# Design Compiler Local Installation Guide

## 🖥️ System Requirements

### Hardware Requirements

- **CPU**: x86_64 (Intel/AMD 64-bit)
- **RAM**: Minimum 8GB, Recommended 16GB+ for large designs
- **Disk Space**: 15-20GB for full installation
- **Network**: Internet connection for license validation

### Operating System Support

- **Linux**: RHEL 7/8, CentOS 7/8, Ubuntu 18.04/20.04/22.04, SUSE Linux
- **Architecture**: 64-bit only (32-bit not supported in recent versions)

### Software Prerequisites

```bash
```

```
# Required system packages (Ubuntu/Debian)
sudo apt update && sudo apt install -y \
    build-essential \
    gcc \
    g++ \
    make \
    tcsh \
    ksh \
    csh \
    libxext6 \
    libxrender1 \
    libxtst6 \
    libxi6 \
    libxrandr2 \
    libxcursor1 \
    libxinerama1 \
    libfreetype6 \
    fontconfig \
    xauth \
    libc6-dev \
    libstdc++6 \
    libgcc-s1

# For RHEL/CentOS/Fedora
sudo yum install -y \
    gcc \
    gcc-c++ \
    make \
    tcsh \
    ksh \
    libXext \
    libXrender \
    libXtst \
    libXi \
    libXrandr \
    libXcursor \
    libXinerama \
    freetype \
    fontconfig \
    xorg-x11-xauth \
    glibc-devel \
    libstdc++ \
    libgcc
```

## 📦 Pre-Installation Setup

### Step 1: Create User and Directories

bash

```bash
# Create dedicated user for Synopsys tools (optional but recommended)
sudo useradd -m -s /bin/bash synopsys
sudo usermod -aG sudo synopsys  # Add to sudo group if needed

# Create installation directories
sudo mkdir -p /opt/synopsys/syn
sudo mkdir -p /opt/synopsys/licenses
sudo mkdir -p /opt/synopsys/libraries
sudo chown -R synopsys:synopsys /opt/synopsys

# Switch to synopsys user
sudo su - synopsys
```

### Step 2: Download Design Compiler

bash

```bash
# Download from Synopsys SolvNet (requires account)
# Navigate to: https://solvnet.synopsys.com
# Download the installer package (typically named like):
# syn_vX.XX-X_linux64.tar.gz

# Example: syn_vS-2021.06-SP5_linux64.tar.gz
# Place the downloaded file in /tmp or home directory
```

### Step 3: Extract Installation Files

bash

```
# Navigate to download location
cd ~/Downloads  # or wherever you downloaded the file

# Extract the installer
tar -xzf syn_v*_linux64.tar.gz

# This creates a directory like: syn_vS-2021.06-SP5_linux64
cd syn_v*_linux64

# List contents to verify
ls -la
# You should see: installer, installer.properties, setup files
```

## 🔧 Installation Process

### Method 1: GUI Installation (Recommended for Beginners)

```bash
# Start GUI installer
./installer

# The installer will launch a graphical interface
# Follow these steps in the GUI:

# 1. Welcome Screen - Click "Next"
# 2. License Agreement - Accept and click "Next"
# 3. Installation Directory - Set to /opt/synopsys/syn
# 4. Feature Selection:
#    ✓ Design Compiler (required)
#    ✓ Design Vision (GUI - recommended)
#    ✓ DC Ultra (optional - for advanced optimization)
#    ✓ Power Compiler (optional - for power optimization)
#    ✓ Documentation (recommended)
# 5. License Configuration:
#    - Choose "Configure license later" or
#    - Enter license server: 27020@your-license-server.com
# 6. Installation Summary - Review and click "Install"
# 7. Wait for installation to complete (15-30 minutes)
```

### Method 2: Silent Installation (Advanced Users)

```bash
```

```
# Create response file for silent installation
cat > dc_install.properties << EOF
# Design Compiler Silent Installation Properties

# Installation directory
INSTALL_DIR=/opt/synopsys/syn

# Features to install
SELECTED_FEATURES=DesignCompiler,DesignVision,DCUltra,PowerCompiler,Documentation

# License configuration
LICENSE_FILE=27020@your-license-server.com

# Accept license agreement
ACCEPT_LICENSE=true

# Installation mode
INSTALL_MODE=silent
EOF

# Run silent installation
./installer -i silent -f dc_install.properties

# Monitor installation progress
tail -f /tmp/installer*.log
```

### Method 3: Command Line Installation

```
bash

# Interactive command line installation
./installer -i console

# Follow the text-based prompts:
# 1. Press Enter to continue
# 2. Type 'accept' for license agreement
# 3. Enter installation path: /opt/synopsys/syn
# 4. Select features (use space to select, enter to continue)
# 5. Configure license server
# 6. Confirm installation
```

## 🔐 License Configuration

## Step 1: Obtain License Information

```bash
# Get your system's host ID (needed for license)
# Method 1: MAC address based
/sbin/ifconfig | grep HWaddr
# or on newer systems:
ip link show | grep link/ether

# Method 2: Using Synopsys lmhostid utility (after installation)
/opt/synopsys/syn/linux64/bin/lmhostid

# Method 3: System UUID (for floating licenses)
sudo dmidecode -s system-uuid
```

## Step 2: License Server Setup (if using local license server)

```bash
# If you have a license file, copy it to the licenses directory
cp synopsys.lic /opt/synopsys/licenses/

# Start license manager (if running local server)
/opt/synopsys/syn/linux64/bin/lmgrd -c /opt/synopsys/licenses/synopsys.lic -l /tmp/lmgrd.log &

# Check license server status
/opt/synopsys/syn/linux64/bin/lmstat -c /opt/synopsys/licenses/synopsys.lic -a
```

## Step 3: License Environment Configuration

```bash
```

```
# Method 1: License server
export SNPSLMD_LICENSE_FILE="27020@your-license-server.com"

# Method 2: License file
export SNPSLMD_LICENSE_FILE="/opt/synopsys/licenses/synopsys.lic"

# Method 3: Multiple sources (server + backup file)
export SNPSLMD_LICENSE_FILE="27020@server1.com:27020@server2.com:/opt/synopsys/licenses/backup.lic"

# Make it permanent
echo 'export SNPSLMD_LICENSE_FILE="27020@your-license-server.com"' >> ~/.bashrc
```

## 🔧 Environment Setup

### Step 1: Create Setup Script

```bash
```

```
# Create comprehensive setup script
cat > /opt/synopsys/setup_dc.sh << 'EOF'
#!/bin/bash

# Design Compiler Installation Path
export SYNOPSYS=/opt/synopsys
export DC_ROOT=/opt/synopsys/syn

# Add DC binaries to PATH
export PATH=$DC_ROOT/bin:$PATH

# Library paths
export LD_LIBRARY_PATH=$DC_ROOT/linux64/lib:$LD_LIBRARY_PATH

# License configuration
export SNPSLMD_LICENSE_FILE=${SNPSLMD_LICENSE_FILE:-"27020@license-server.com"}
export LM_LICENSE_FILE=$SNPSLMD_LICENSE_FILE

# Design Compiler specific environment
export SYNOPSYS_SYN_ROOT=$DC_ROOT
export TARGET_ARCH=linux64
export ARCH=linux64

# Setup file location
export SYNOPSYS_DC_SETUP_PATH="$PWD:$HOME/.synopsys_dc.setup:$DC_ROOT/admin/setup"

# Display configuration for Design Vision (GUI)
export DISPLAY=${DISPLAY:-:0.0}

# Memory optimization
export DC_EXEC_64BIT=true
export DC_USE_UNIFIED_MEMORY=true

# Tcl configuration
export DC_TCL_STARTUP_FILE="$HOME/.synopsys_dc.setup"

# Print setup information
echo "Design Compiler Environment Configured"
echo "===================================="
echo "SYNOPSYS: $SYNOPSYS"
echo "DC_ROOT: $DC_ROOT"
echo "License: $SNPSLMD_LICENSE_FILE"
echo "Target Architecture: $TARGET_ARCH"
echo ""

# Test basic functionality
if command -v dc_shell >/dev/null 2>&1; then
    echo "✓ dc_shell is available"
else
    echo "✗ dc_shell not found in PATH"
fi

if command -v design_vision >/dev/null 2>&1; then
    echo "✓ design_vision is available"
else
    echo "✗ design_vision not found in PATH"
fi
EOF

# Make script executable
chmod +x /opt/synopsys/setup_dc.sh
```

## Step 2: User Configuration

```bash
```

```
# Add setup to user's bashrc
echo "source /opt/synopsys/setup_dc.sh" >> ~/.bashrc

# Create user-specific DC setup file
cat > ~/.synopsys_dc.setup << 'EOF'
# User-specific Design Compiler Setup

# Set target library (example - adjust for your libraries)
# set target_library {slow.db}
# set link_library {* slow.db fast.db}

# Set search paths for libraries
# set search_path {. /path/to/your/libraries}

# Set symbol library for schematic view
# set symbol_library {generic.sdb}

# Enable command logging
history keep 100

# Set default working directory
# define_design_lib WORK -path ./work

# Custom aliases
alias ll "ls -la"
alias h history

# Load custom procedures if they exist
if {[file exists "./dc_scripts/custom_procs.tcl"]} {
    source "./dc_scripts/custom_procs.tcl"
}

# Print welcome message
echo "Design Compiler User Setup Loaded"
EOF
```

## ✅ Installation Verification

### Step 1: Basic Installation Check

bash

```
# Source the environment
source /opt/synopsys/setup_dc.sh

# Check installation
ls -la /opt/synopsys/syn/bin/

# Verify executables
which dc_shell
which design_vision

# Check version
dc_shell -help | head -10
```

### Step 2: License Verification

```
bash

# Test license connectivity
dc_shell -f /dev/null -x "exit" 2>&1 | grep -i license

# Check available licenses
lmstat -c $SNPSLMD_LICENSE_FILE -f | grep -E "(Users of|Total of)"

# Test license checkout
echo "exit" | dc_shell -64bit
```

### Step 3: Functional Testing

```
bash
```

```
# Create test directory
mkdir -p ~/dc_test
cd ~/dc_test

# Create simple test design
cat > test_design.v << 'EOF'
module test_adder (
  input [3:0] a,
  input [3:0] b,
  input cin,
  output [3:0] sum,
  output cout
);
  assign {cout, sum} = a + b + cin;
endmodule
EOF

# Create synthesis script
cat > synth_test.tcl << 'EOF'
# Simple synthesis test script

# Read design
read_verilog test_design.v

# Set current design
current_design test_adder

# Link design (use generic libraries for test)
link

# Check design
check_design

# Compile (basic synthesis)
compile

# Report results
report_area
report_timing

# Write results
write -format verilog -output test_adder_synth.v
```

```
# Exit
exit
EOF

# Run synthesis test
dc_shell -f synth_test.tcl | tee synthesis.log

# Check if synthesis completed successfully
if [ -f test_adder_synth.v ]; then
    echo "✓ Synthesis test PASSED"
    echo "Generated files:"
    ls -la *.v *.log
else
    echo "✗ Synthesis test FAILED"
    echo "Check synthesis.log for errors"
fi
```

### Step 4: GUI Testing (Design Vision)

```bash
# Test Design Vision GUI (requires X11)
# Make sure DISPLAY is set correctly
echo $DISPLAY

# Launch Design Vision
design_vision &

# If GUI doesn't start, check X11 forwarding:
xauth list
xdpyinfo | head

# For remote access via SSH:
ssh -X username@hostname
```

## 🚀 Advanced Configuration

### Technology Library Setup

```bash
```

```
# Create directories for technology libraries
mkdir -p /opt/synopsys/libraries/{db,mw,tf,symbols}

# Example library configuration in ~/.synopsys_dc.setup
cat >> ~/.synopsys_dc.setup << 'EOF'

# Technology library configuration
set search_path {. /opt/synopsys/libraries/db}
set target_library {your_slow_lib.db}
set link_library {* your_slow_lib.db your_fast_lib.db}
set symbol_library {/opt/synopsys/libraries/symbols/generic.sdb}

# Milkyway library for physical design
set mw_reference_library {/opt/synopsys/libraries/mw/your_tech_lib}
set mw_design_library "./milkyway_lib"

# TLU+ files for parasitic extraction
set tluplus_file "/opt/synopsys/libraries/tf/your_tech.tlup"
EOF
```

### Performance Optimization

```bash
# Add to ~/.synopsys_dc.setup for better performance
cat >> ~/.synopsys_dc.setup << 'EOF'

# Performance optimization settings
set compile_advanced_power_optimization true
set compile_enable_constant_propagation_with_no_boundary_opt true
set hdlin_enable_presto_for_verilog true

# Memory optimization
set dc_allow_ram_map true
set compile_ultra_ungroup_small_hierarchies true

# Multi-core optimization (if supported by license)
set_host_options -max_cores 4
EOF
```

## 🛠️ Troubleshooting

## Common Installation Issues

### Issue 1: Permission Denied

```bash
# Fix ownership issues
sudo chown -R synopsys:synopsys /opt/synopsys
sudo chmod -R 755 /opt/synopsys/syn/bin

# Add execute permissions
chmod +x /opt/synopsys/syn/bin/*
```

### Issue 2: Missing Libraries

```bash
# Check for missing system libraries
ldd /opt/synopsys/syn/linux64/bin/dc_shell

# Install missing dependencies
sudo apt install -y libc6-dev libstdc++6 libgcc-s1

# For older systems, may need 32-bit compatibility
sudo apt install -y lib32stdc++6 lib32gcc-s1
```

### Issue 3: License Issues

```bash
# Debug license problems
export LM_LICENSE_DEBUG=1
dc_shell -f /dev/null -x "exit"

# Check license server connectivity
telnet your-license-server.com 27020

# Verify license file format
head -20 /opt/synopsys/licenses/synopsys.lic
```

### Issue 4: Display Issues (GUI)

```bash
```

```
# For local machine
export DISPLAY=:0.0
xhost +local:

# For SSH access
ssh -X username@hostname
echo $DISPLAY

# Test X11
xeyes &  # Should show eyes following mouse
```

### Log Analysis

bash

```
# Check installation logs
ls -la /tmp/installer*.log
tail -100 /tmp/installer*.log

# Check license manager logs
tail -100 /tmp/lmgrd.log

# DC runtime logs
tail -100 ~/.synopsys_dc.log
```

## 📚 Post-Installation Steps

### Documentation Access

bash

```
# Access documentation
firefox /opt/synopsys/syn/doc/index.html &

# Or from command line
dc_shell
dc_shell> man compile
dc_shell> help set_max_delay
```

### Create Aliases and Shortcuts

bash

```
# Add convenient aliases to ~/.bashrc
cat >> ~/.bashrc << 'EOF'

# Design Compiler aliases
alias dc='dc_shell'
alias dv='design_vision &'
alias dc_help='dc_shell -help'

# Directory shortcuts
alias cddc='cd /opt/synopsys/syn'
alias cdlib='cd /opt/synopsys/libraries'
EOF
```

### Backup Configuration

```bash
# Create backup of working configuration
tar -czf dc_config_backup.tar.gz \
    ~/.synopsys_dc.setup \
    ~/.bashrc \
    /opt/synopsys/setup_dc.sh

# Store in safe location
mv dc_config_backup.tar.gz /opt/synopsys/backups/
```

## 🎯 Verification Checklist

- □ System requirements met
- □ Installation completed without errors
- □ License server accessible
- □ dc_shell launches successfully
- □ design_vision GUI works (if X11 available)
- □ Simple synthesis test passes
- □ Technology libraries configured
- □ Environment setup persistent across sessions
- □ Documentation accessible
- □ Backup created

## 📞 Support Resources

### Synopsys Support

- **SolvNet**: https://solvnet.synopsys.com
- **Documentation**: Installed at `/opt/synopsys/syn/doc/`
- **Training**: Synopsys University courses

### Community Resources

- **Design Compiler Reference Manual**: Essential reading
- **Synopsys Users Group**: Online forums and discussions
- **University Resources**: Academic license support

This completes the comprehensive Design Compiler local installation guide. The installation should be fully functional for synthesis workflows.