2020.2

# Vivado Design Suite - HLx Editions - 2020.2 Full Product Installation

## Important Information

VivadoR Design Suite 2020.2 is now available
Download Includes

Download Type

- Public access support for the Xilinx® VersalTM Platforms
- Petalinux now a part of Xilinx Unified Installer
- Access Block Design container now to create team-based designs
- Abstract Shell for Dynamic Function eXchange
- 2020.2 Introduces VitisTM HLS for Vivado flows
- Add-on for MATLAB® and Simulink® (Unified Model Composer and System Generator)

We strongly recommend to use the web installers as it reduces download time and saves significant disk space.

Please see **Installer Information** for details.
Last Updated

Answers

Documentation

Support Forums
Vivado Design Suite

HLX Editions (All

Editions)

Full Product

Installation

Nov 24, 2020

2020.x - Vivado

Known Issues

Release Notes

OS Support Update

What's New in

Vivado

Installation and

Licensing

# ultra Large

We strongly recommend using the latest

releases available.

Version

2024.2

2023

2024.1

2023.2

2023.1

Vivado Archive

2023.2

ISE Archive

CAE Vendor Libraries

2022

Archive

2010.1

Sc

A

обо

EXTENSIONS: MARKETPLACE

verilog

SystemMerikig

Verilog

FPGA

# SystemVerilog and Verilog Formatter

Settings

x

Extension: Verilog-HDL/SystemVerilog/Bluespec SystemVerilog

Extension: SystemVerilog and Verilog Formatter

xY

@ext:morh.veriloghdl

Beautify SystemVerilog and Verilog code in VSCode through Verible

**Borja** Penuelas

## Verilog-HDL/SystemVerilog/Bluespec SystemVerilog

Verilog-HDL/SystemVerilog/Bluespec SystemVerilog support for VS C...

**Masahiro Hiramori**

**verilog**
Update

An extension aim at making Verilog programs program easilier.
Gtylcara&Gewinn
109 109
User

Extensions (45)

Verilog configurat... (45)

# Setting

Install

**Verilog HDL**
170K ★5

Verilog HDL Language Support for Visual Studio Code. leafvmaple

# Verilog Testbench

Install

91K ★ 4.5

45 Settings FoundEx Y

Backup and Sync Settings

If enabled, Icarus Verilog will be run at the file location for linting. Else it will be run at workspace folder. Disabled by default.

Verilog > Linting › Iverilog › System Verilog: Standard

Select the standard rule to be used when Icarus Verilog linting for SystemVerilog files.

SystemVerilog2012

Verilog > Linting > Iverilog > Verilog HDL: Standard

Select the standard rule to be used when Icarus Verilog linting for Verilog-HDL files.

Verilog-2005

Verilog › Linting: Linter

Select the verilog linter. Possible values are 'iverilog', 'verilator', 'modelsim', 'xvlog', 'slang' or 'none'.

xvlog

Avlog

## verilog-testbench-instance

iverilog

Truecrab

Install

verilator

modelsim

slang

## Verilog Format

76K 3

none

`verilog`

`format`

Console application for apply format to verilog file. Ericson Joseph

Install

## Verilog Snippet

44K 5

A snippet for verilog

default

Modelsim while linting.

Verilog > Linting › Modelsim: Run **At** File Location

It enabled, Modelsim will be run at the file location for linting. Else it will be run at workspace folder. Disabled by

default.

Verilog > Linting > Modelsim: Work

Add Modelsim work library here.

--column_limit 80 --indentation_spaces 4

ctrl + shift + p visual studio code : formatter 단축키

C:\Xilinx\Vivado\2020.2\bin

"C:/Users/CHK/AppData/Local/Programs/Microsoft VS Code/code.exe" -g [file name]:[line number]

**250227_adder**

↑
←
Q
> 내 PC > 로컬 디스크 (C:) > Working > FPGA_Harman_25_1 > 250227_adder >
2502
X
+

+ 새로 만들기~

> 사진

스캔
N 정렬.
v
≡ 보기.
v

이름

수정한 날짜
유형
크기

250227_adder.cache
2025-02-28 오전 9:26
파일 폴더

> 첨부 파일
250227_adder.hw
2025-02-28 오전 9:26
파일 폴더

250227_adder.ip_user_files

250227_adder.runs
2025-02-28 오전 9:26

2025-02-28 오전 9:26
파일 폴더

파일 폴더
바탕 화면

250227_adder.sim
↓ 다운로드

문서

사진

음악
250227_adder.srcs

- 250227_adder.xpr
4

# To Source

2025-02-28 오전 9:26
파일 폴더

2025-02-28 오전 9:26
파일 폴더

2025-02-27 오후 5:15
Vivado Project File
11KB

1. 안보이면,

2. basys3 압축파일 풀어 놓기

(

```
//assign sum = a ^ b; //assign c = a &
b;
```

# // 게이트 프리미티브:

# Verilog lib 기본

xor (s, a, b); // (출력, 입력, 입력 **1**,

and (c, a, b);

## and

241 c

ᴥ

..)

C

Сти

op

*nor*

t

013/46

*xor*

top :

"full_Adder"

HAI Sum

над

b

c

πJW-(2 out sun

W_C

# Sum Full adder

c

over

flowibe

1011

1

S[3]

502]

SCIJ

S[0]

SC] FAQ Cin

C

FAITH

SCI

S COT

FAO

*c*

Klein

813 FA3 Cin

c

# "Vector, n:o

# 8bit 3.0, 4:1

W[3] b[3]

[[+] 6[+]

aci] bel]

*[[*o] b[o]

+

*の*

LSB

o

자료명

wire (MS12=

(6B)] 2420

3

:. bit size

a

فری

Cin

Vector f

```
input [3:0] a
input [3:0] b,
input cin,
output [3:0] s,
output c
```

-4bit a

```
);
wire [3:0] W_C;
full_adder U_fae
```

0번 bit

.a(a[0]),

bit지 점

.b(b[0]),

.cin(1'b0),

.s(s[0]),

.c(w_c[0])

# Penter key of key

とり

bitt k

2311h

진법

+ -인명

θ

○

10

값

```
.c(w_c[0])


full_adder U_fa1(
.a(a[1]),
.b(b[1]),
7
module fa_4 (
8
input [3:0] a,// 4bit vertor
9
input [3:0] b,
10
11
12
13
);
14
15
16
```

```
input cin,
output [3:0] s,
output c

wire [3:0] w_c;
full_adder U_fa0(
.a(a[0]),
.b(b[0]),
.cin(1'b0),
.sum(s[0]),
);
);
```

```
32
33
34
35
);
36
37
38
39
40
41
42
);
43
endmodule
.cin(w_c[0]),
.sum(s[1]),
.c(w_c[1])

full_adder U_fa2(
.a(a[2]),
.b(b[2]),
.cin(w_c[1]),
.sum(s[2]),
.c(w_c[2])

full_adder U_fa4(
```

.a(a[3]),

.b(b[3]),

.cin(w_c[2]),

.sum(s[3]),

.c(c)

SC3]

SC2]

SCIJ

# C WLC3

SB FA3 Cin

SC

FA2 Cin FA

SCI

C

*c*

LCO SCOT FAO

c

**W[3]** 6[3]

[[+] 6[+]

**aci**] bcl]

a[o] b[o]

11

Vector,

8

o

Abit

3

04:1

자료명)

wire (MGR:(] 2+201

CMS12=

w: bit size

Ų

K+Cin

S [3:0],

# module

추상호

c

Сти

# GC Fu 4 Gu

acs: 0] 6[3:0]

sw

4674

4

b

# ㅌㅌ

# bitstream 생성

X
**Σ**

**Flow Navigator**
K
**ELABORATED DESIGN** - xc7a35topg236-1

Language Tem
**Sources** **Netl** x ? – OG
**Project Summary**
>

#IP Catalog
Q

# FAX

# $sum

# Carry.

✓ IP INTEGRATOR
R fa 4

>

Create Block De
Nets (16)

Leaf Cells (1)

Open Block De
a
U_fa0 (full_adder)

Generate Block
U_fa1 (full_adder)
b

U_fa2 (full_adder)

## 주석

# Bet

#et_property -dict { PACKAGE_PIN W5
IOSTANDARD LVCMOS33 } [get_ports clk]

#create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5} [get_ports clk]

## Switches

# рти вуз

#set_property -dict { PACKAGE PIN V17 #set_property -dict { PACKAGE_PIN V16 #set_property -dict { PACKAGE_PIN W16 #set_property -dict { PACKAGE_PIN W17 #set_property -dict { PACKAGE_PIN W15 #set_property -dict { PACKAGE_PIN V15 #set_property -dict { PACKAGE_PIN W14 #set_property -dict { PACKAGE_PIN W13 #set_property -dict { PACKAGE_PIN V2 #set_property -dict { PACKAGE_PIN T3 #set_property -dict { PACKAGE_PIN T2 #set_property -dict { PACKAGE_PIN R3 #set_property -dict { PACKAGE_PIN W2 #set_property -dict { PACKAGE_PIN U1 #set_property -dict { PACKAGE_PIN T1 #set_property -dict { PACKAGE_PIN R2

# port 012/2

sw
IOSTANDARD LVCMOS 33 } [get_ports sw[0]}] IOSTANDARD LVCMOS33 } [get_ports {sw[1]}] IOSTANDARD LVCMOS33 } [get_ports {sw[2]}] IOSTANDARD LVCMOS33 } [get_ports {sw[3]}] IOSTANDARD LVCMOS33 } [get_ports {sw[4]}] IOSTANDARD LVCMOS33 } [get_ports {sw[5]}] IOSTANDARD LVCMOS33 } [get_ports {sw[6]}] IOSTANDARD LVCMOS33 } [get_ports {sw[7]}] IOSTANDARD LVCMOS33 } [get_ports {sw[8]}] IOSTANDARD LVCMOS33 } [get_ports {sw[9]}] IOSTANDARD LVCMOS33 } [get_ports {sw[10]}] IOSTANDARD LVCMOS33} [get_ports {sw[11]}] IOSTANDARD LVCMOS33 } [get_ports {sw[12]}] IOSTANDARD LVCMOS33 } [get_ports {sw[13]}] IOSTANDARD LVCMOS33 } [get_ports {sw[14]}] IOSTANDARD LVCMOS33 } [get_ports {sw[15]}]
✓ SIMULATION

Run Simulation
U_fa4 (full_adder)

**Properties** ? - ☐ GX

✓ **RTL ANALYSIS**
–

✓ **Open Elabora**

Report Me
Select an object to see

properties
Report DR

Report Noi
# LED

# 16214

3.3V
**Artix-7** U3 LD11
P3 LD12 N3 LD13

QC OK 11

SWOO
wV17

L1
P1 LD14 Li 2015 một
LD15
SW1
w V16

**7-segment**

**ADIGILENT**

16214

3.3V **Display**

SW2

EXILINX

w W16

W4 -w

AN3

LINEAR

8.8.8.8

5536

8116

IC7

8000 8418

**BASYS** 3

I UNIVERSITY PROGRAM

4x1024

V4 w

AN2

SW3

w W17

U4 -w

AN1

SW4

w W15

U2 -w

ANO

LDP

LDE

a

120

22

SW5

a

wV15

comman

SW6

w W14

# 0.0.0.0.7-Seg.

1970
(71) (UT)
183
(72)
(731
(V2)
28
9
MO

ך'

b

SW7

w W13

Wym CA

ww

FNT)

**Slide**

CB

b

[3:0]

a

W6

ww

G

**Switches**

CC

SW8

www V2

U8 ww

(3:0]

CD

V8 w

SW9 ○ w T3

U5 - CE

SW10

MT2

w

# V5 w CF

U7 -w

CG

DP

SW11

w R3

V7 w

# op

SW12

ww2

# SUN F, DP

SW13

w-U1

SW14
w T1

SW15
w R2

f
(D
a
CU

b

g

d

(a)
c
010

ѣ

G

1

Common Cathode

a

b

c -

d-

e

f

g

o oo

dp-

**AAAAAAZ**

**7-segment**

3.3V **Display**

AN2

L1

Li LD15

Common Anode Common

a

b

W4 w V4 w

AN3

U4 w

U2 -w-

o

d

e

f

g

dp

(b)

그림 5.1 7-세그먼트

# BCD 비트 비트수많게

# Decoder.

4

0000

0001

a

888

998

# 888

PAN1

ANO

command

CA

# É...Ó. 7-Seg.

0.0.0.0

FNX)

LLSB

W6

m CB

U8

m CC

V8 m CD

U5 -w CE

V5 m CF

CG

U7

W

DP

V7

MSB

a

レ

LED

Je

**g**

✓ Common Anode

ON

**0**

a

# Anode

ONF

0

し

ナノ

2

dp

3

4

5

6

# dmiths

میری

## DP GFE dcba Codes

11000000 co: 8bit

# 6166006

F9

A4

1011 0000 во

*100/100* / 99

92

الهی

до

# E

## RRD L F BIP 144

## [3:0]

Sum

# FND_Com

MCB

# 4bit me

<

Li LD15

W4

V4 M

U4

LSB LUZ

b

9

A

**d**

*B*

e

C

Cь

f

Al

дь

**g**

dp

F

SE

# Abit

*

## Command

CA

W7

W6

ww

U8 -w

V8 U5

3 88 8 85 8

m CD

V5 w U7 w V7w

CE

CF

DP

# BCD to seg

BCD

# seq [3:0]

AN3

**7-segment**

3.3V **Display**

AN2

PANER

ANO

F

# 8.8.8.8.7-Seg

FNX)

항상 감시하다

대상

항상 대상 이벤트감시

always @(bcd) (begin

(실행한다.

이벤트) Sensitivity lists

end

"

q

a

4bit

S

*4,*

Aobber.

b

# bcc 값이 변화시

module bcdtoseg (

input[3:0] bcd, // [3:0] sum 값

output (reg/ [7:0] seg

);

저장하다. 유지하라.

// always 구문 출력으로 reg type을 가져야 한다.

always @(bcd) begin

if

==Case

case (bcd)

4'h0: seg =
8'hc0;

4'h1: seg = 8'hF9; 4'h2: seg = 8'hA4;

4'h3: seg = 8'hB0;

4'h4: seg =
8'h99;

4'h5. ceo - Q'h92•
sum

*4*

FND_ Control

Bastsey

*Common*

日日日

# LED Display

## Calculator

## Seg

### Common

**Artix-7** U3

U3 LD11

3.3V

P3 LD12

4

FND_ Control

SWO

w- V17

SW1

WV16

a

q

4bt

Adbber.

*の*

C

sum

*4.*

W_Sum

Bastsey.

# seg. 3210

SW2 WW16

N3 LD13 P1 LD14, LD15 L1 _LD15

mô Comm 3.3V

# Seg_comm

**7-segment** 3.3V **Display**

W4

m

V4 w

SW3

ww W17

## seg

日 日 日 日

U4 -w-

SW4 W15

U2 -w

AN3

AN2

AN1

ANO

SW5

WV15

SW6
w W14

# LED

**Slide Switches**
SW8
ww V2

SW9
w T3

SW10
w- T2
SW7 W13

# USB

# Segwz

WZ M CA W6 www

U8 w

## 388

CB

CC

V8 m CD

w

U5 - CE V5 m CF U7 -w CG

DP

## 8.8.8.8.

SW11 R3
V7 w

SW12
www W2

MSB

SW13
w-U1

SW14
wT1

SW15
ww R2

# (Calculator) BTN-/ jbtn

آسا

Sey

K

*common*

Seg_sel IND

Controller

Decoder

seg_comm

v

a

# Wat

b

# #

# Adder.

c

S

W_Sum

# Bastseg.

2x4 Ded

sum

*4*

seg. 32

sey

日日日

FUD Control

4 Seg_Comm

BIN

2x4

4

always

всотобед f

case

Sey

日日日日

LED

Adder

# H/W

# Autor

Abit +

fbit ]2

12-15918

# 1-0921-

sw:16

변경하여라.

*바뀐 부분

스크린 캡춰 .

# 톰 Testbench

# or

## Calculator

FVD Control

2x4 Dea

동

동작사진

4

*2*

BTN=

btn

ती

sum

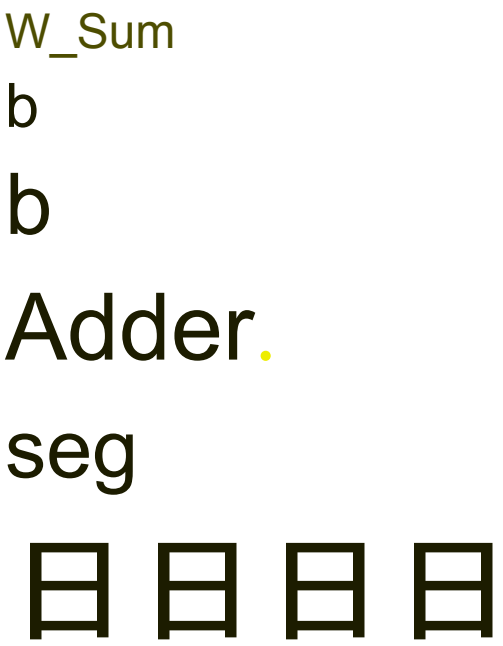
a

Abit s

*4*

Bastseg.

&


W_Sum
b
b
Adder.
seg
日 日 日 日
C
LED