Comp 331: Intro to Hardware
Data-paths and Control Units
GCD Extension to 8 Bit Values

Modification of 4-bit data-paths to 8 bits (simple extension)
Modification of Control Logic to allow separate loading of x and y registers

Comp 331: Intro to Hardware
Data-paths and Control Units – Square Roots

```c
integer sqrt (integer a) {
    integer square = 1;
    integer delta = 3;
    while (square <= a) {
        square = square + delta;
        delta = delta + 2;
    }
    return (delta/2 -1);
}
```

Control Path

Data-path

Data and Control Path
Generic Register

library IEEE;
use IEEE_STD_LOGIC_1164.all;

entity regr2 is
  generic(N: integer;
    BIT0: STD_LOGIC;
    BIT1: STD_LOGIC);
  port(
    d: in STD_LOGIC_VECTOR(N-1 downto 0);
    load: in STD_LOGIC;
    reset: in STD_LOGIC;
    clk: in STD_LOGIC;
    q: out STD_LOGIC_VECTOR(N-1 downto 0)
  );
end regr2;

architecture regr2 of regr2 is
begin
  process(clk, reset)
  begin
    if reset = '1' then
      q <= (others => '0');
      q(0) <= BIT0;
      q(1) <= BIT1;
    elsif (clk'event and clk = '1' then
      if load = '1' and clk = '1' then
        q <= d;
      end if;
    end if;
  end process;
end regr2;

Top Design for Square Root Algorithm

Implement Example 65

Implement
Listing 9.6 (modified)
Listing 9.7 (regr2.vhd)
Listing 9.8 (SQRTctrl.vhd)
Listing 9.9 (SQRT.vhd)
Listing 9.10 (sqrt_top.vhd)