execute(); store()`
- Green sub-network: register bank (textual or schematic)



Formal Semantics



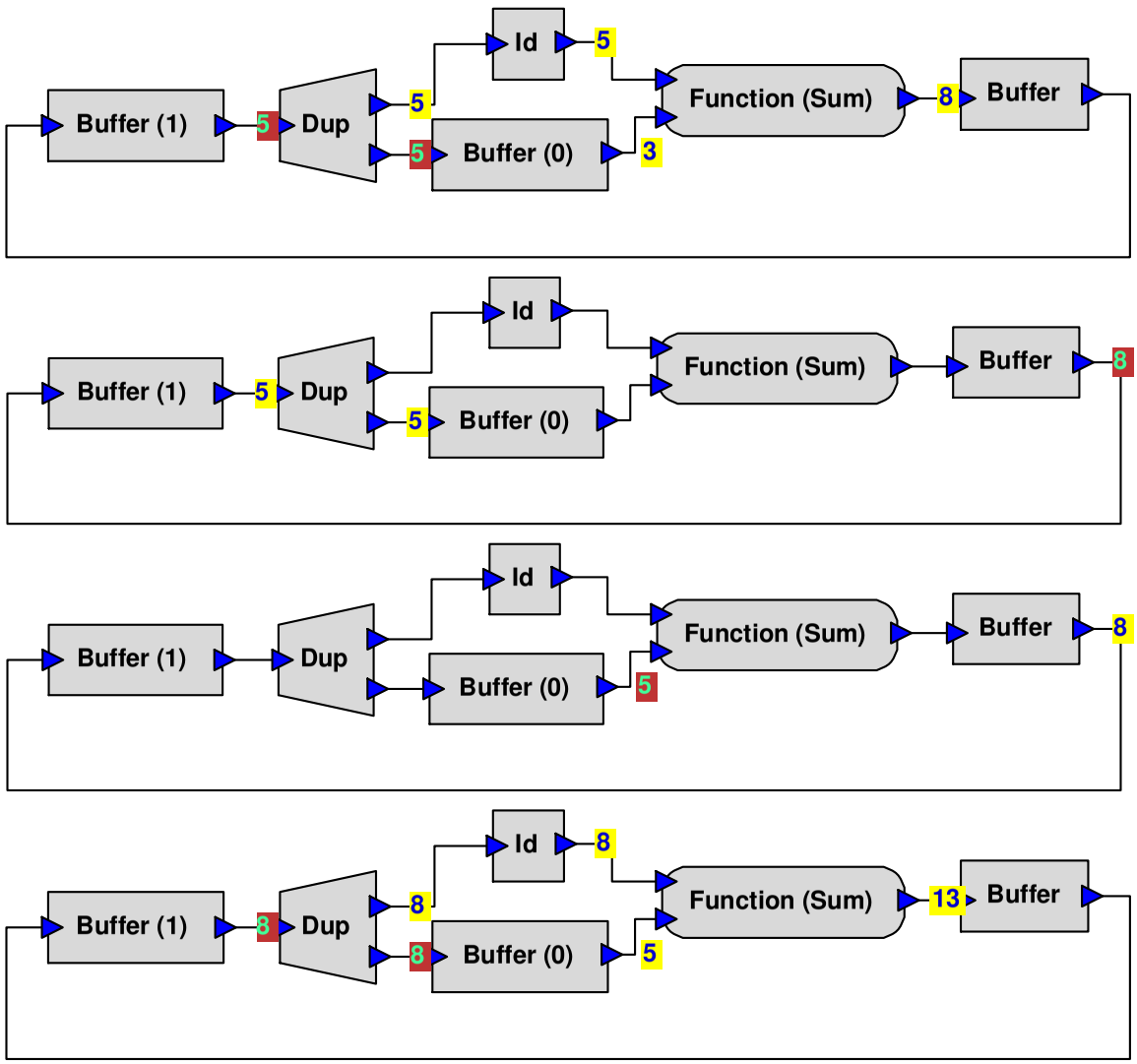
APA

- complex actions
- bundled dataflow blocks as basic constructs
- Ada-like rendezvous communication primitive
- transition rules define APA semantics

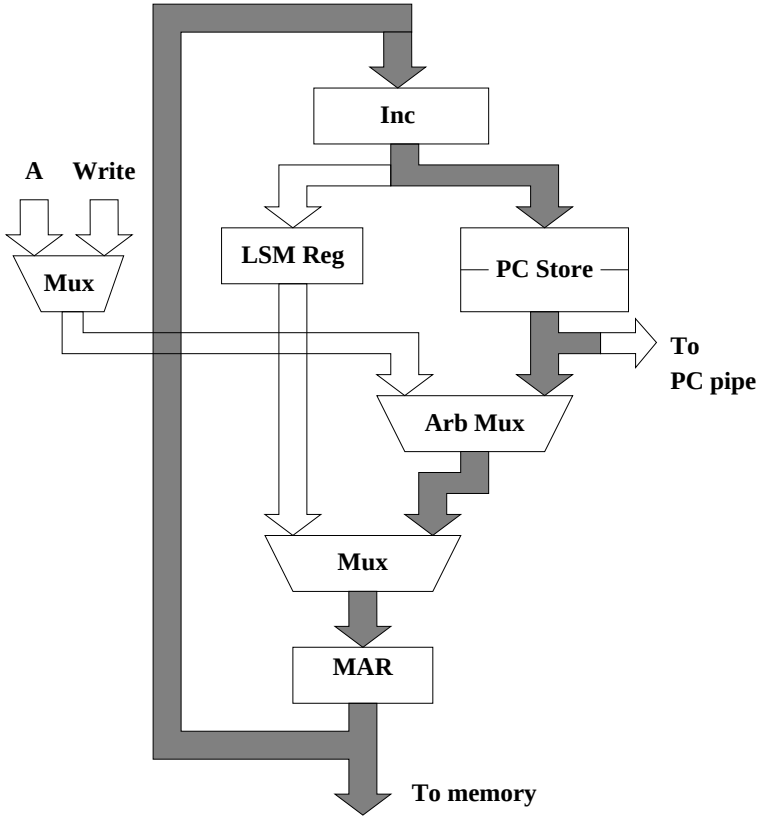
Rainbow sub-languages all map to APA:

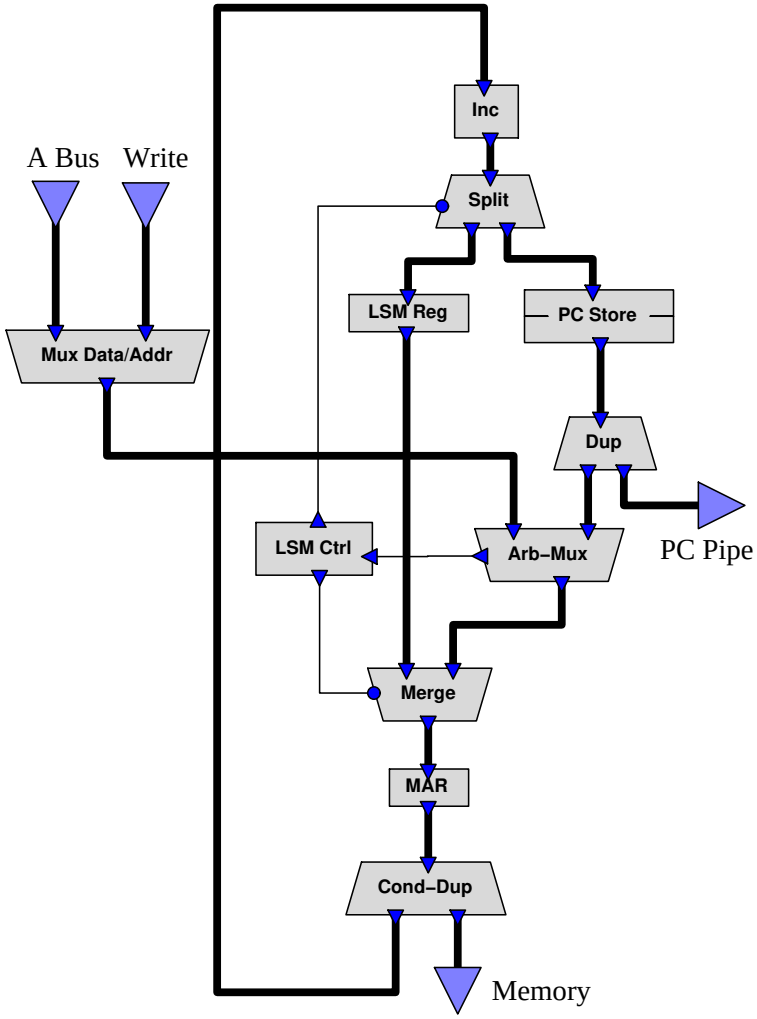
- interworking
- semantic consistency

Simulation


Buffer (1)
5
Dup
5
5
Id
5
Buffer (0)
3
Function (Sum)
8
Buffer
Buffer (1)
5
Dup
5
Id
Buffer (0)
Function (Sum)
Buffer
8
Buffer (1)
Dup
Id
Buffer (0)
5
Function (Sum)
Buffer
8
Buffer (1)
8
Dup
8
8
Id
8
Buffer (0)
5
Function (Sum)
13
Buffer

Summary


A
Write
Mux
Inc
LSM Reg
PC Store
To
PC pipe
Arb Mux
Mux
MAR
To memory


A Bus
Write
Mux Data/Addr
Inc
Split
LSM Reg
PC Store
Dup
LSM Ctrl
Arb–Mux
PC Pipe
Merge
MAR
Cond–Dup
Memory

- micropipeline design methodology
- rainbow Design framework with interworking
- abstract modelling
- application-specific methods
- prototype tool implementation
  - Visual Green editor/simulator
  - Rainbow expression compiler/evaluator
  - embedded into extensible design environment
- formal semantics
- analysis

URL:

`http://www.cs.man.ac.uk/fmethods/projects/AHV-PROJECT/ahv-project.html`