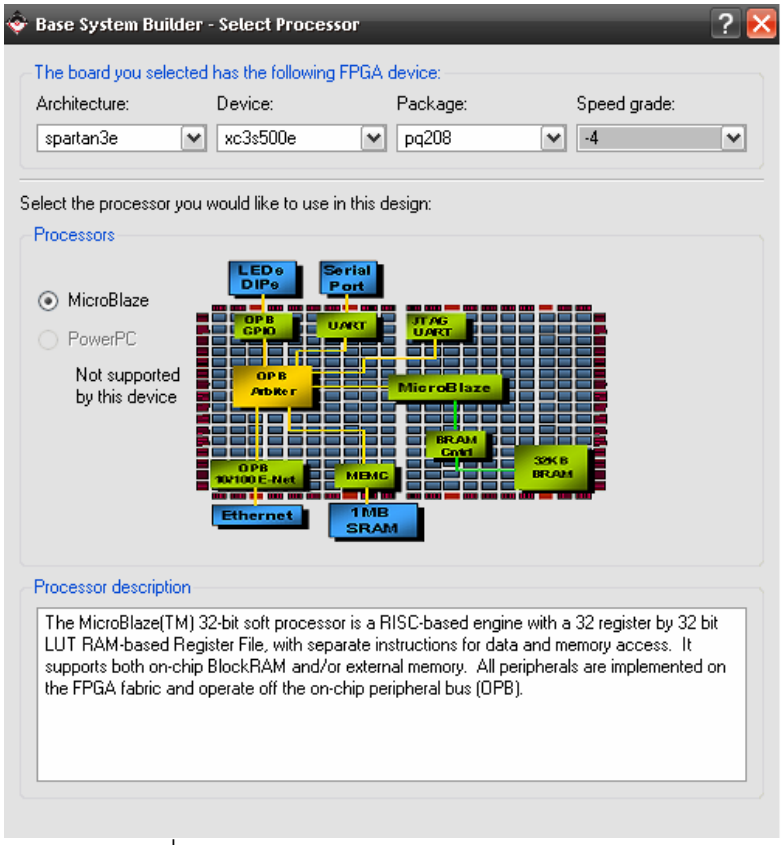
ภาคผนวก ข

การสร้างระบบ MicroBlaze ด้วย Xilinx Platform Studio

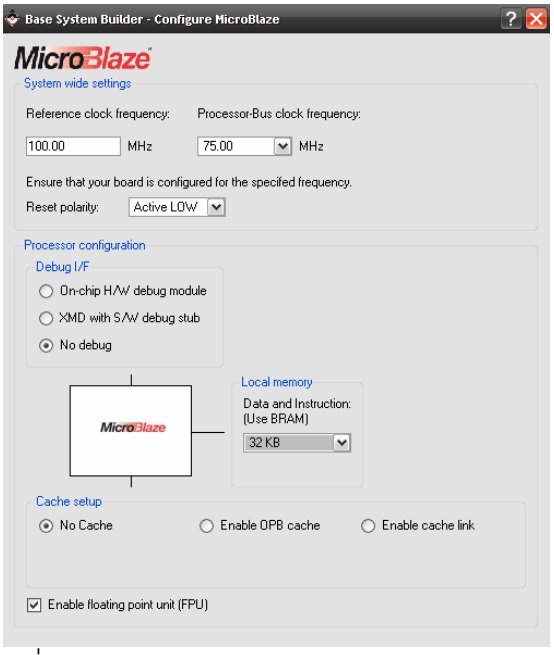
งานวิจัยนี้ใช้ Xilinx Platform Studio 8.1i (XPS 8.1i) ในการสร้างระบบ MicroBlaze ซึ่งมีขั้นตอนดังนี้

1. เมื่อเปิด XPS 8.1i ให้เลือกเมนู Base System Builder wizard (recommended) กด OK
2. เลือก Folder ที่ต้องการเก็บ Project พิมพ์ชื่อ Project แล้วกด OK
3. เมื่อขึ้นหน้าต่าง Base System Builder – Welcome ให้เลือก I would like to create a new design แล้วกด Next
4. ที่หน้า Base System Builder – Select Board ให้เลือก I would like to create a system for a custom board แล้วกด Next
5. ที่หน้า Select Processor ให้เลือกดังรูปที่ ข.1 แล้วกด Next



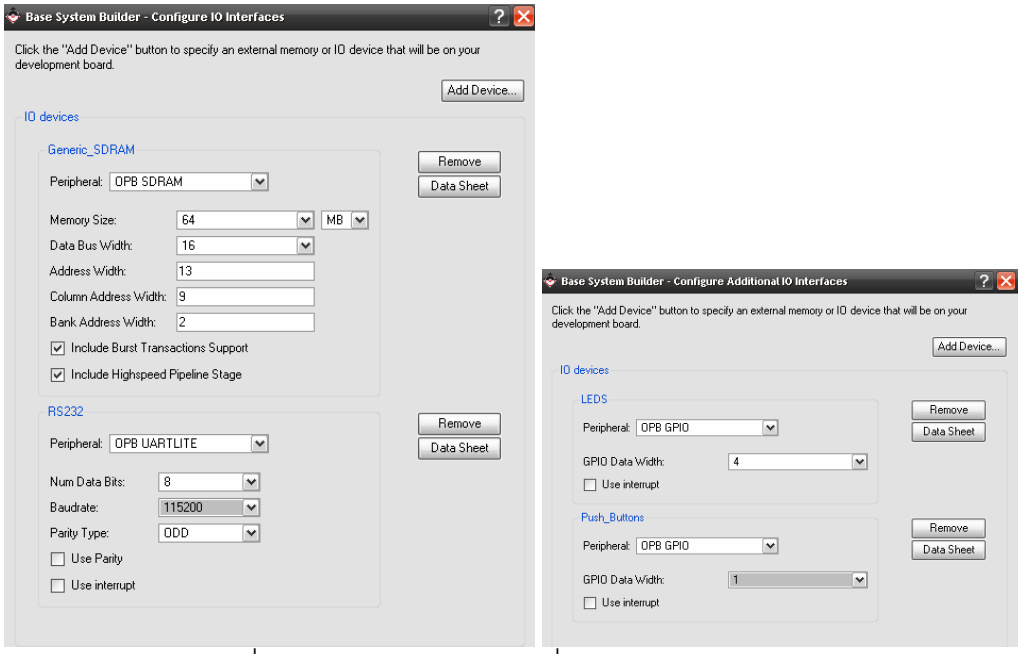
รูปที่ ข.1 การเลือกอุปกรณ์สำหรับ MicroBlaze

6. ที่หน้า Configure MicroBlaze ให้เลือกดังรูปที่ ข.2 แล้วกด Next



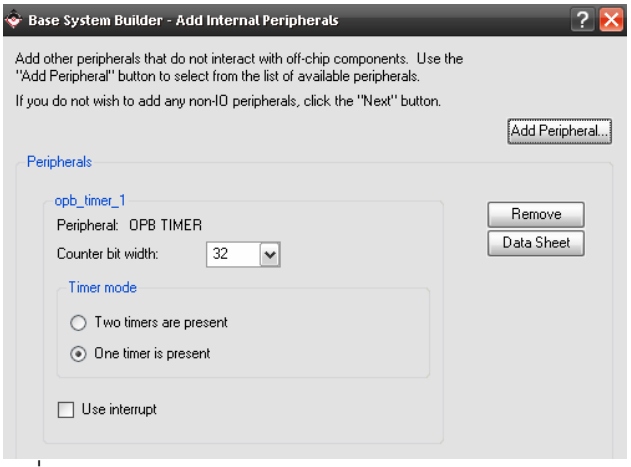
รูปที่ ข.2 การกำหนดคุณลักษณะของ MicroBlaze

7. ที่หน้า Configure IO Interfaces ให้เลือก SDRAM: Gerneric_SDRAM, UART: RS232, GPIO: LEDS และ GPIO: Push_Buttons ดังรูปที่ ข.3 แล้วกด Next



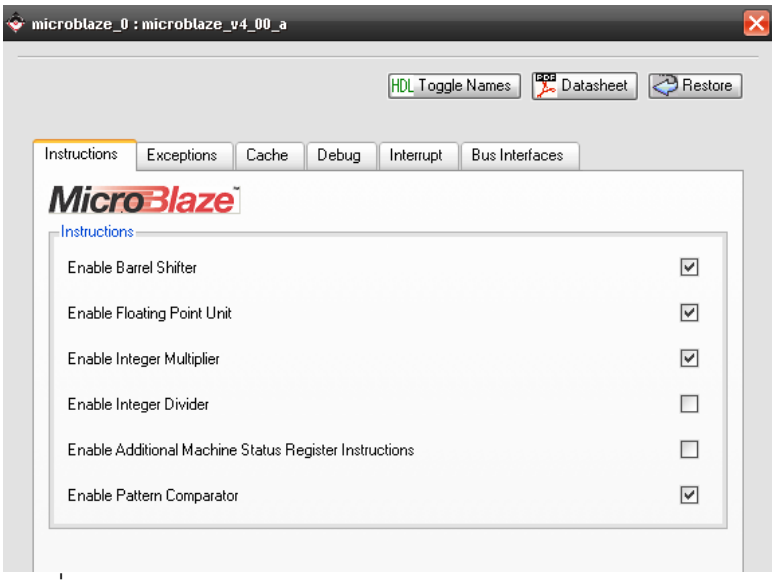
รูปที่ ข.3 การกำหนดอุปกรณ์เชื่อมต่อกับ MicroBlaze

8. ที่หน้า Add Internal Peripherals ให้เลือก OPB TIMER ดังรูปที่ ข.4 แล้วกด Next



รูปที่ ข.4 การกำหนดอุปกรณ์ Timer สำหรับ MicroBlaze

9. ที่หน้า Sorftware Setup กด Next, Next, Next, Generate, Finish
10. เมื่อเข้ามาที่หน้า System Assembly View คลิ๊กขวาที่ microblaze_0 เลือก Configure IP ดังรูปที่ ข.5



รูปที่ ข.5 การกำหนดความสามารถในการทำงานของ MicroBlaze

11. คลิ๊กขวาที่ Generic_SDRAM เลือก Configure IP เลือกที่ SDRAM Parameters เปลี่ยนค่า TRAS = 44000, TRC = 66000, TRFC = 66000 (ค่าต่างๆ ขึ้นอยู่กับอุปกรณ์ที่ใช้)

12. กำหนด UCF File ดังนี้ (การกำหนดอาจแตกต่างกัน ขึ้นอยู่กับอุปกรณ์ที่ใช้)

```
 Net sys_clk_pin LOC=P183;
 Net sys_rst_pin LOC=P194;
## System level constraints
Net sys_clk_pin TNM_NET = sys_clk_pin;
TIMESPEC TS_sys_clk_pin = PERIOD sys_clk_pin 10000 ps;
Net sys_rst_pin TIG;

## IO Devices constraints

#### Module RS232 constraints

# Net fpga_0_RS232_req_to_send_pin LOC=;
 Net fpga_0_RS232_RX_pin LOC=P181;
 Net fpga_0_RS232_TX_pin LOC=P180;

#### Module LEDS constraints

 Net fpga_0_LEDS_GPIO_d_out_pin<0> LOC=p163;
 Net fpga_0_LEDS_GPIO_d_out_pin<1> LOC=p162;
 Net fpga_0_LEDS_GPIO_d_out_pin<2> LOC=p93;
 Net fpga_0_LEDS_GPIO_d_out_pin<3> LOC=p94;

#### Module Push_Buttons constraints

 Net fpga_0_Push_Buttons_GPIO_in_pin<0> LOC=p184;

#### Module Generic_SDRAM constraints

 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<0> LOC=P31;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<1> LOC=P30;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<2> LOC=P29;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<3> LOC=P28;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<4> LOC=P25;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<5> LOC=P24;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<6> LOC=P23;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<7> LOC=P22;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<8> LOC=P19;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<9> LOC=P18;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<10> LOC=P16;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<11> LOC=P15;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<12> LOC=P12;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<13> LOC=P11;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<14> LOC=P9;
 Net fpga_0_Generic_SDRAM_SDRAM_DQ_pin<15> LOC=P8;
 Net fpga_0_Generic_SDRAM_SDRAM_Addr_pin<0> LOC=P36;
 Net fpga_0_Generic_SDRAM_SDRAM_Addr_pin<1> LOC=P39;
 Net fpga_0_Generic_SDRAM_SDRAM_Addr_pin<2> LOC=P202;
 Net fpga_0_Generic_SDRAM_SDRAM_Addr_pin<3> LOC=P40;
 Net fpga_0_Generic_SDRAM_SDRAM_Addr_pin<4> LOC=P41;
 Net fpga_0_Generic_SDRAM_SDRAM_Addr_pin<5> LOC=P42;
 Net fpga_0_Generic_SDRAM_SDRAM_Addr_pin<6> LOC=P45;
 Net fpga_0_Generic_SDRAM_SDRAM_Addr_pin<7> LOC=P47;
 Net fpga_0_Generic_SDRAM_SDRAM_Addr_pin<8> LOC=P48;
 Net fpga_0_Generic_SDRAM_SDRAM_Addr_pin<9> LOC=P196;
 Net fpga_0_Generic_SDRAM_SDRAM_Addr_pin<10> LOC=P197;
 Net fpga_0_Generic_SDRAM_SDRAM_Addr_pin<11> LOC=P199;
 Net fpga_0_Generic_SDRAM_SDRAM_Addr_pin<12> LOC=P200;
 Net fpga_0_Generic_SDRAM_SDRAM_DQM_pin<1> LOC=P5;
 Net fpga_0_Generic_SDRAM_SDRAM_DQM_pin<0> LOC=P33;
 Net fpga_0_Generic_SDRAM_SDRAM_WEn_pin LOC=P4;
 Net fpga_0_Generic_SDRAM_SDRAM_CKE_pin LOC=P35;
 Net fpga_0_Generic_SDRAM_SDRAM_CSn_pin LOC=P206;
 Net fpga_0_Generic_SDRAM_SDRAM_CASn_pin LOC=P3;
 Net fpga_0_Generic_SDRAM_SDRAM_RASn_pin LOC=P2;
 Net fpga_0_Generic_SDRAM_SDRAM_Clk_pin LOC=P34;
 Net fpga_0_Generic_SDRAM_SDRAM_BankAddr_pin<1> LOC=P205;
 Net fpga_0_Generic_SDRAM_SDRAM_BankAddr_pin<0> LOC=P203;
```

13. สร้าง Custom Peripheral (Look-Up Table) โดย

13.1 ไปที่ Hardware, Create and Import Peripheral Wizard กด Next

13.2 เลือก Create template for a new peripheral กด next

13.3 เลือก To an XPS project กด Next

13.4 ตั้งชื่อ table กด Next

13.5 เลือก On-chip Peripheral Bus (OPB) กด Next

13.6 เลือก User logic address range support กด Next

13.7 เลือก Number of user address ranges: 1,

เลือก Data width of each address range 32 bit กด Next

13.8 ที่หน้า IP Interconnect กด Next, Next, Next, Finish

13.9 แก้ไขไฟล์ …/pcores/…/hdl/vhdl/user_logic.vhd ตั้งแต่บรรทัด *architecture IMP of user_logic is* ดังนี้

```
architecture IMP of user_logic is

  type DO_TYPE is array (0 to C_NUM_ADDR_RNG-1) of std_logic_vector(0 to C_MAX_AR_DWIDTH-1);
  signal ar_data_out                : DO_TYPE;
  signal ar_address                 : std_logic_vector(0 to 11);
  signal ar_select                  : std_logic_vector(0 to 0);
  signal ar_read_enable             : std_logic;
  signal ar_read_ack_dly1           : std_logic;
  signal ar_read_ack                : std_logic;
  signal ar_write_ack               : std_logic;

begin

  ar_select      <= Bus2IP_ArCS;
  ar_read_enable <= ( Bus2IP_ArCS(0) ) and Bus2IP_RNW;
  ar_read_ack    <= ar_read_ack_dly1;
  ar_write_ack   <= ( Bus2IP_ArCS(0) ) and not(Bus2IP_RNW);
  ar_address     <= Bus2IP_Addr(C_AWIDTH-14 to C_AWIDTH-3);

  BRAM_RD_ACK_PROC : process( Bus2IP_Clk ) is
  begin
    if ( Bus2IP_Clk'event and Bus2IP_Clk = '1' ) then
      if ( Bus2IP_Reset = '1' ) then
        ar_read_ack_dly1 <= '0';
      else
        ar_read_ack_dly1 <= ar_read_enable;
      end if;
    end if;
  end process BRAM_RD_ACK_PROC;

  READ_ADDR : process( ar_address ) is
  begin
    case ar_address is
                        -- hamming window
                        when x"000" => ar_data_out(0) <= x"3da3d70a";     when x"001" => ar_data_out(0) <= x"3da4203f";
                        when x"002" => ar_data_out(0) <= x"3da4fbd3";     when x"003" => ar_data_out(0) <= x"3da669a4";
                        when x"004" => ar_data_out(0) <= x"3da86978";     when x"005" => ar_data_out(0) <= x"3daafb01";
                        when x"006" => ar_data_out(0) <= x"3dae1dd8";     when x"007" => ar_data_out(0) <= x"3db1d180";
                        when x"008" => ar_data_out(0) <= x"3db61567";     when x"009" => ar_data_out(0) <= x"3dbae8e1";
                        when x"00a" => ar_data_out(0) <= x"3dc04b30";     when x"00b" => ar_data_out(0) <= x"3dc63b7d";
                        when x"00c" => ar_data_out(0) <= x"3dccb8dc";     when x"00d" => ar_data_out(0) <= x"3dd3c24b";
                        when x"00e" => ar_data_out(0) <= x"3ddb56b1";     when x"00f" => ar_data_out(0) <= x"3de374e1";
```

```
when x"010" => ar_data_out(0) <= x"3dec1b99";    when x"011" => ar_data_out(0) <= x"3df5497f";
when x"012" => ar_data_out(0) <= x"3dfefd27";    when x"013" => ar_data_out(0) <= x"3e049a87";
when x"014" => ar_data_out(0) <= x"3e09f7d0";    when x"015" => ar_data_out(0) <= x"3e0f9597";
when x"016" => ar_data_out(0) <= x"3e1572ff";    when x"017" => ar_data_out(0) <= x"3e1b8f1c";
when x"018" => ar_data_out(0) <= x"3e21e8fd";    when x"019" => ar_data_out(0) <= x"3e287fa5";
when x"01a" => ar_data_out(0) <= x"3e2f520d";    when x"01b" => ar_data_out(0) <= x"3e365f25";
when x"01c" => ar_data_out(0) <= x"3e3da5d7";    when x"01d" => ar_data_out(0) <= x"3e4524ff";
when x"01e" => ar_data_out(0) <= x"3e4cdb74";    when x"01f" => ar_data_out(0) <= x"3e54c803";
when x"020" => ar_data_out(0) <= x"3e5ce970";    when x"021" => ar_data_out(0) <= x"3e653e79";
when x"022" => ar_data_out(0) <= x"3e6dc5d1";    when x"023" => ar_data_out(0) <= x"3e767e25";
when x"024" => ar_data_out(0) <= x"3e7f661b";    when x"025" => ar_data_out(0) <= x"3e843e28";
when x"026" => ar_data_out(0) <= x"3e88dfad";    when x"027" => ar_data_out(0) <= x"3e8d96e4";
when x"028" => ar_data_out(0) <= x"3e926313";    when x"029" => ar_data_out(0) <= x"3e974379";
when x"02a" => ar_data_out(0) <= x"3e9c3755";    when x"02b" => ar_data_out(0) <= x"3ea13de2";
when x"02c" => ar_data_out(0) <= x"3ea65658";    when x"02d" => ar_data_out(0) <= x"3eab7fec";
when x"02e" => ar_data_out(0) <= x"3eb0b9d2";    when x"02f" => ar_data_out(0) <= x"3eb60338";
when x"030" => ar_data_out(0) <= x"3ebb5b4c";    when x"031" => ar_data_out(0) <= x"3ec0c13b";
when x"032" => ar_data_out(0) <= x"3ec6342c";    when x"033" => ar_data_out(0) <= x"3ecbb348";
when x"034" => ar_data_out(0) <= x"3ed13db4";    when x"035" => ar_data_out(0) <= x"3ed6d292";
when x"036" => ar_data_out(0) <= x"3edc7106";    when x"037" => ar_data_out(0) <= x"3ee21830";
when x"038" => ar_data_out(0) <= x"3ee7c72e";    when x"039" => ar_data_out(0) <= x"3eed7d1f";
when x"03a" => ar_data_out(0) <= x"3ef3391f";    when x"03b" => ar_data_out(0) <= x"3ef8fa4a";
when x"03c" => ar_data_out(0) <= x"3efebfbc";    when x"03d" => ar_data_out(0) <= x"3f024447";
when x"03e" => ar_data_out(0) <= x"3f0529ed";    when x"03f" => ar_data_out(0) <= x"3f08105d";
when x"040" => ar_data_out(0) <= x"3f0af724";    when x"041" => ar_data_out(0) <= x"3f0dddce";
when x"042" => ar_data_out(0) <= x"3f10c3e8";    when x"043" => ar_data_out(0) <= x"3f13a8fe";
when x"044" => ar_data_out(0) <= x"3f168c9d";    when x"045" => ar_data_out(0) <= x"3f196e53";
when x"046" => ar_data_out(0) <= x"3f1c4dac";    when x"047" => ar_data_out(0) <= x"3f1f2a36";
when x"048" => ar_data_out(0) <= x"3f220380";    when x"049" => ar_data_out(0) <= x"3f24d918";
when x"04a" => ar_data_out(0) <= x"3f27aa8d";    when x"04b" => ar_data_out(0) <= x"3f2a7770";
when x"04c" => ar_data_out(0) <= x"3f2d3f50";    when x"04d" => ar_data_out(0) <= x"3f3001c0";
when x"04e" => ar_data_out(0) <= x"3f32be51";    when x"04f" => ar_data_out(0) <= x"3f357497";
when x"050" => ar_data_out(0) <= x"3f382425";    when x"051" => ar_data_out(0) <= x"3f3acc91";
when x"052" => ar_data_out(0) <= x"3f3d6d70";    when x"053" => ar_data_out(0) <= x"3f40065c";
when x"054" => ar_data_out(0) <= x"3f4296eb";    when x"055" => ar_data_out(0) <= x"3f451eb8";
when x"056" => ar_data_out(0) <= x"3f479d5f";    when x"057" => ar_data_out(0) <= x"3f4a127c";
when x"058" => ar_data_out(0) <= x"3f4c7dad";    when x"059" => ar_data_out(0) <= x"3f4ede92";
when x"05a" => ar_data_out(0) <= x"3f5134cd";    when x"05b" => ar_data_out(0) <= x"3f538000";
when x"05c" => ar_data_out(0) <= x"3f55bfd1";    when x"05d" => ar_data_out(0) <= x"3f57f3e5";
when x"05e" => ar_data_out(0) <= x"3f5a1be5";    when x"05f" => ar_data_out(0) <= x"3f5c377c";
when x"060" => ar_data_out(0) <= x"3f5e4655";    when x"061" => ar_data_out(0) <= x"3f60481e";
when x"062" => ar_data_out(0) <= x"3f623c89";    when x"063" => ar_data_out(0) <= x"3f642346";
when x"064" => ar_data_out(0) <= x"3f65fc0a";    when x"065" => ar_data_out(0) <= x"3f67c68c";
when x"066" => ar_data_out(0) <= x"3f698285";    when x"067" => ar_data_out(0) <= x"3f6b2faf";
when x"068" => ar_data_out(0) <= x"3f6ccdc8";    when x"069" => ar_data_out(0) <= x"3f6e5c90";
when x"06a" => ar_data_out(0) <= x"3f6fdbc8";    when x"06b" => ar_data_out(0) <= x"3f714b35";
when x"06c" => ar_data_out(0) <= x"3f72aa9e";    when x"06d" => ar_data_out(0) <= x"3f73f9cc";
when x"06e" => ar_data_out(0) <= x"3f75388b";    when x"06f" => ar_data_out(0) <= x"3f7666aa";
when x"070" => ar_data_out(0) <= x"3f7783f9";    when x"071" => ar_data_out(0) <= x"3f78904d";
when x"072" => ar_data_out(0) <= x"3f798b7b";    when x"073" => ar_data_out(0) <= x"3f7a755d";
when x"074" => ar_data_out(0) <= x"3f7b4dce";    when x"075" => ar_data_out(0) <= x"3f7c14ad";
when x"076" => ar_data_out(0) <= x"3f7cc9da";    when x"077" => ar_data_out(0) <= x"3f7d6d3a";
when x"078" => ar_data_out(0) <= x"3f7dfeb3";    when x"079" => ar_data_out(0) <= x"3f7e7e2e";
when x"07a" => ar_data_out(0) <= x"3f7eeb98";    when x"07b" => ar_data_out(0) <= x"3f7f46e0";
when x"07c" => ar_data_out(0) <= x"3f7f8ff7";    when x"07d" => ar_data_out(0) <= x"3f7fc6d2";
when x"07e" => ar_data_out(0) <= x"3f7feb69";    when x"07f" => ar_data_out(0) <= x"3f7ffdb6";
-- cos
when x"080" => ar_data_out(0) <= x"3f800000";    when x"081" => ar_data_out(0) <= x"3f7fec43";
when x"082" => ar_data_out(0) <= x"3f7fb10f";    when x"083" => ar_data_out(0) <= x"3f7f4e6d";
when x"084" => ar_data_out(0) <= x"3f7ec46d";    when x"085" => ar_data_out(0) <= x"3f7e1324";
when x"086" => ar_data_out(0) <= x"3f7d3aac";    when x"087" => ar_data_out(0) <= x"3f7c3b28";
when x"088" => ar_data_out(0) <= x"3f7b14be";    when x"089" => ar_data_out(0) <= x"3f79c79d";
when x"08a" => ar_data_out(0) <= x"3f7853f8";    when x"08b" => ar_data_out(0) <= x"3f76ba07";
when x"08c" => ar_data_out(0) <= x"3f74fa0b";    when x"08d" => ar_data_out(0) <= x"3f731447";
when x"08e" => ar_data_out(0) <= x"3f710908";    when x"08f" => ar_data_out(0) <= x"3f6ed89e";
when x"090" => ar_data_out(0) <= x"3f6c835e";    when x"091" => ar_data_out(0) <= x"3f6a09a7";
when x"092" => ar_data_out(0) <= x"3f676bd8";    when x"093" => ar_data_out(0) <= x"3f64aa59";
when x"094" => ar_data_out(0) <= x"3f61c598";    when x"095" => ar_data_out(0) <= x"3f5ebe05";
when x"096" => ar_data_out(0) <= x"3f5b941a";    when x"097" => ar_data_out(0) <= x"3f584853";
when x"098" => ar_data_out(0) <= x"3f54db31";    when x"099" => ar_data_out(0) <= x"3f514d3d";
when x"09a" => ar_data_out(0) <= x"3f4d9f02";    when x"09b" => ar_data_out(0) <= x"3f49d112";
```

```
when x"09c" => ar_data_out(0) <= x"3f45e403";        when x"09d" => ar_data_out(0) <= x"3f41d870";
when x"09e" => ar_data_out(0) <= x"3f3daef9";        when x"09f" => ar_data_out(0) <= x"3f396842";
when x"0a0" => ar_data_out(0) <= x"3f3504f3";        when x"0a1" => ar_data_out(0) <= x"3f3085bb";
when x"0a2" => ar_data_out(0) <= x"3f2beb4a";        when x"0a3" => ar_data_out(0) <= x"3f273656";
when x"0a4" => ar_data_out(0) <= x"3f226799";        when x"0a5" => ar_data_out(0) <= x"3f1d7fd1";
when x"0a6" => ar_data_out(0) <= x"3f187fc0";        when x"0a7" => ar_data_out(0) <= x"3f13682a";
when x"0a8" => ar_data_out(0) <= x"3f0e39da";        when x"0a9" => ar_data_out(0) <= x"3f08f59b";
when x"0aa" => ar_data_out(0) <= x"3f039c3d";        when x"0ab" => ar_data_out(0) <= x"3efc5d27";
when x"0ac" => ar_data_out(0) <= x"3ef15aea";        when x"0ad" => ar_data_out(0) <= x"3ee63375";
when x"0ae" => ar_data_out(0) <= x"3edae880";        when x"0af" => ar_data_out(0) <= x"3ecf7bca";
when x"0b0" => ar_data_out(0) <= x"3ec3ef15";        when x"0b1" => ar_data_out(0) <= x"3eb8442a";
when x"0b2" => ar_data_out(0) <= x"3eac7cd4";        when x"0b3" => ar_data_out(0) <= x"3ea09ae5";
when x"0b4" => ar_data_out(0) <= x"3e94a031";        when x"0b5" => ar_data_out(0) <= x"3e888e93";
when x"0b6" => ar_data_out(0) <= x"3e78cfcc";        when x"0b7" => ar_data_out(0) <= x"3e605c13";
when x"0b8" => ar_data_out(0) <= x"3e47c5c2";        when x"0b9" => ar_data_out(0) <= x"3e2f10a2";
when x"0ba" => ar_data_out(0) <= x"3e164083";        when x"0bb" => ar_data_out(0) <= x"3dfab273";
when x"0bc" => ar_data_out(0) <= x"3dc8bd36";        when x"0bd" => ar_data_out(0) <= x"3d96a905";
when x"0be" => ar_data_out(0) <= x"3d48fb30";        when x"0bf" => ar_data_out(0) <= x"3cc90ab0";
-- sin
when x"0c0" => ar_data_out(0) <= x"248d3132";
--fb1 (10)
when x"0c1" => ar_data_out(0) <= x"3ec7dd13";        when x"0c2" => ar_data_out(0) <= x"3f71325f";
when x"0c3" => ar_data_out(0) <= x"3fbf99e9";        when x"0c4" => ar_data_out(0) <= x"3fe869b2";
when x"0c5" => ar_data_out(0) <= x"3fe71686";        when x"0c6" => ar_data_out(0) <= x"3fc16c64";
when x"0c7" => ar_data_out(0) <= x"3f88170b";        when x"0c8" => ar_data_out(0) <= x"3f1d0a2b";
when x"0c9" => ar_data_out(0) <= x"3e925a2f";        when x"0ca" => ar_data_out(0) <= x"3e1a10e4";
--fb2 (13)
when x"0cb" => ar_data_out(0) <= x"3e474bd1";        when x"0cc" => ar_data_out(0) <= x"3efa4e70";
when x"0cd" => ar_data_out(0) <= x"3f6fd1e9";        when x"0ce" => ar_data_out(0) <= x"3fb17aea";
when x"0cf" => ar_data_out(0) <= x"3fdb6974";        when x"0d0" => ar_data_out(0) <= x"3fecbde4";
when x"0d1" => ar_data_out(0) <= x"3fe3fa5e";        when x"0d2" => ar_data_out(0) <= x"3fc5b5a6";
when x"0d3" => ar_data_out(0) <= x"3f9a411e";        when x"0d4" => ar_data_out(0) <= x"3f565c1a";
when x"0d5" => ar_data_out(0) <= x"3f027d3b";        when x"0d6" => ar_data_out(0) <= x"3e8c1804";
when x"0d7" => ar_data_out(0) <= x"3e220ecc";
--fb3 (16)
when x"0d8" => ar_data_out(0) <= x"3e602d11";        when x"0d9" => ar_data_out(0) <= x"3ee92968";
when x"0da" => ar_data_out(0) <= x"3f4b7dc5";        when x"0db" => ar_data_out(0) <= x"3f94d1f3";
when x"0dc" => ar_data_out(0) <= x"3fbec163";        when x"0dd" => ar_data_out(0) <= x"3fdcf9ff";
when x"0de" => ar_data_out(0) <= x"3febbe26";        when x"0df" => ar_data_out(0) <= x"3fea2dc1";
when x"0e0" => ar_data_out(0) <= x"3fd9dc3a";        when x"0e1" => ar_data_out(0) <= x"3fbe24ae";
when x"0e2" => ar_data_out(0) <= x"3f9b6704";        when x"0e3" => ar_data_out(0) <= x"3f6c9d5a";
when x"0e4" => ar_data_out(0) <= x"3f2670b3";        when x"0e5" => ar_data_out(0) <= x"3ed706ba";
when x"0e6" => ar_data_out(0) <= x"3e821602";        when x"0e7" => ar_data_out(0) <= x"3e2809f3";
--fb4 (21)
when x"0e8" => ar_data_out(0) <= x"3e2e91f8";        when x"0e9" => ar_data_out(0) <= x"3e988f17";
when x"0ea" => ar_data_out(0) <= x"3f03b6a4";        when x"0eb" => ar_data_out(0) <= x"3f4931f8";
when x"0ec" => ar_data_out(0) <= x"3f89b153";        when x"0ed" => ar_data_out(0) <= x"3facc7a7";
when x"0ee" => ar_data_out(0) <= x"3fca3e52";        when x"0ef" => ar_data_out(0) <= x"3fdf7a7f";
when x"0f0" => ar_data_out(0) <= x"3feafec2";        when x"0f1" => ar_data_out(0) <= x"3fec638b";
when x"0f2" => ar_data_out(0) <= x"3fe43651";        when x"0f3" => ar_data_out(0) <= x"3fd3c859";
when x"0f4" => ar_data_out(0) <= x"3fbcf568";        when x"0f5" => ar_data_out(0) <= x"3fa1e937";
when x"0f6" => ar_data_out(0) <= x"3f84e8ed";        when x"0f7" => ar_data_out(0) <= x"3f50485e";
when x"0f8" => ar_data_out(0) <= x"3f1b1fbc";        when x"0f9" => ar_data_out(0) <= x"3edb2d7d";
when x"0fa" => ar_data_out(0) <= x"3e9440c8";        when x"0fb" => ar_data_out(0) <= x"3e4957cc";
when x"0fc" => ar_data_out(0) <= x"3e1be4ba";
--fb5 (27)
when x"0fd" => ar_data_out(0) <= x"3e1ce3a8";        when x"0fe" => ar_data_out(0) <= x"3e5e4d76";
when x"0ff" => ar_data_out(0) <= x"3eb0de9b";        when x"100" => ar_data_out(0) <= x"3f061531";
when x"101" => ar_data_out(0) <= x"3f3c2d92";        when x"102" => ar_data_out(0) <= x"3f762e25";
when x"103" => ar_data_out(0) <= x"3f97dbd1";        when x"104" => ar_data_out(0) <= x"3fb27022";
when x"105" => ar_data_out(0) <= x"3fc934a1";        when x"106" => ar_data_out(0) <= x"3fdaefce";
when x"107" => ar_data_out(0) <= x"3fe6d506";        when x"108" => ar_data_out(0) <= x"3fec8369";
when x"109" => ar_data_out(0) <= x"3febfe44";        when x"10a" => ar_data_out(0) <= x"3fe5a0be";
when x"10b" => ar_data_out(0) <= x"3fda0e52";        when x"10c" => ar_data_out(0) <= x"3fca21a0";
when x"10d" => ar_data_out(0) <= x"3fb6dace";        when x"10e" => ar_data_out(0) <= x"3fa14e8b";
when x"10f" => ar_data_out(0) <= x"3f8a9671";        when x"110" => ar_data_out(0) <= x"3f6786be";
when x"111" => ar_data_out(0) <= x"3f3ba439";        when x"112" => ar_data_out(0) <= x"3f1344e9";
when x"113" => ar_data_out(0) <= x"3edfc313";        when x"114" => ar_data_out(0) <= x"3ea547bc";
when x"115" => ar_data_out(0) <= x"3e718b85";        when x"116" => ar_data_out(0) <= x"3e368b1f";
when x"117" => ar_data_out(0) <= x"3e1a60fa";
--fb6 (34)
```

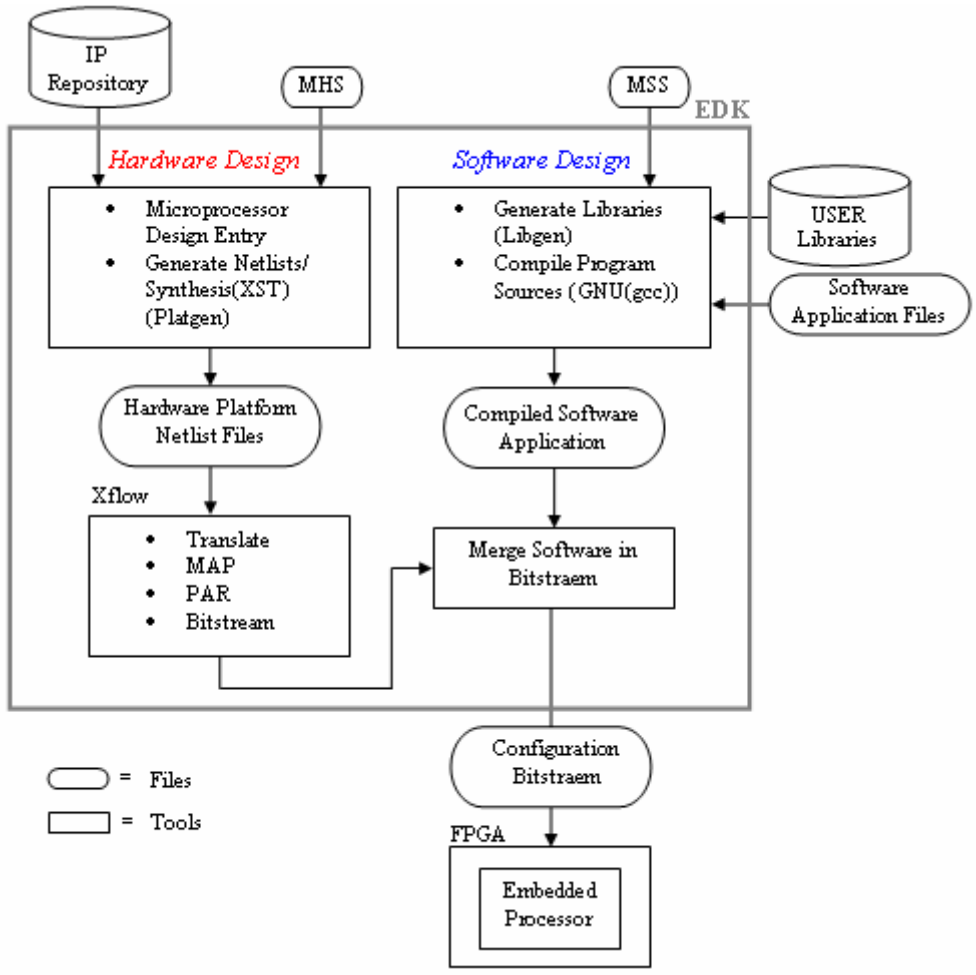
```
                when x"118" => ar_data_out(0) <= x"3e200de1";       when x"119" => ar_data_out(0) <= x"3e52fa10";
                when x"11a" => ar_data_out(0) <= x"3e97c6b7";       when x"11b" => ar_data_out(0) <= x"3ed77980";
                when x"11c" => ar_data_out(0) <= x"3f124a64";       when x"11d" => ar_data_out(0) <= x"3f3d62ea";
                when x"11e" => ar_data_out(0) <= x"3f6ad31e";       when x"11f" => ar_data_out(0) <= x"3f8c3ca1";
                when x"120" => ar_data_out(0) <= x"3fa22de3";       when x"121" => ar_data_out(0) <= x"3fb65d8c";
                when x"122" => ar_data_out(0) <= x"3fc80f3b";       when x"123" => ar_data_out(0) <= x"3fd6ae11";
                when x"124" => ar_data_out(0) <= x"3fe1ce8f";       when x"125" => ar_data_out(0) <= x"3fe92e9c";
                when x"126" => ar_data_out(0) <= x"3fecb3e1";       when x"127" => ar_data_out(0) <= x"3fec68dd";
                when x"128" => ar_data_out(0) <= x"3fe878fa";       when x"129" => ar_data_out(0) <= x"3fe12be9";
                when x"12a" => ar_data_out(0) <= x"3fd6e097";       when x"12b" => ar_data_out(0) <= x"3fca07e7";
                when x"12c" => ar_data_out(0) <= x"3fbb1f80";       when x"12d" => ar_data_out(0) <= x"3faaacb9";
                when x"12e" => ar_data_out(0) <= x"3f9937ef";       when x"12f" => ar_data_out(0) <= x"3f874838";
                when x"130" => ar_data_out(0) <= x"3f6abf52";       when x"131" => ar_data_out(0) <= x"3f47f036";
                when x"132" => ar_data_out(0) <= x"3f270127";       when x"133" => ar_data_out(0) <= x"3f08b66e";
                when x"134" => ar_data_out(0) <= x"3edb6f3b";       when x"135" => ar_data_out(0) <= x"3ead1b60";
                when x"136" => ar_data_out(0) <= x"3e87443c";       when x"137" => ar_data_out(0) <= x"3e54fb65";
                when x"138" => ar_data_out(0) <= x"3e2e3c27";       when x"139" => ar_data_out(0) <= x"3e1a81e9";
                --fb7 (43)
                when x"13a" => ar_data_out(0) <= x"3e1cb919";       when x"13b" => ar_data_out(0) <= x"3e3c3832";
                when x"13c" => ar_data_out(0) <= x"3e76a0b7";       when x"13d" => ar_data_out(0) <= x"3ea47da6";
                when x"13e" => ar_data_out(0) <= x"3ed7e063";       when x"13f" => ar_data_out(0) <= x"3f09c101";
                when x"140" => ar_data_out(0) <= x"3f2aa68e";       when x"141" => ar_data_out(0) <= x"3f4d9023";
                when x"142" => ar_data_out(0) <= x"3f716f8f";       when x"143" => ar_data_out(0) <= x"3f8aa057";
                when x"144" => ar_data_out(0) <= x"3f9c07e5";       when x"145" => ar_data_out(0) <= x"3fac7f6c";
                when x"146" => ar_data_out(0) <= x"3fbba4c9";       when x"147" => ar_data_out(0) <= x"3fc92431";
                when x"148" => ar_data_out(0) <= x"3fd4b928";       when x"149" => ar_data_out(0) <= x"3fde2ef1";
                when x"14a" => ar_data_out(0) <= x"3fe56093";       when x"14b" => ar_data_out(0) <= x"3fea387b";
                when x"14c" => ar_data_out(0) <= x"3fecafc3";       when x"14d" => ar_data_out(0) <= x"3feccd3a";
                when x"14e" => ar_data_out(0) <= x"3feaa430";       when x"14f" => ar_data_out(0) <= x"3fe6531b";
                when x"150" => ar_data_out(0) <= x"3fe0021a";       when x"151" => ar_data_out(0) <= x"3fd7e16e";
                when x"152" => ar_data_out(0) <= x"3fce27e3";       when x"153" => ar_data_out(0) <= x"3fc31147";
                when x"154" => ar_data_out(0) <= x"3fb6dce0";       when x"155" => ar_data_out(0) <= x"3fa9cc00";
                when x"156" => ar_data_out(0) <= x"3f9c20a3";       when x"157" => ar_data_out(0) <= x"3f8e1c36";
                when x"158" => ar_data_out(0) <= x"3f7ffce6";       when x"159" => ar_data_out(0) <= x"3f6408c3";
                when x"15a" => ar_data_out(0) <= x"3f48cef8";       when x"15b" => ar_data_out(0) <= x"3f2eb9e4";
                when x"15c" => ar_data_out(0) <= x"3f162a29";       when x"15d" => ar_data_out(0) <= x"3efeeb77";
                when x"15e" => ar_data_out(0) <= x"3ed5ce6f";       when x"15f" => ar_data_out(0) <= x"3eb17aea";
                when x"160" => ar_data_out(0) <= x"3e9256a3";       when x"161" => ar_data_out(0) <= x"3e715eba";
                when x"162" => ar_data_out(0) <= x"3e497685";       when x"163" => ar_data_out(0) <= x"3e2d3324";
                when x"164" => ar_data_out(0) <= x"3e1ca732";
                --fb8 (55)
                when x"165" => ar_data_out(0) <= x"3e199633";       when x"166" => ar_data_out(0) <= x"3e2ade82";
                when x"167" => ar_data_out(0) <= x"3e4d672b";       when x"168" => ar_data_out(0) <= x"3e7fef30";
                when x"169" => ar_data_out(0) <= x"3ea07a4a";       when x"16a" => ar_data_out(0) <= x"3ec76074";
                when x"16b" => ar_data_out(0) <= x"3ef3bae6";       when x"16c" => ar_data_out(0) <= x"3f124640";
                when x"16d" => ar_data_out(0) <= x"3f2c6801";       when x"16e" => ar_data_out(0) <= x"3f47beba";
                when x"16f" => ar_data_out(0) <= x"3f63c794";       when x"170" => ar_data_out(0) <= x"3f80018d";
                when x"171" => ar_data_out(0) <= x"3f8dfb9e";       when x"172" => ar_data_out(0) <= x"3f9b9884";
                when x"173" => ar_data_out(0) <= x"3fa8a30e";       when x"174" => ar_data_out(0) <= x"3fb4eaeb";
                when x"175" => ar_data_out(0) <= x"3fc04522";       when x"176" => ar_data_out(0) <= x"3fca8c64";
                when x"177" => ar_data_out(0) <= x"3fd3a145";       when x"178" => ar_data_out(0) <= x"3fdb6a57";
                when x"179" => ar_data_out(0) <= x"3fe1d429";       when x"17a" => ar_data_out(0) <= x"3fe6d12f";
                when x"17b" => ar_data_out(0) <= x"3fea599b";       when x"17c" => ar_data_out(0) <= x"3fec6b1a";
                when x"17d" => ar_data_out(0) <= x"3fed0886";       when x"17e" => ar_data_out(0) <= x"3fec3991";
                when x"17f" => ar_data_out(0) <= x"3fea0a5a";       when x"180" => ar_data_out(0) <= x"3fe68b05";
                when x"181" => ar_data_out(0) <= x"3fe1cf43";       when x"182" => ar_data_out(0) <= x"3fdbede5";
                when x"183" => ar_data_out(0) <= x"3fd50059";       when x"184" => ar_data_out(0) <= x"3fcd223d";
                when x"185" => ar_data_out(0) <= x"3fc470e3";       when x"186" => ar_data_out(0) <= x"3fbb0ae0";
                when x"187" => ar_data_out(0) <= x"3fb10f9e";       when x"188" => ar_data_out(0) <= x"3fa69ef4";
                when x"189" => ar_data_out(0) <= x"3f9bd8c0";       when x"18a" => ar_data_out(0) <= x"3f90dc90";
                when x"18b" => ar_data_out(0) <= x"3f85c948";       when x"18c" => ar_data_out(0) <= x"3f7579b3";
                when x"18d" => ar_data_out(0) <= x"3f5fa7f7";       when x"18e" => ar_data_out(0) <= x"3f4a53e0";
                when x"18f" => ar_data_out(0) <= x"3f35b09b";       when x"190" => ar_data_out(0) <= x"3f21edaa";
                when x"191" => ar_data_out(0) <= x"3f0f3696";       when x"192" => ar_data_out(0) <= x"3efb656e";
                when x"193" => ar_data_out(0) <= x"3edb0a10";       when x"194" => ar_data_out(0) <= x"3ebd9817";
                when x"195" => ar_data_out(0) <= x"3ea34372";       when x"196" => ar_data_out(0) <= x"3e8c36f8";
                when x"197" => ar_data_out(0) <= x"3e7128e3";       when x"198" => ar_data_out(0) <= x"3e50e97f";
                when x"199" => ar_data_out(0) <= x"3e37d027";       when x"19a" => ar_data_out(0) <= x"3e25ec6b";
                when x"19b" => ar_data_out(0) <= x"3e1b3dba";     when others => ar_data_out(0) <= (others => '0');
  end case;
end process READ_ADDR;
```

```
IP2BUS_ARDATA_PROC : process( ar_data_out, ar_select ) is
begin
  case ar_select is
    when "1" => IP2Bus_ArData <= ar_data_out(0);
    when others => IP2Bus_ArData <= (others => '0');
  end case;
end process IP2BUS_ARDATA_PROC;

IP2Bus_Ack      <= ar_write_ack or ar_read_ack;
IP2Bus_Error    <= '0';
IP2Bus_Retry    <= '0';
IP2Bus_ToutSup  <= '0';

end IMP;
```

14. ไปที่ IP Catalog, Project Repository คลิ๊กขวาที่ชื่ออุปกรณ์ (table) เลือก Add IP
15. ที่หน้า System Assembly View กดต่ออุปกรณ์ (กดที่รูปวงกลมหน้าชื่ออุปกรณ์ให้เป็นสีทึบ) แล้วคลิ๊กที่ Addresses เลือก size ของ table_0 เป็น 64K แล้วกด Generate Addresses
16. Add Software Application Project แล้วเขียนโปรแกรม *.c



รูปที่ ข.6 แผนผังการออกแบบระบบโดยใช้ EDK