OUTLINE OF PART 3

- Continuation of implementation techniques
- The TANGRAM language for high-level descriptions of handshake circuits
- Implementation of TANGRAM designs

NULL-CONVENTION LOGIC

- Generalization of Muller C-element
- N inputs
- M out of N inputs should be all ‘0’ or ‘1’ for the gate’s output to switch to ‘0’ or ‘1’.
- Also called “threshold gate with hysteresis”.
- See Figures 5.11 and 5.12 of [Sparsoe 2001].

TRANSISTOR-LEVEL IMPLEMENTATIONS (1)

- Strongly indicating full adder example in Figure 5.14 of [Sparsoe 2001].
- Long chain of pull-up transistors can be replaced by a single transistor controlled by intermediate signal.

TRANSISTOR-LEVEL IMPLEMENTATIONS (2)

- Martin has proposed efficient implementations of weakly-indicating full adder:
  - For dependencies of validity and empty indication see Figure 5.15 of [Sparsoe 2001]
  - Empty indication in constant time.
- Implementations with shared pull-down network:
  - See Figure 5.16 of [Sparsoe 2001]
  - Transistor count comparable to traditional full adder.
HYBRID FUNCTION BLOCKS (1)

- Sometimes it is interesting to mix “dual rail” and “bundled data” signals in one block.
- See Figure 5.17 of [Sparsoe 2001]
- Carry arrives as bundled data and is converted to dual rail internally.
- Requires a complex completion detector.
- Complexity of completion detector can be reduced by mixing strongly and weakly indicating carry circuits.

HYBRID FUNCTION BLOCKS (2)

- Multiplexers and demultiplexers can benefit from hybrid implementations:
  - Bundled data for data signals
  - Dual rail for control signals
- See Figure 5.19 of [Sparsoe 2001].

MUTUAL EXCLUSION, ARBITRATION AND METASTABILITY

- Mutual exclusion element (MUTEX) should pass one of two incoming requests.
- Danger of metastability.
- See Figure 5.20 of [Sparsoe 2001].
- Mutex element can be used in handshake arbitration circuit, see Figure 5.21 of [Sparsoe 2001].

FURTHER READING ON TANGRAM

TANGRAM DESIGN FLOW

- Specify computation in high-level language Tangram
- Map description on hardware in a 1-to-1 fashion:
  - First onto “handshake circuit”, an abstract graph-based representation of the description
  - Then onto hardware primitives corresponding to elements in handshake circuit
  - Finish with “peephole optimization” (to eliminate local inefficiencies), placement and routing

THE TANGRAM LANGUAGE

- Based on Hoare’s CSP (Communicating Sequential Processes) which is also the basis of many other languages
- Consists of modules communicating through point-to-point channels
- Module behavior can be parallel or sequential (the word “sequential” in CSP is somewhat misleading)
- Has also some similarity to VHDL simulation model that consists of communicating truly sequential processes
- Its commercial successor, as used in products of Handshake Solutions, is called Haste.

TANGRAM EXAMPLES

- See Figures 8.2 to 8.5 of [Sparsoe 2001]:
  - Shift register with one input and one output channel (“\texttt{in?x}” indicates a read operation on channel \texttt{in} whose result is assigned to \texttt{x}; “\texttt{out!y}” means a write operation on channel \texttt{out} using the value of \texttt{y}).
  - Note internal channel and parallel composition with “||” in FIFO example.
  - Note “;” for sequential composition in first GCD; “\texttt{do \_od}” statement is a while loop.
  - Note “guarded do loop” which repeats until all guards are false.

BUNDLED-DATA CHANNEL TYPES

- Four types:
  - Nonput channel
  - Push channel
  - Pull channel
  - Biput channel
- Channel has passive (°) and active (•) endpoints
- See Figure 7.1 of [Sparsoe 2001]
CORRESPONDING HANDSHAKE CIRCUITS

- Each primitive is drawn as a circle containing a symbol representing its functionality.
- Active (black dot) and passive (white dot) channel interfaces are drawn against the circles.
- Arrows in channels indicate data-flow direction, if any.
- See Figures 8.6 to 8.8 of [Sparsoe 2001] using various primitives:
  - repeater, sequencer, variable, transferrer, passivator, etc.

DATA-VALID SCHEMES FOR FOUR-PHASE BUNDLED DATA

- “Return to zero” in request and acknowledge signals does not necessarily imply return to zero in data.
- See Figures 3 and 4 of [Peeters 1995] for push and pull channels with early, broad and late data-valid schemes.
- Reduced broad scheme, makes return-to-zero phase irredundant, see Figures 5, 6 and 8 of [Peeters 1995].
  - “Matched delay” can be halved, or one gets extra safety margin.

MATCHED-DELAY IMPLEMENTATION

- Matched delays are multiples of a unit-delay circuit built from a NAND and NOR gate connected in series:

  ![Diagram of matched delay circuit]

- Buffers are added to the netlist to make sure that signal transitions are fast.