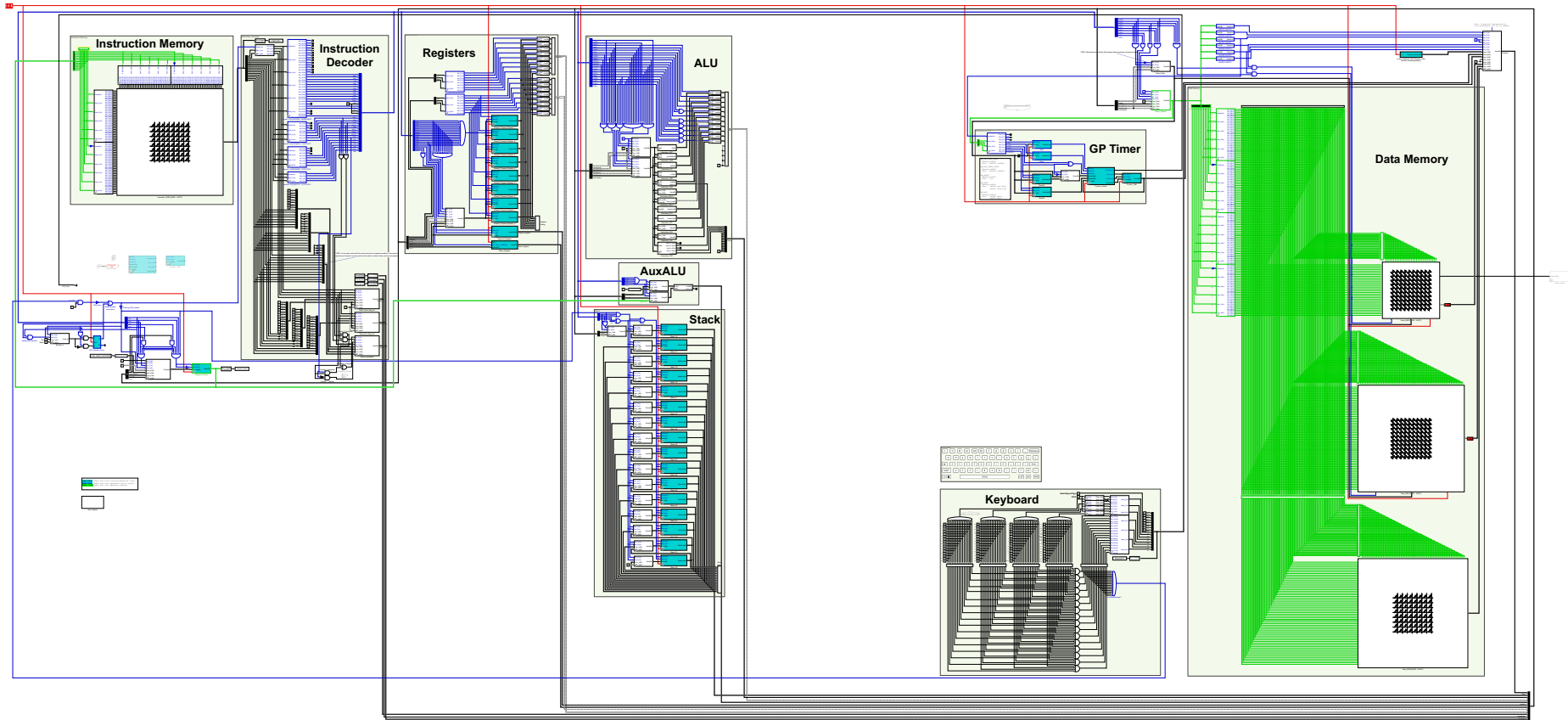
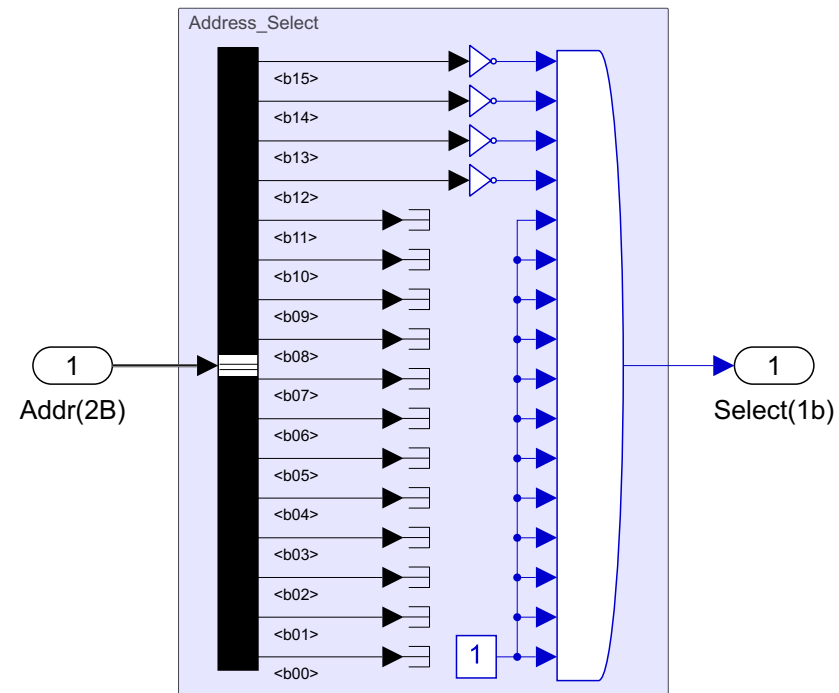
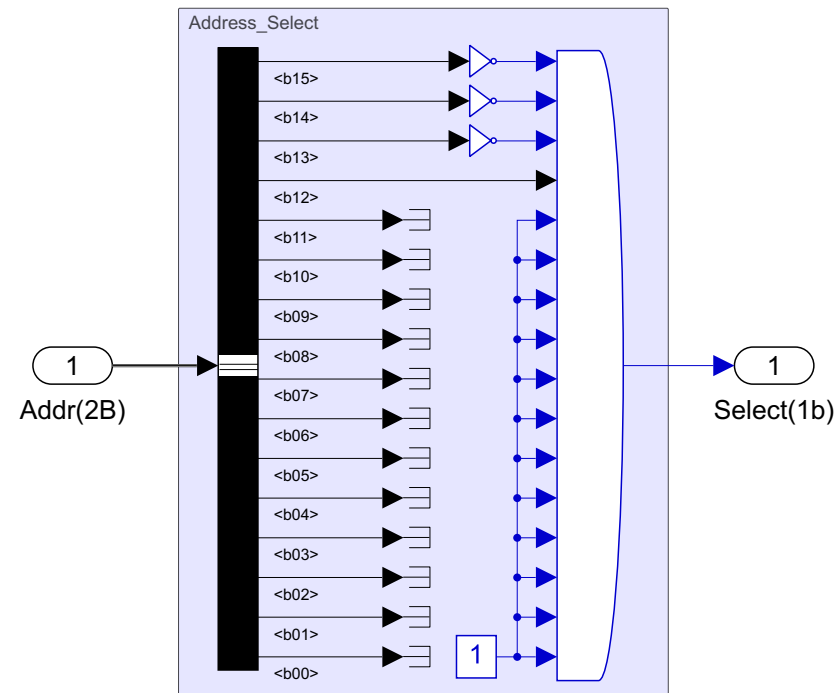


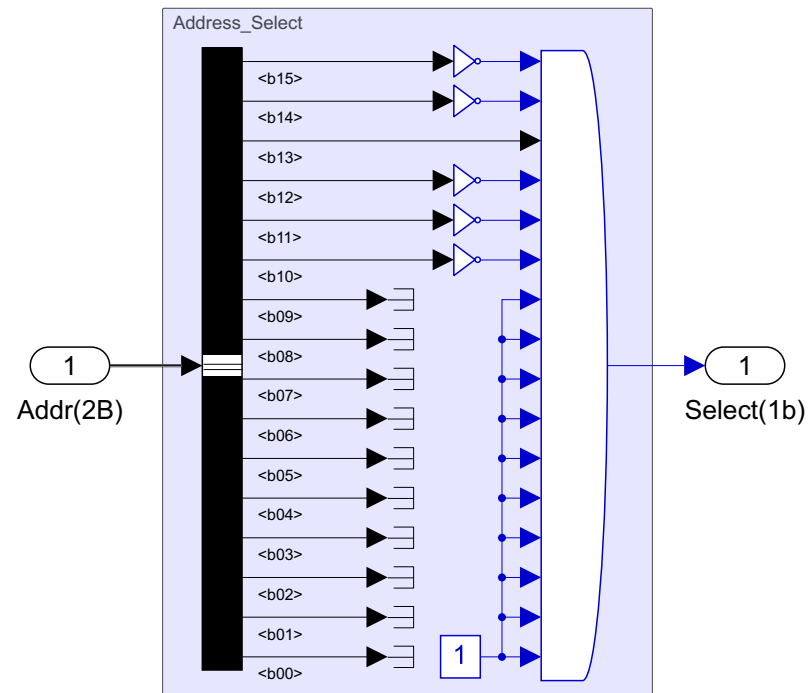
Computer

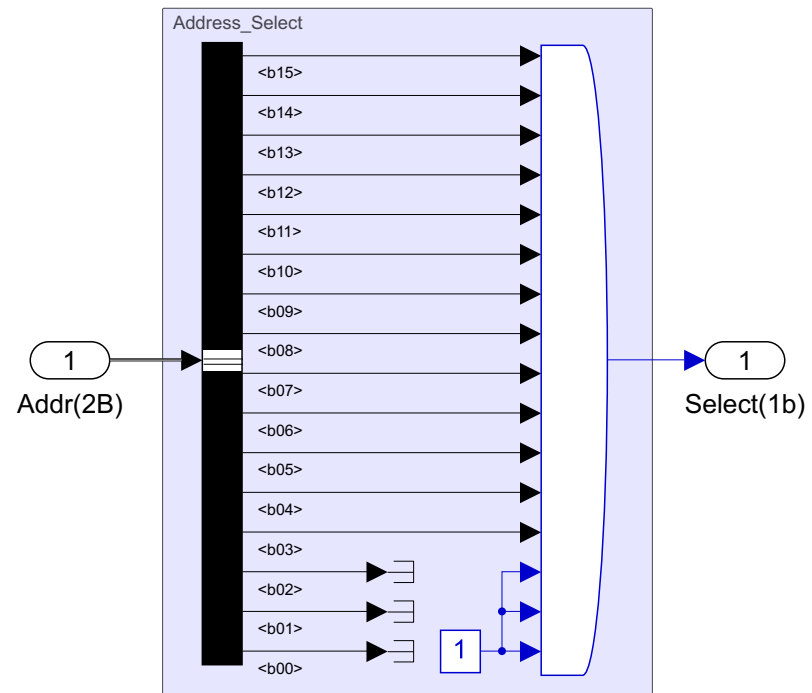


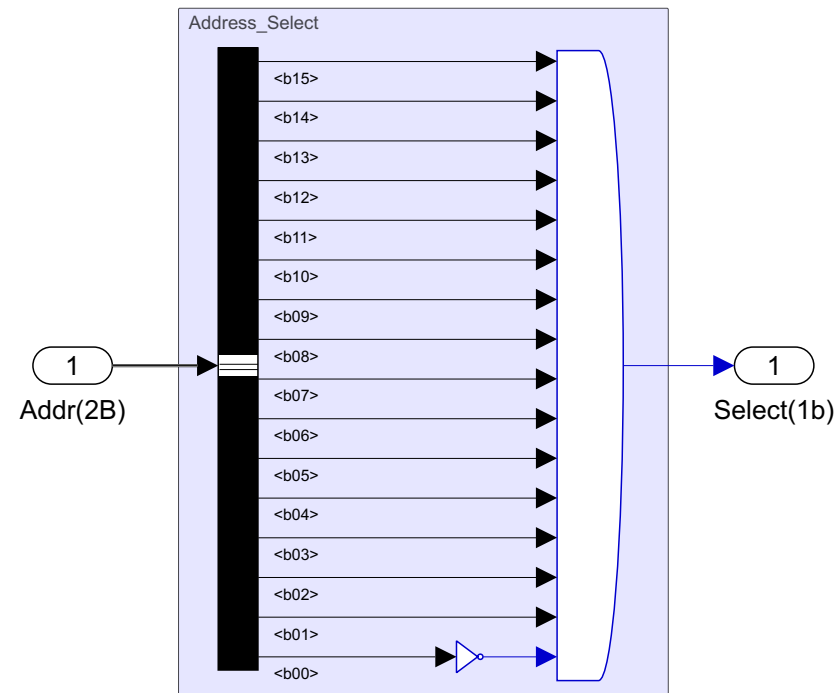
F:\Workspace\05_Projekty\Computer\Computer.slx

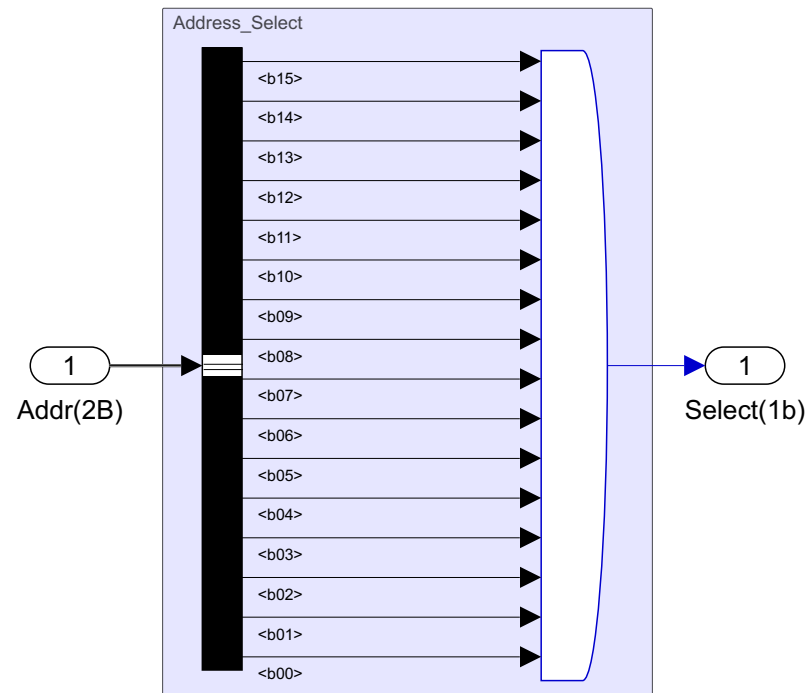




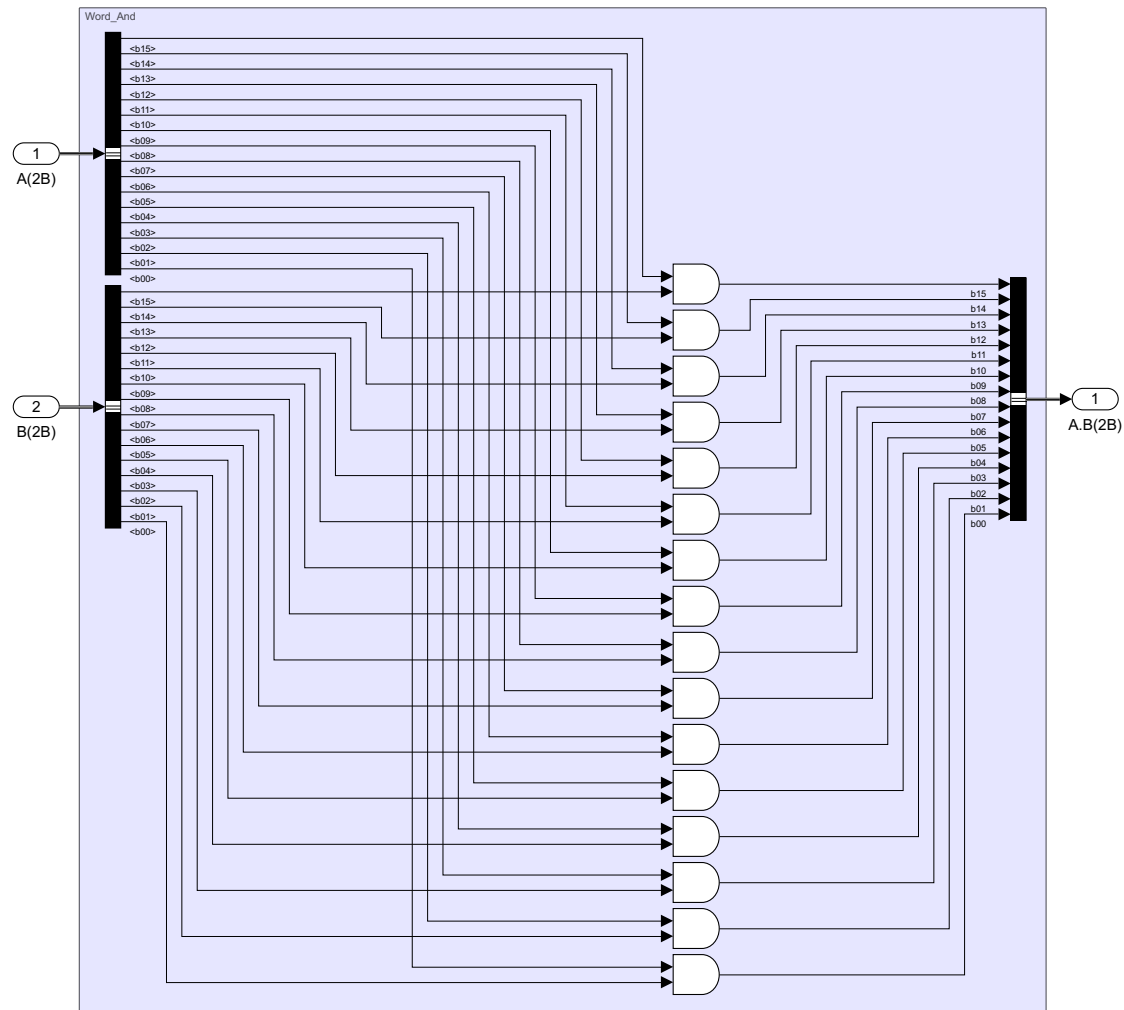




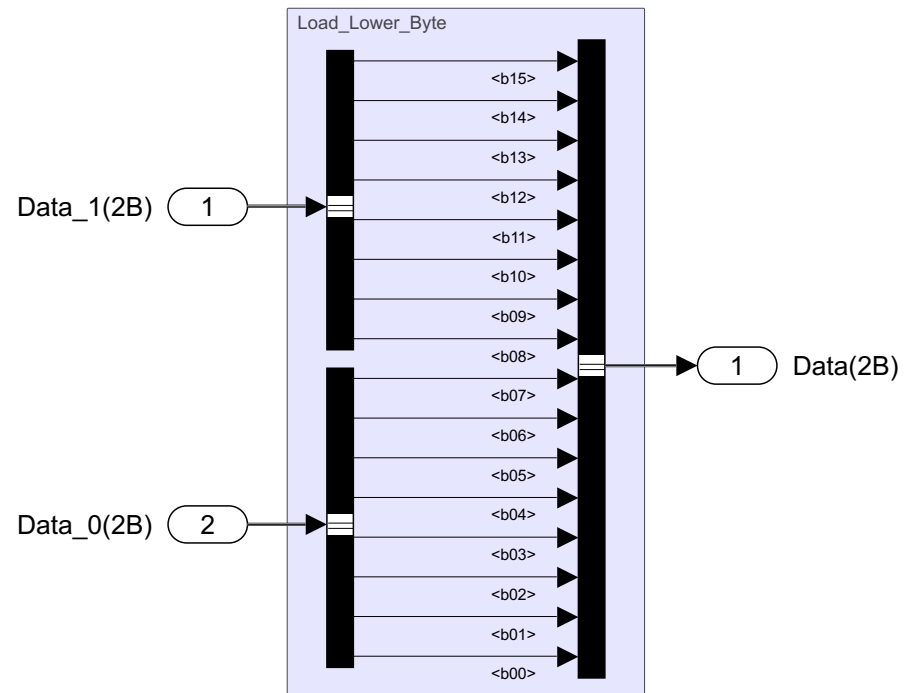


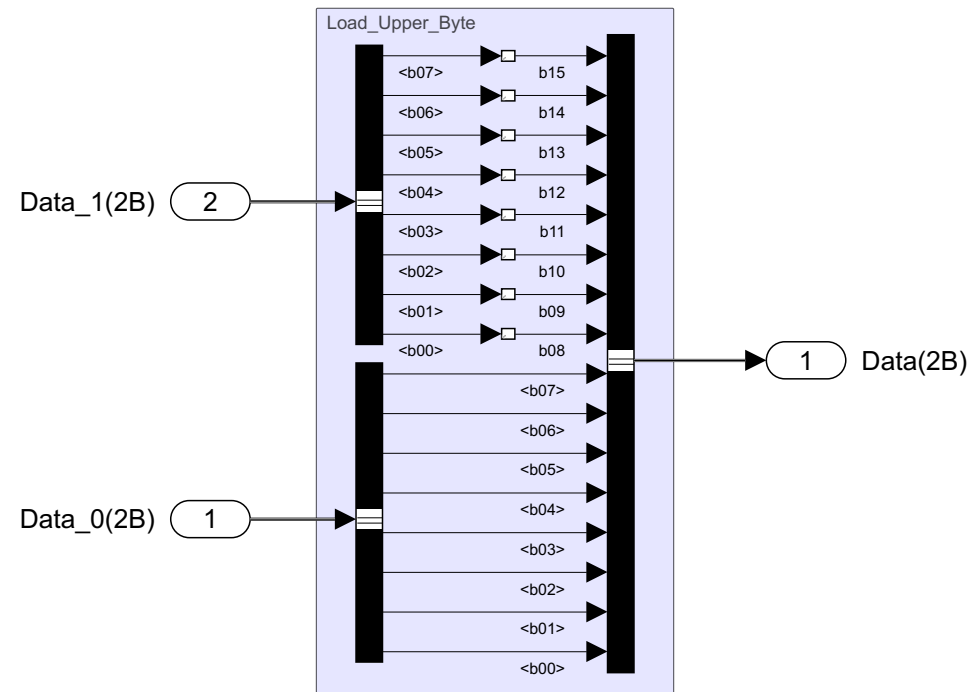


Computer/Instruction_AND

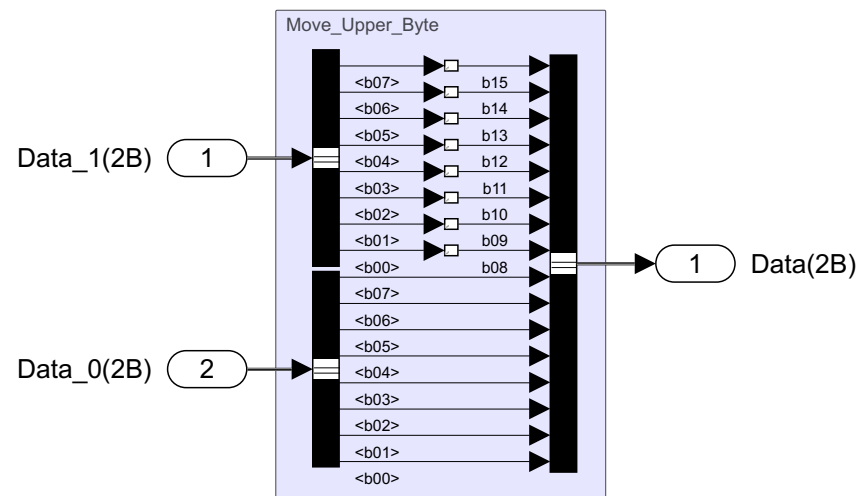


F:\Workspace\05_Projekty\Computer\Computer.slx

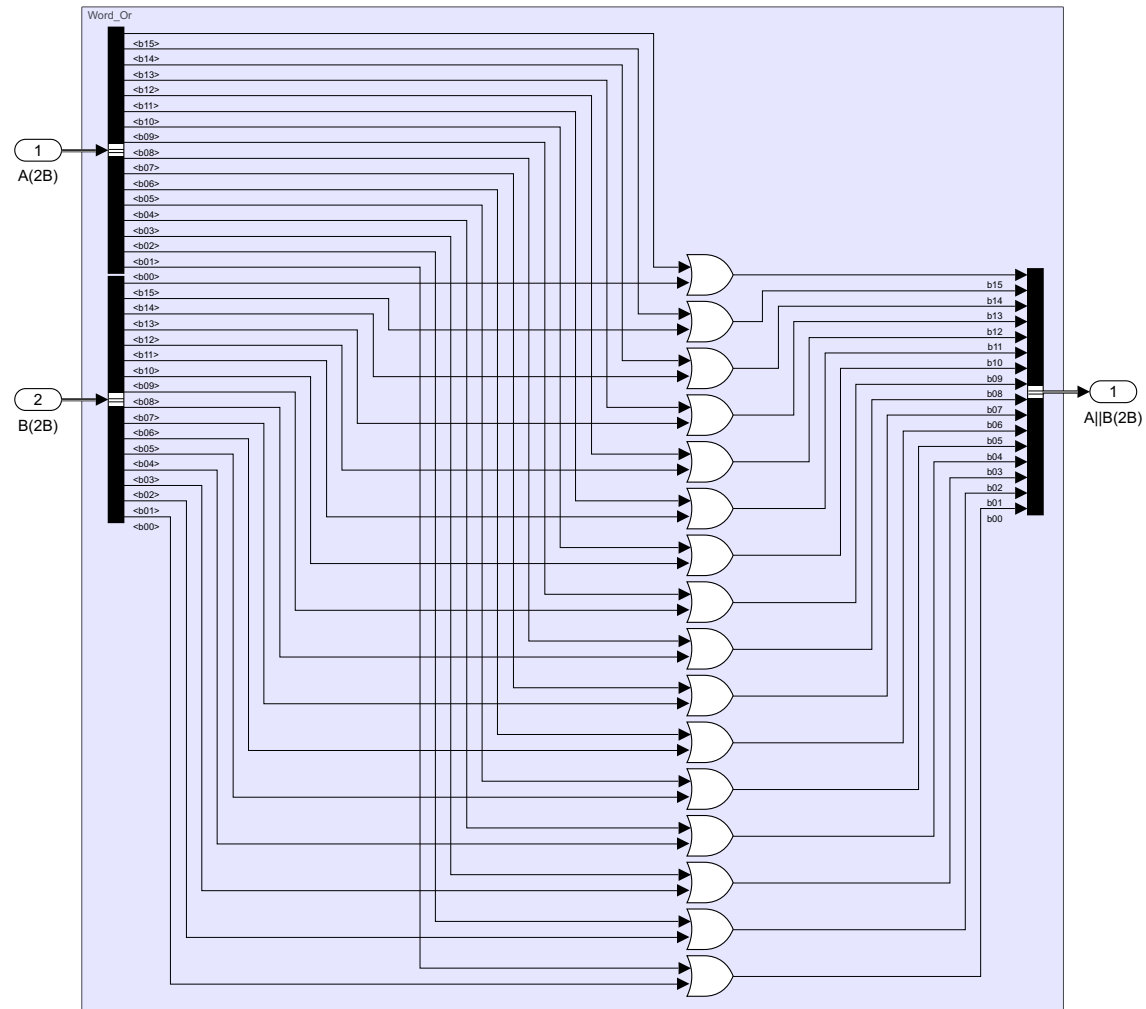




Computer/Instruction_MOVU

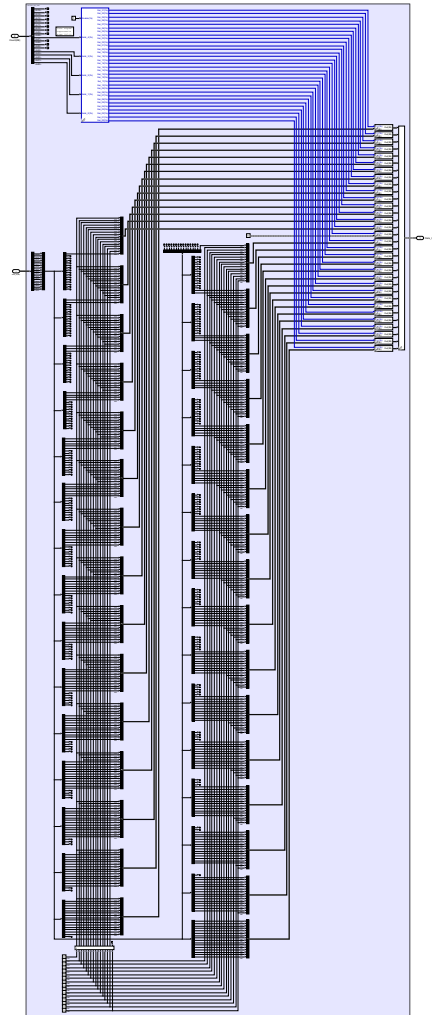


Computer/Instruction_OR

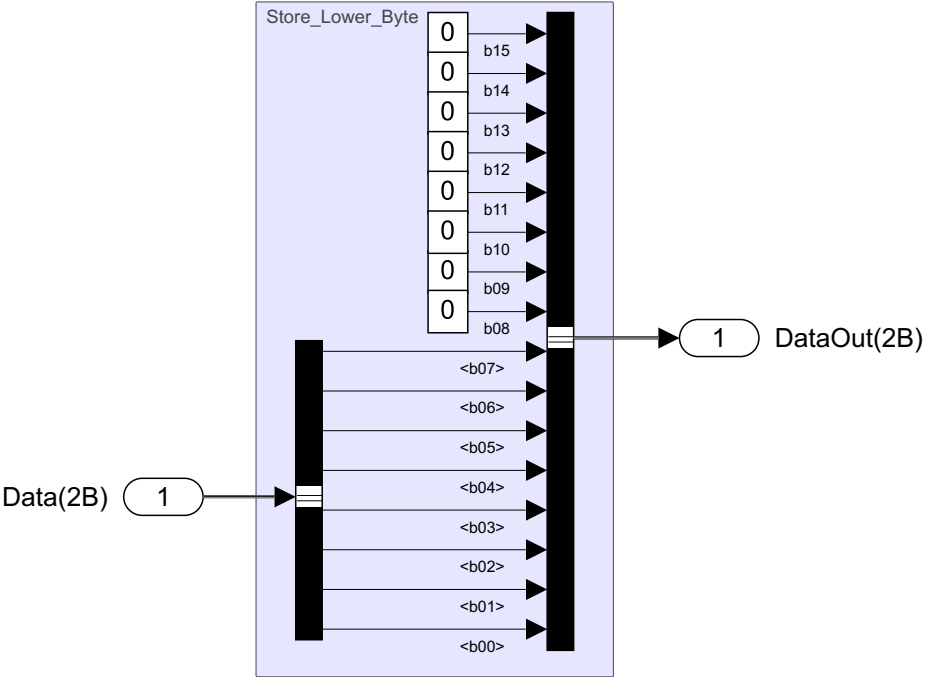


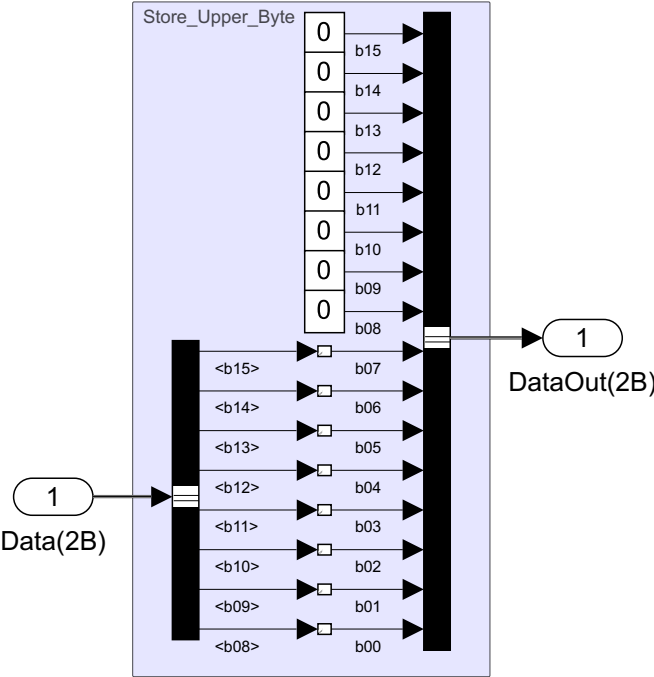
F:\Workspace\05_Projekty\Computer\Computer.slx

Computer/Instruction_SHIFT



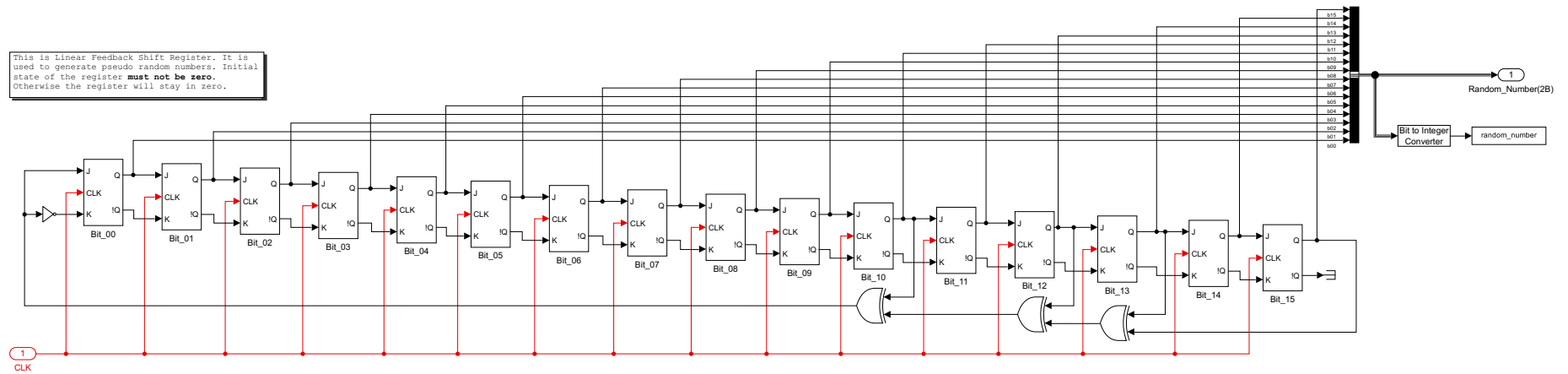
F:\Workspace\05_Projekty\Computer\Computer.slx



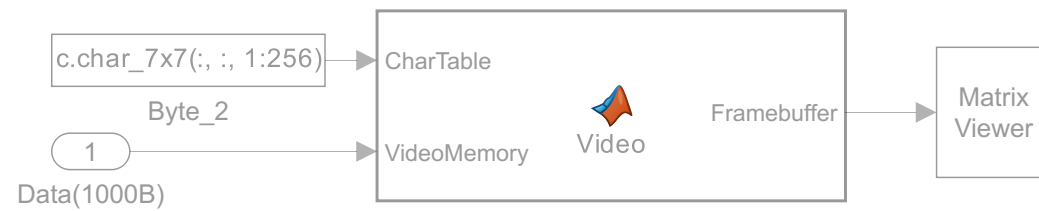


Computer/Linear_Feedback_Shift_Register_15bit (random number generator)

This is Linear Feedback Shift Register. It is used to generate pseudo random numbers. Initial state of the register **must not be zero**. Otherwise the register will stay in zero.



F:\Workspace\05_Projekty\Computer\Computer.slx



```

function Framebuffer = Video(CharTable, VideoMemory)
    VIDEO_MEM_SIZE = 1000;
    MAX_X = 40;
    MAX_Y = 25;
    CHAR_SIZE = 8;
    Framebuffer = zeros(MAX_Y*CHAR_SIZE, MAX_X*CHAR_SIZE, 'uint8');

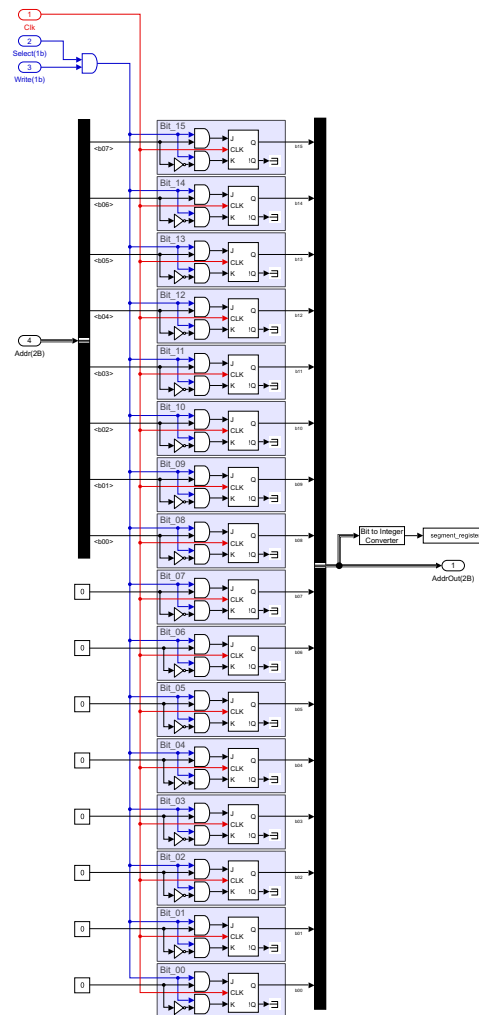
    for (i = 1 : VIDEO_MEM_SIZE)
        y = idivide(uint16(i) - 1, uint16(MAX_X)) + 1;
        x = mod(uint16(i) - 1, uint16(MAX_X)) + 1;

        if (0 == VideoMemory(i))
            % Interpret as a space/blank character.
            ordNum = uint8(32);
        else
            ordNum = VideoMemory(i);
        end

        Framebuffer((CHAR_SIZE * (y - 1) + 1) : (CHAR_SIZE * (y - 1) + CHAR_SIZE), ...
                    (CHAR_SIZE * (x - 1) + 1) : (CHAR_SIZE * (x - 1) + CHAR_SIZE)) ...
            = CharTable(:, :, ordNum);
    end
end
end

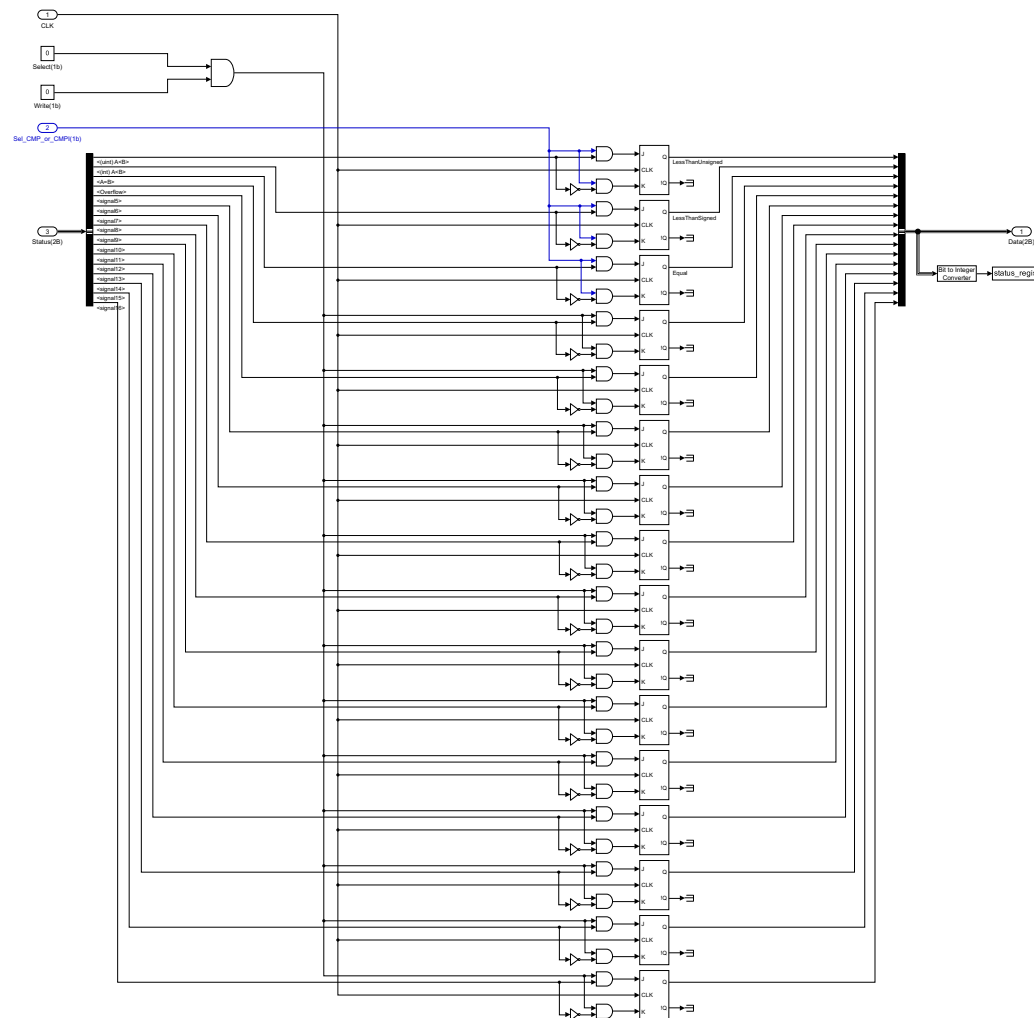
```

Computer/Segment_Register

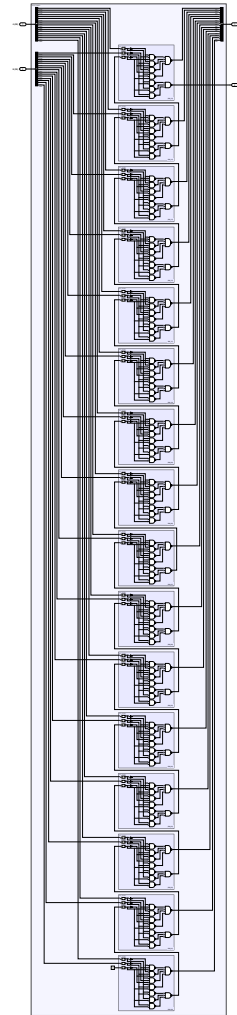


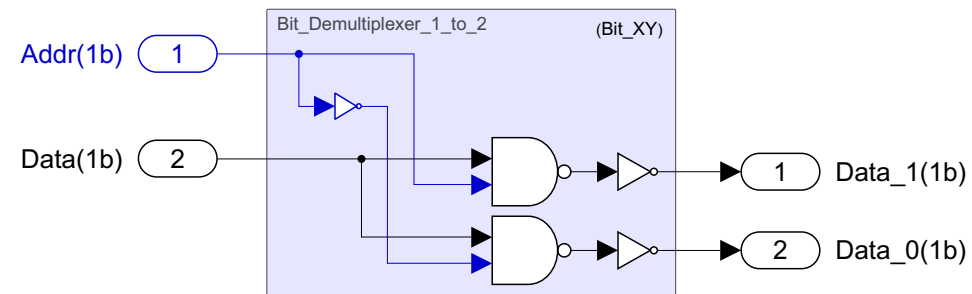
F:\Workspace\05_Projekty\Computer\Computer.slx

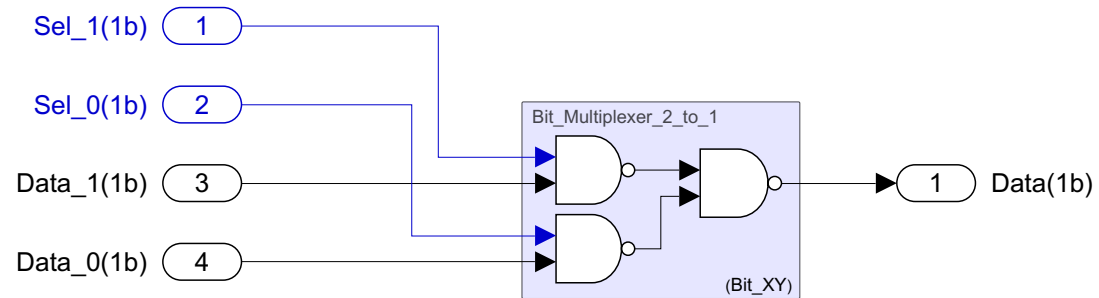
Computer/Status_Register



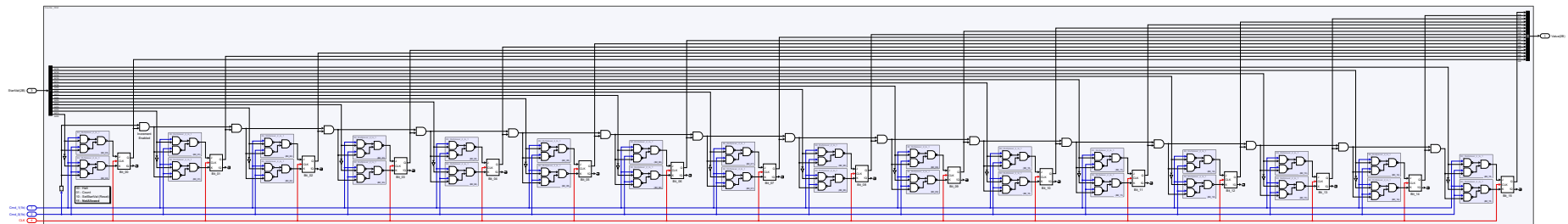
F:\Workspace\05_Projekty\Computer\Computer.slx





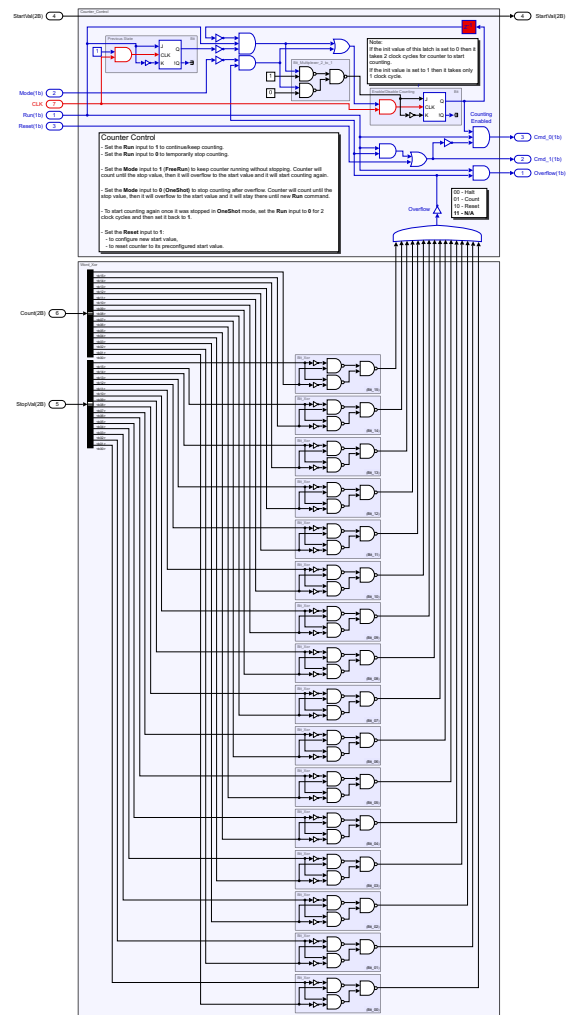


Counters_Lib/Counter_16bit

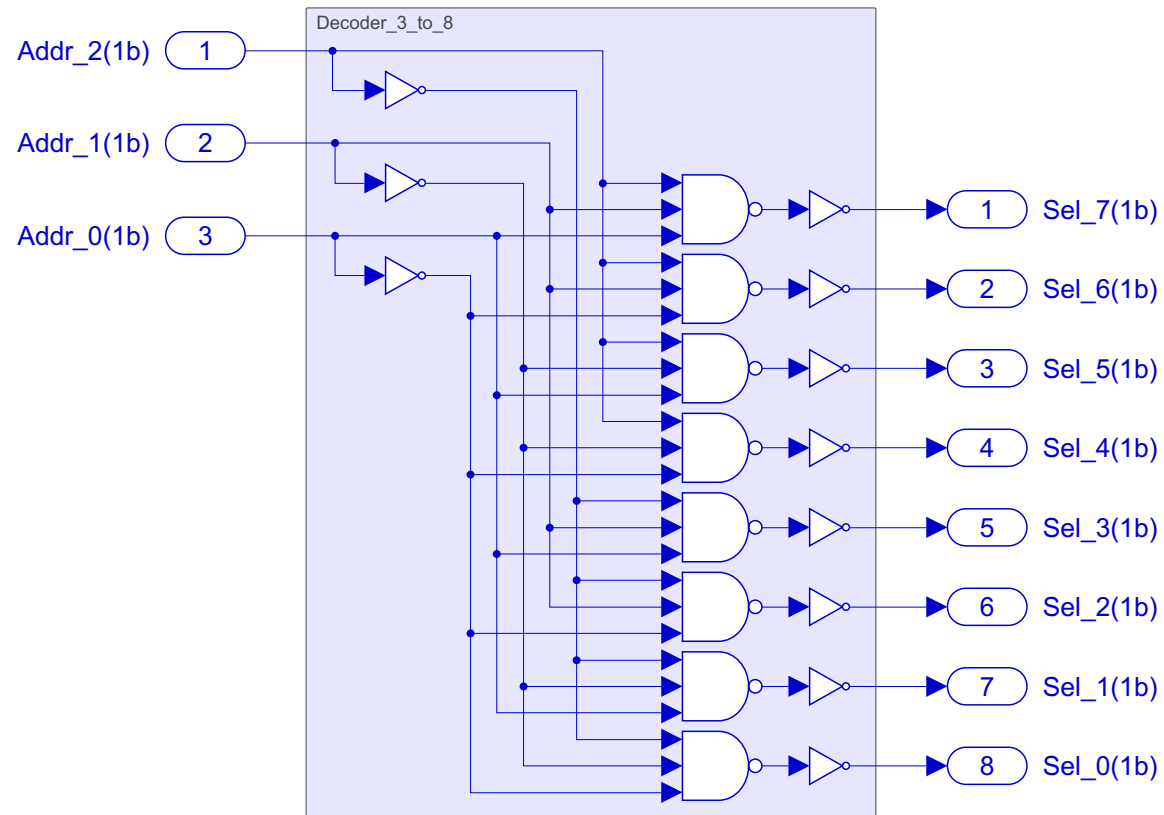


F:\Workspace\05_Projekty\Common\Library_Matlab\Simulink\Counters\Counters_Lib.slx

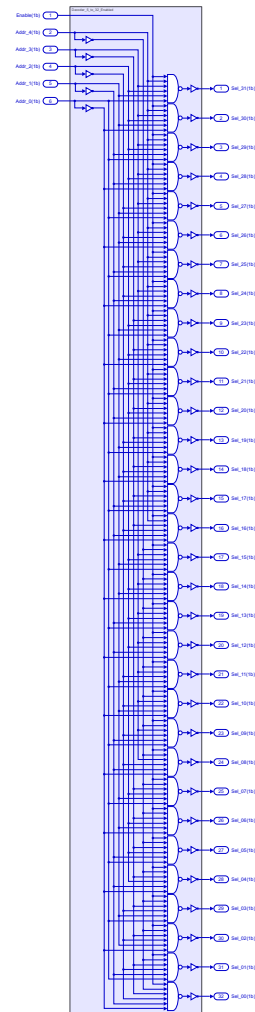
Counters_Lib/Counter_Control



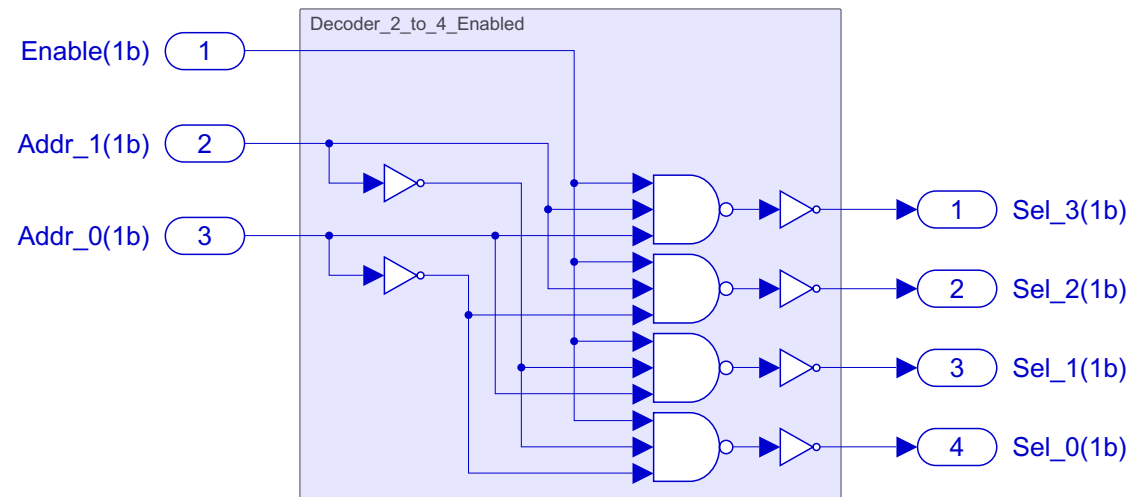
F:\Workspace\05_Projekty\Common\Library_Matlab\Simulink\Counters\Counters_Lib.slx



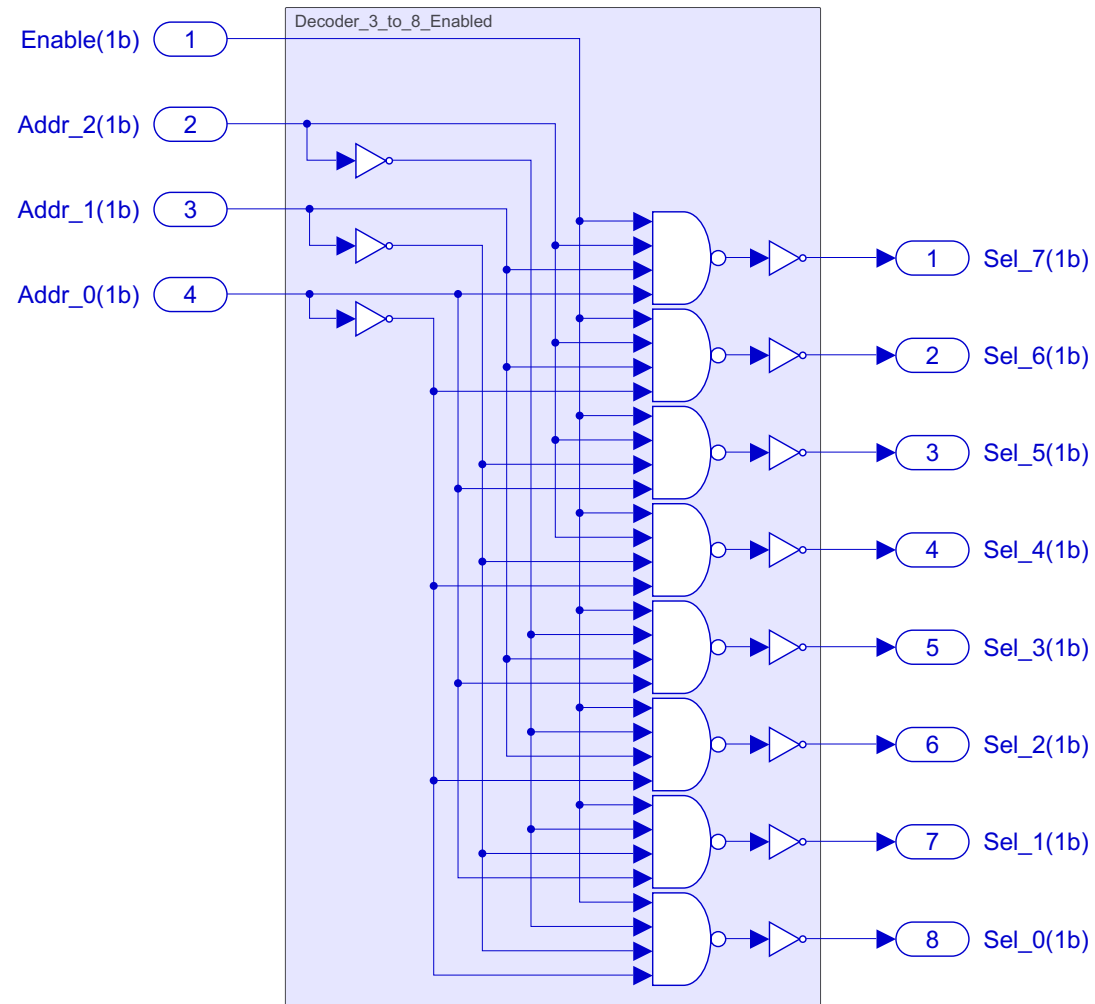
Decoders_Lib/Dec_32_Enbl



F:\Workspace\05_Projekty\Common\Library_Matlab\Simulink\Decoders\Decoders_Lib.mdl

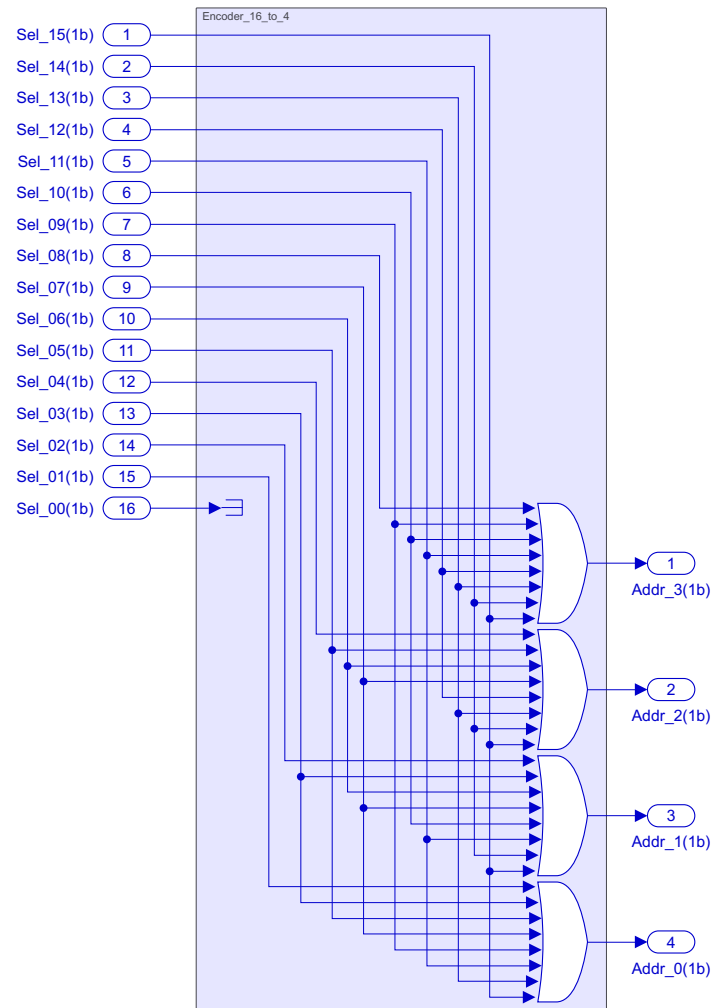


Decoders_Lib/Dec_8_Enbl

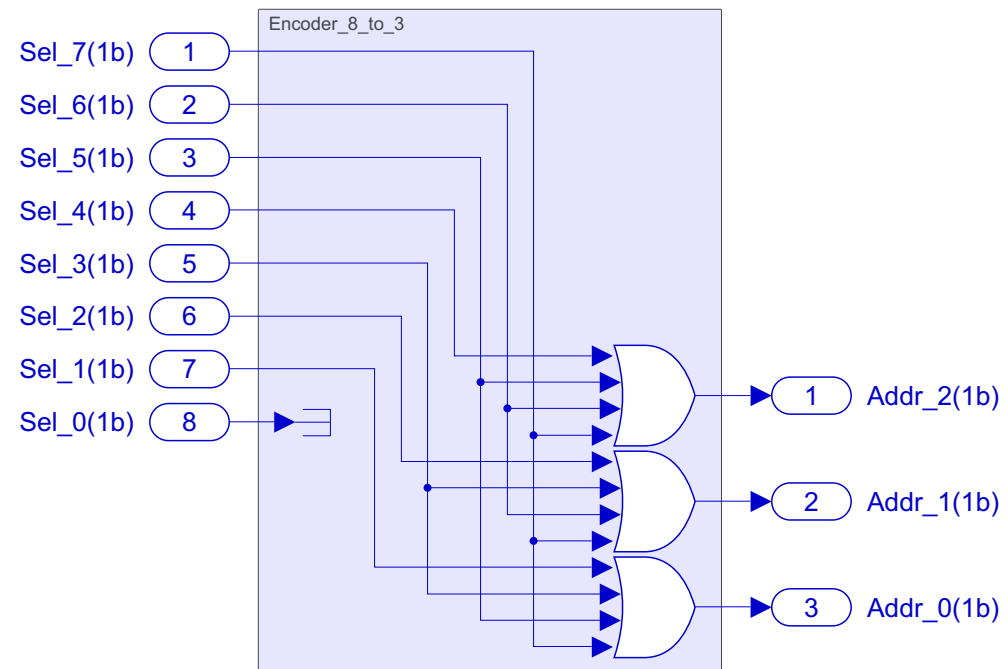


F:\Workspace\05_Projekty\Common\Library_Matlab\Simulink\Decoders\Decoders_Lib.mdl

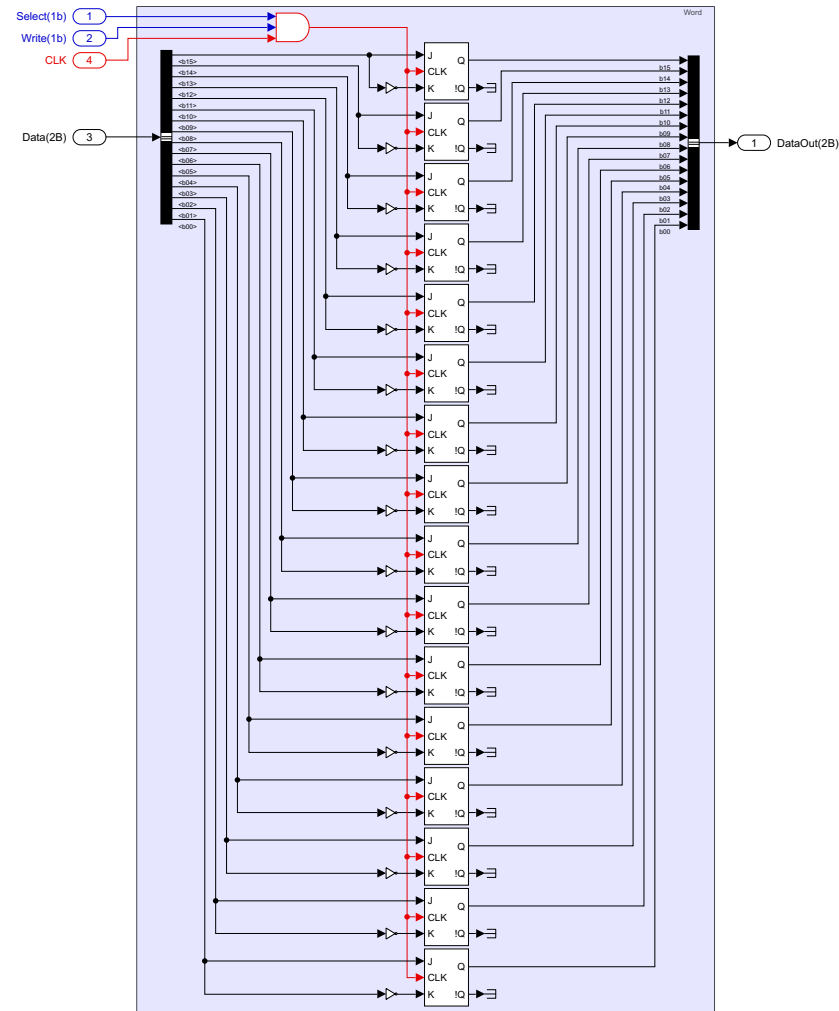
Encoders_Lib/Enc_16



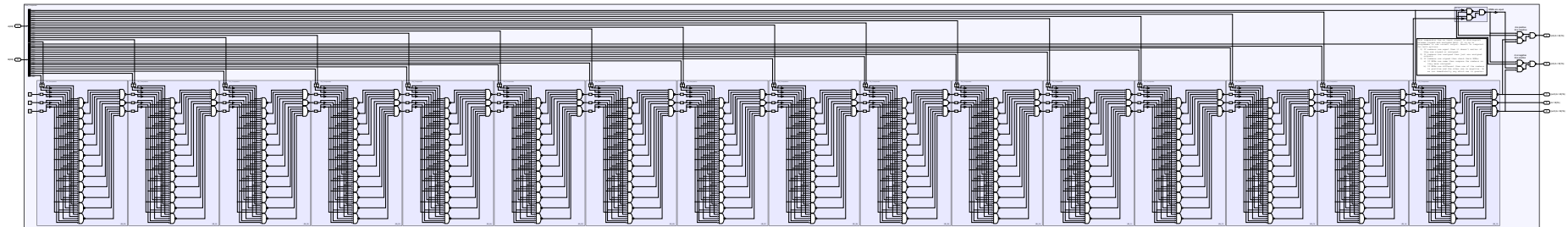
F:\Workspace\05_Projekty\Common\Library_Matlab\Simulink\Encoders\Encoders_Lib.mdl

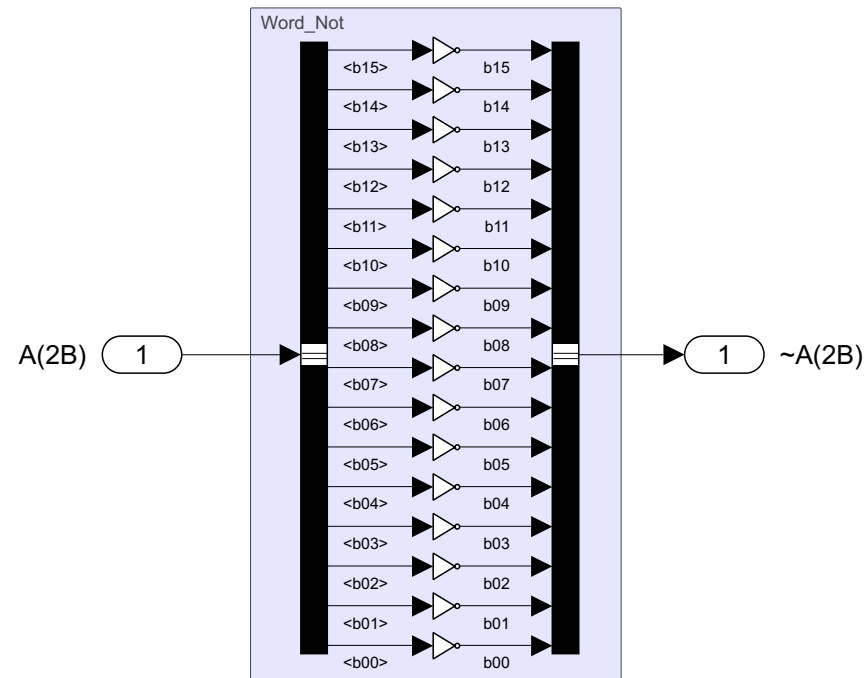


Memory_Blocks_Lib/Word

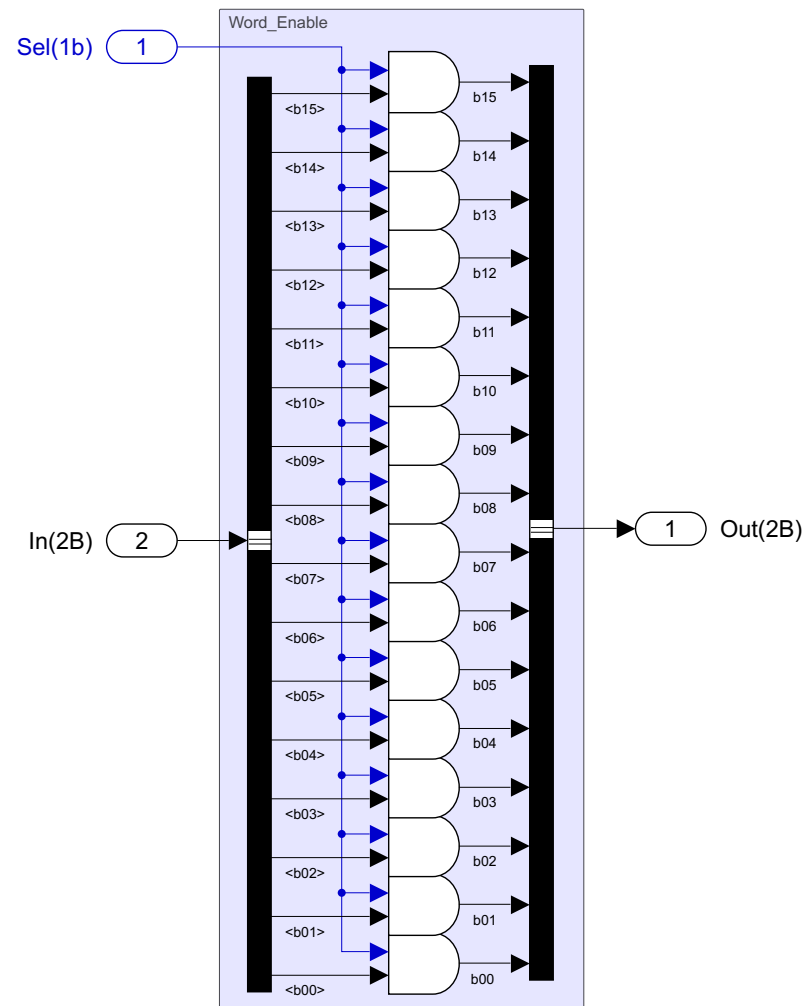


F:\Workspace\05_Projekty\Common\Library_Matlab\Simulink\Memory_Blocks\Memory_Blocks_Lib.mdl



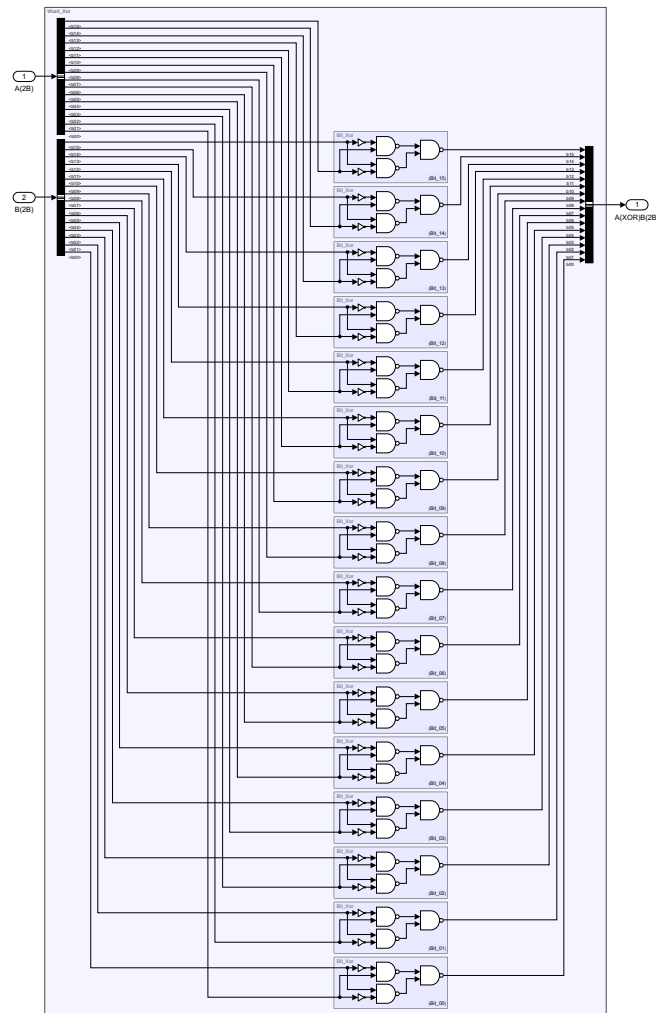


Miscellaneous/Word_Enable

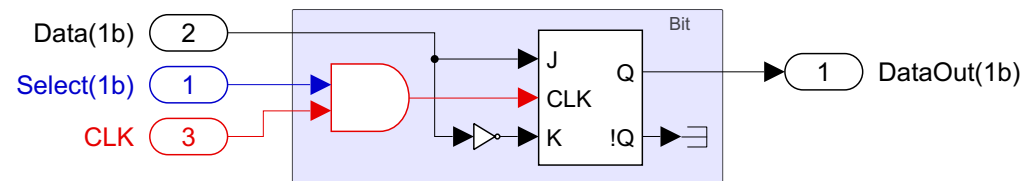


F:\Workspace\05_Projekty\Common\Library_Matlab\Simulink\Miscellaneous\Miscellaneous.mdl

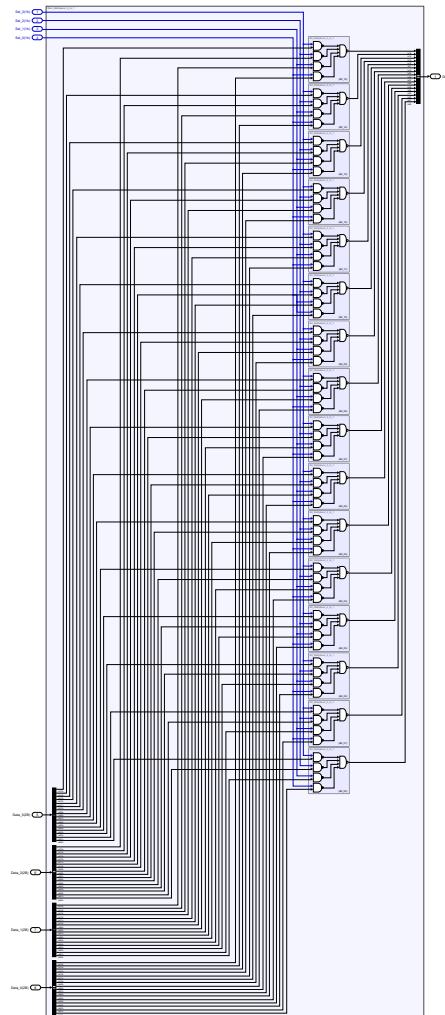
Arithmetic_Operations_Lib/Word_Xor



F:\Workspace\05_Projekty\Common\Library_Matlab\Simulink\Arithmetic_Operations\Arithmetic_Operations_Lib.mdl

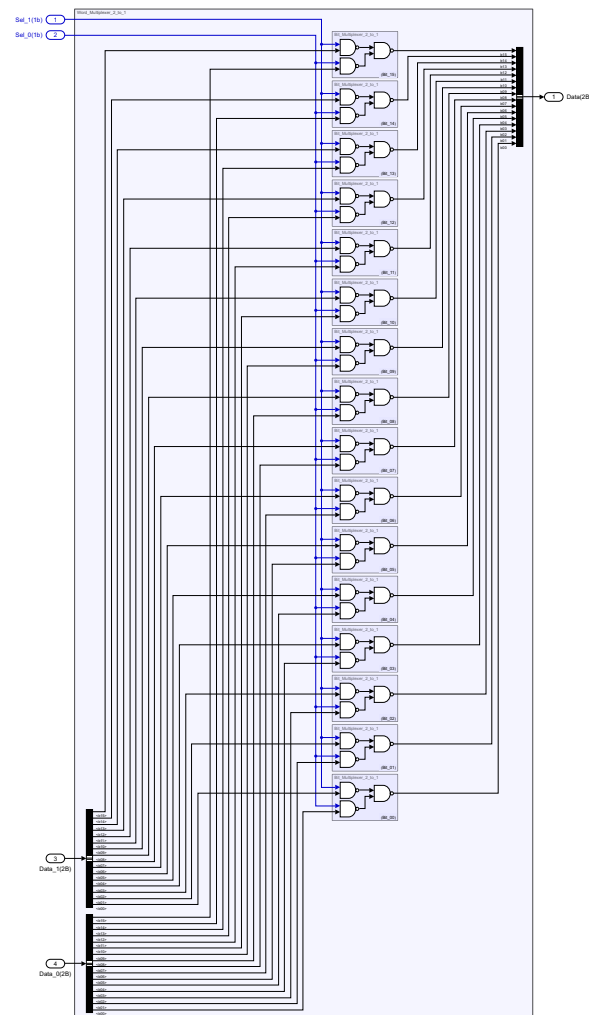


Word_Multiplexers_Lib/Word_Mux_4



F:\Workspace\05_Projekty\Common\Library_Matlab\Simulink\Multiplexers_Word\Word_Multiplexers_Lib.mdl

Word_Multiplexers_Lib/Word_Mux_2



F:\Workspace\05_Projekty\Common\Library_Matlab\Simulink\Multiplexers_Word\Word_Multiplexers_Lib.mdl

