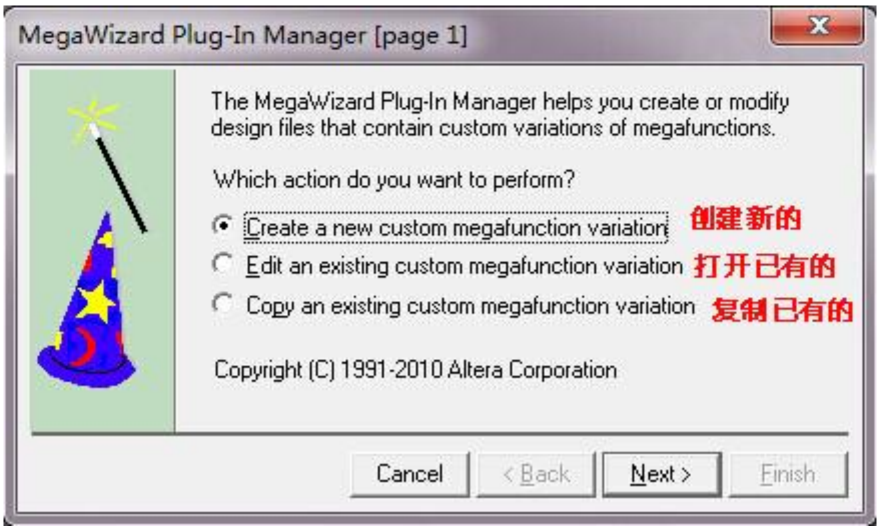
打开 quartus 软件，然后 TOOL，选择 MegaWizard 选项，弹出如下图：

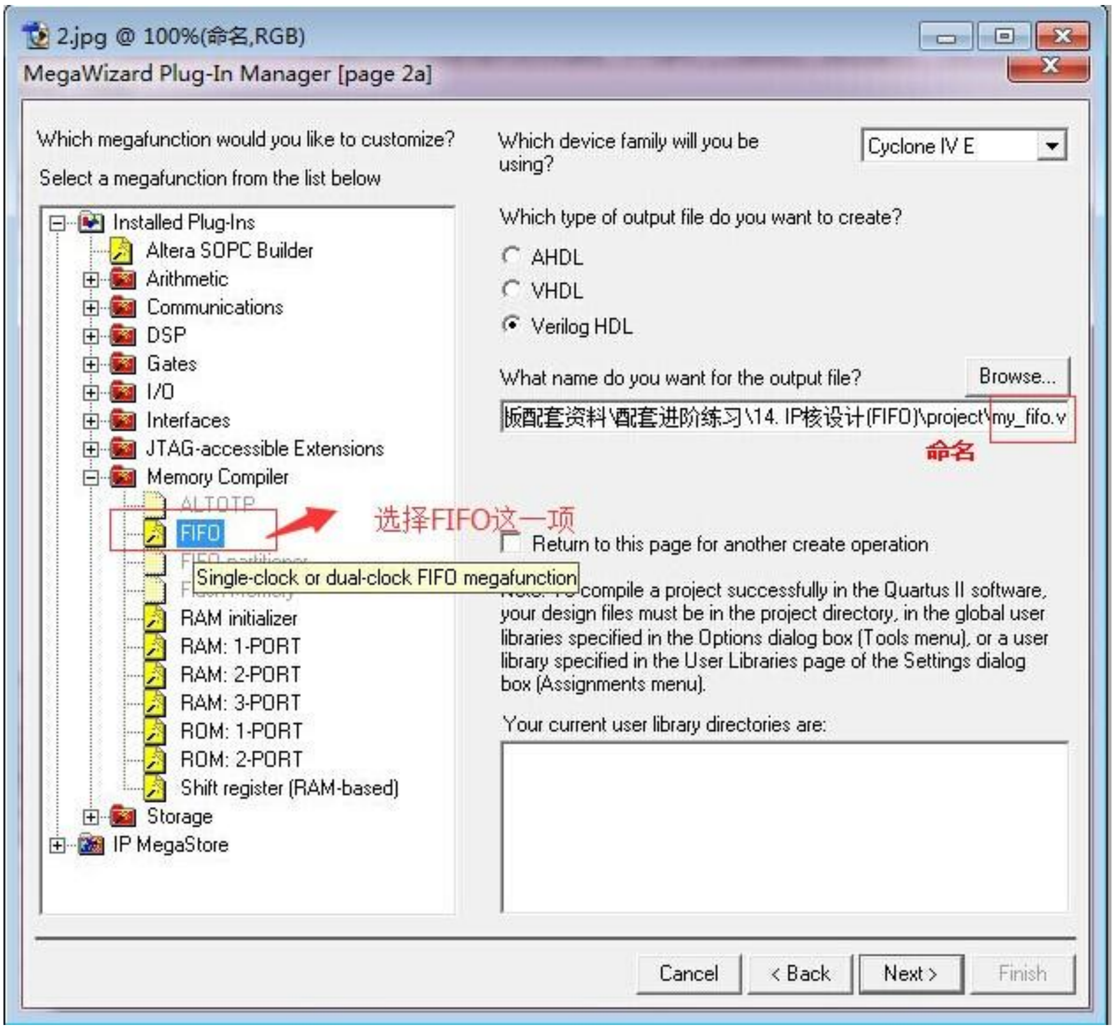

MegaWizard Plug-In Manager [page 1]

The MegaWizard Plug-In Manager helps you create or modify design files that contain custom variations of megafunctions.

Which action do you want to perform?

- [✓ Create a new custom megafunction variation] 创建新的
- Edit an existing custom megafunction variation 打开已有的
- Copy an existing custom megafunction variation 复制已有的

Copyright (C) 1991-2010 Altera Corporation


2.jpg @ 100%(命名,RGB)

MegaWizard Plug-In Manager [page 2a]

Which megafunction would you like to customize?

Select a megafunction from the list below

- Installed Plug-Ins
  - Altera SOPC Builder
  - Arithmetic
  - Communications
  - DSP
  - Gates
  - I/O
  - Interfaces
  - JTAG-accessible Extensions
  - Memory Compiler
    - ALTOTP
    - FIFO
    - FIFO partitioner
    - Flash Memory
    - RAM initializer
    - RAM: 1-PORT
    - RAM: 2-PORT
    - RAM: 3-PORT
    - ROM: 1-PORT
    - ROM: 2-PORT
    - Shift register (RAM-based)
  - Storage
- IP MegaStore

Single-clock or dual-clock FIFO megafunction

选择FIFO这一项

Which device family will you be using? Cyclone IV E

Which type of output file do you want to create?

- AHDL
- VHDL
- [✓ Verilog HDL]

What name do you want for the output file?

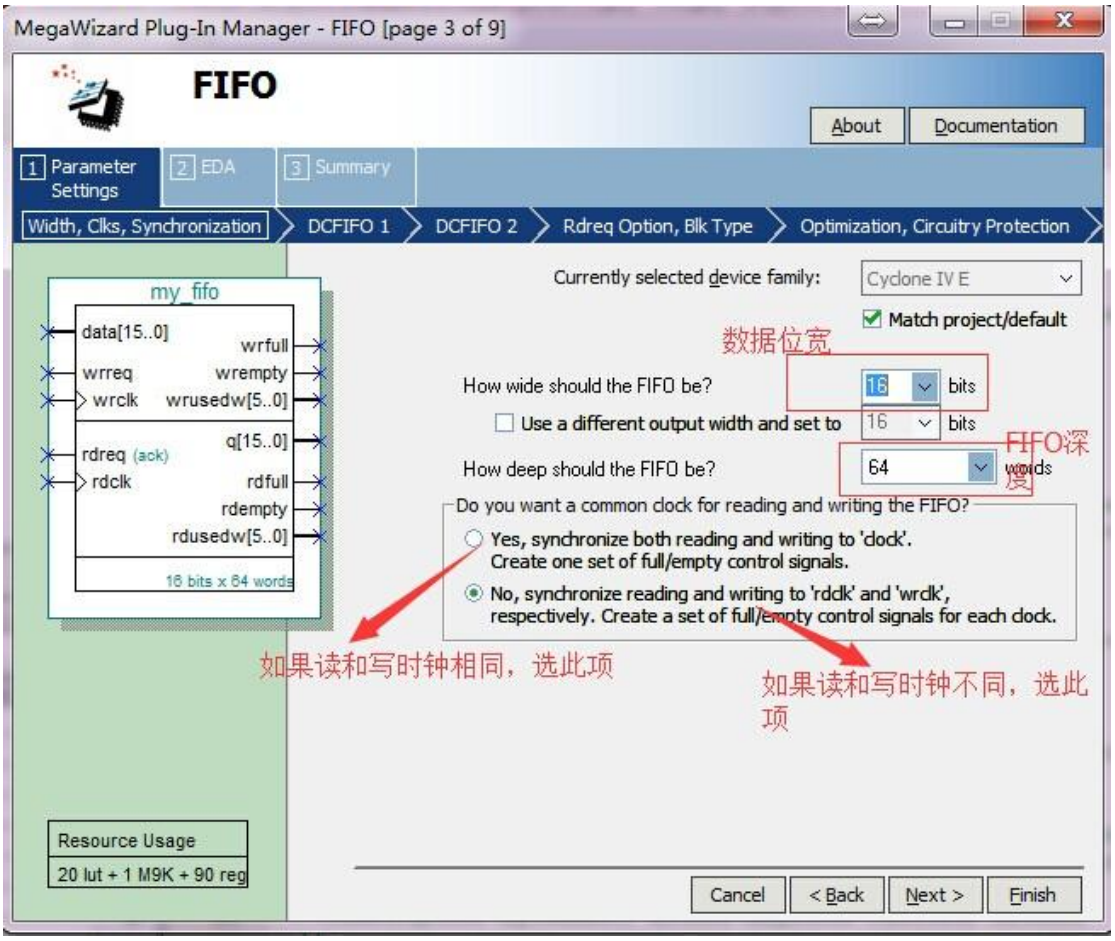
版配套资料\配套进阶练习\14. IP核设计(FIFO)\project\my_fifo.v

命名

Return to this page for another create operation

Note: To compile a project successfully in the Quartus II software, your design files must be in the project directory, in the global user libraries specified in the Options dialog box (Tools menu), or a user library specified in the User Libraries page of the Settings dialog box (Assignments menu).

Your current user library directories are:



MegaWizard Plug-In Manager - FIFO [page 3 of 9]

FIFO

About Documentation

1 Parameter Settings | 2 EDA | 3 Summary

Width, Clks, Synchronization > DCFIFO 1 > DCFIFO 2 > Rdreq Option, Blk Type > Optimization, Circuitry Protection

Currently selected device family: Cyclone IV E

[✓] Match project/default

数据位宽

How wide should the FIFO be? 16 bits

[ ] Use a different output width and set to 16 bits

FIFO深度

How deep should the FIFO be? 64 words

Do you want a common clock for reading and writing the FIFO?

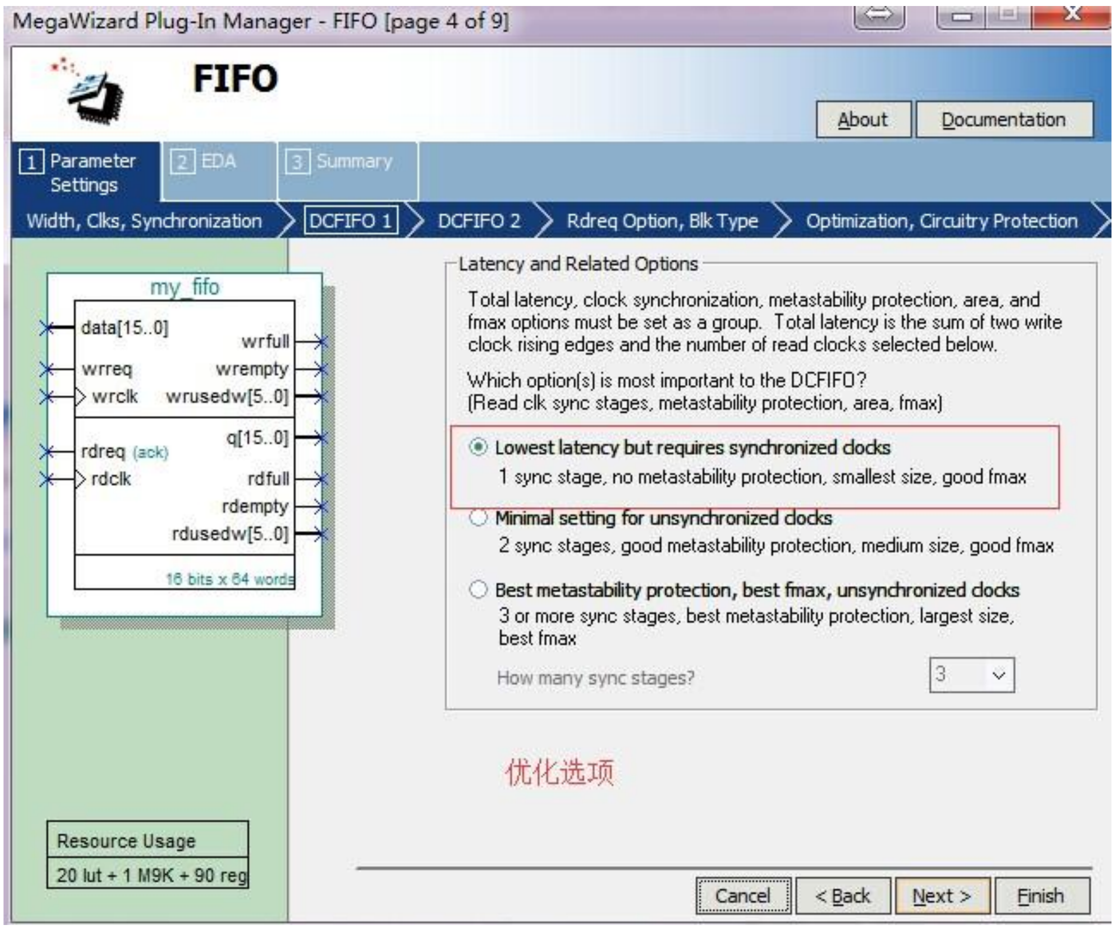
( ) Yes, synchronize both reading and writing to 'clock'. Create one set of full/empty control signals.

[✓ (●) No, synchronize reading and writing to 'rdclk' and 'wrclk', respectively. Create a set of full/empty control signals for each clock.]

如果读和写时钟相同，选此项

如果读和写时钟不同，选此项

Resource Usage
20 lut + 1 M9K + 90 reg



MegaWizard Plug-In Manager - FIFO [page 4 of 9]

FIFO

About Documentation

1 Parameter Settings | 2 EDA | 3 Summary

Width, Clks, Synchronization > DCFIFO 1 > DCFIFO 2 > Rdreq Option, Blk Type > Optimization, Circuitry Protection

Resource Usage
20 lut + 1 M9K + 90 reg

Latency and Related Options

Total latency, clock synchronization, metastability protection, area, and fmax options must be set as a group. Total latency is the sum of two write clock rising edges and the number of read clocks selected below.

Which option(s) is most important to the DCFIFO?
(Read clk sync stages, metastability protection, area, fmax)

[✓ Lowest latency but requires synchronized clocks]
1 sync stage, no metastability protection, smallest size, good fmax

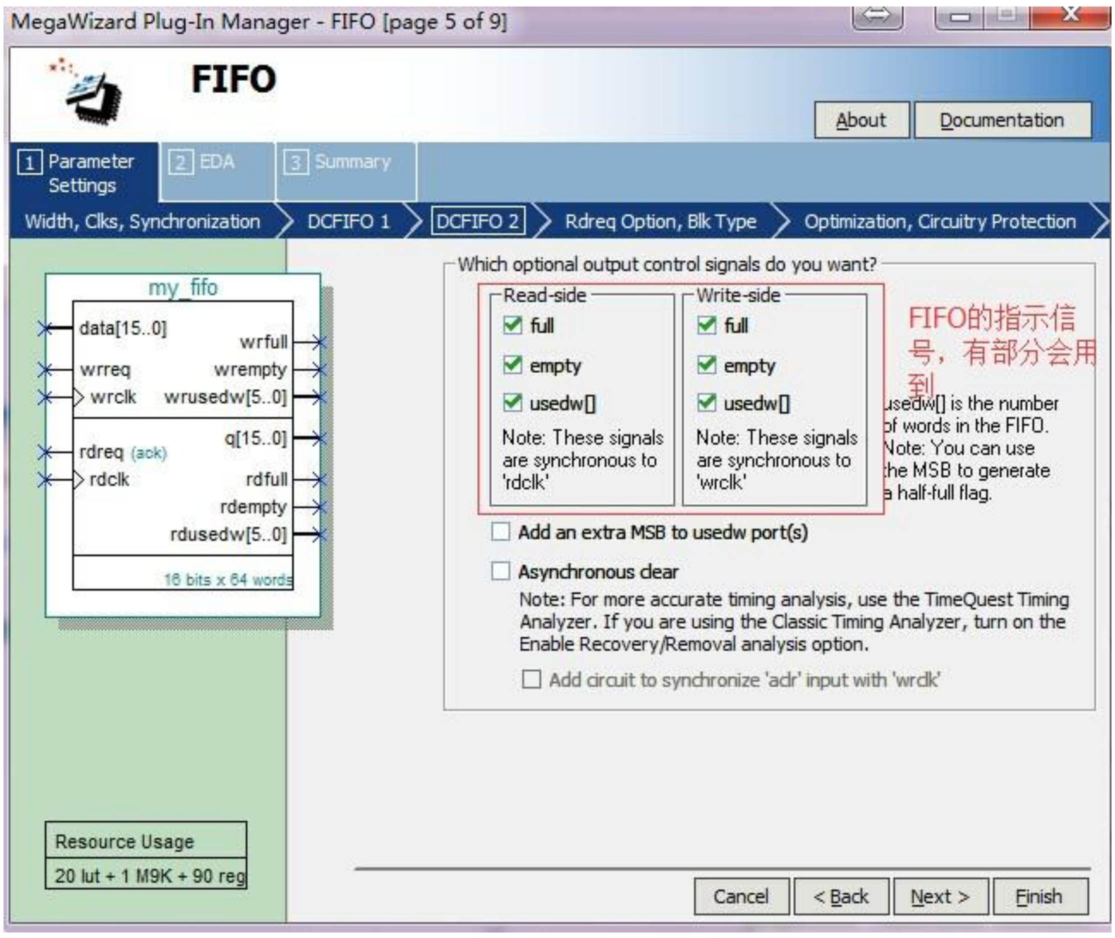
Minimal setting for unsynchronized clocks
2 sync stages, good metastability protection, medium size, good fmax

Best metastability protection, best fmax, unsynchronized clocks
3 or more sync stages, best metastability protection, largest size, best fmax

How many sync stages? 3

优化选项

Cancel | < Back | Next > | Finish



MegaWizard Plug-In Manager - FIFO [page 5 of 9]

FIFO

Which optional output control signals do you want?

Read-side
- [✓] full
- [✓] empty
- [✓] usedw[]

Note: These signals are synchronous to 'rdclk'

Write-side
- [✓] full
- [✓] empty
- [✓] usedw[]

Note: These signals are synchronous to 'wrclk'

FIFO的指示信号，有部分会用到

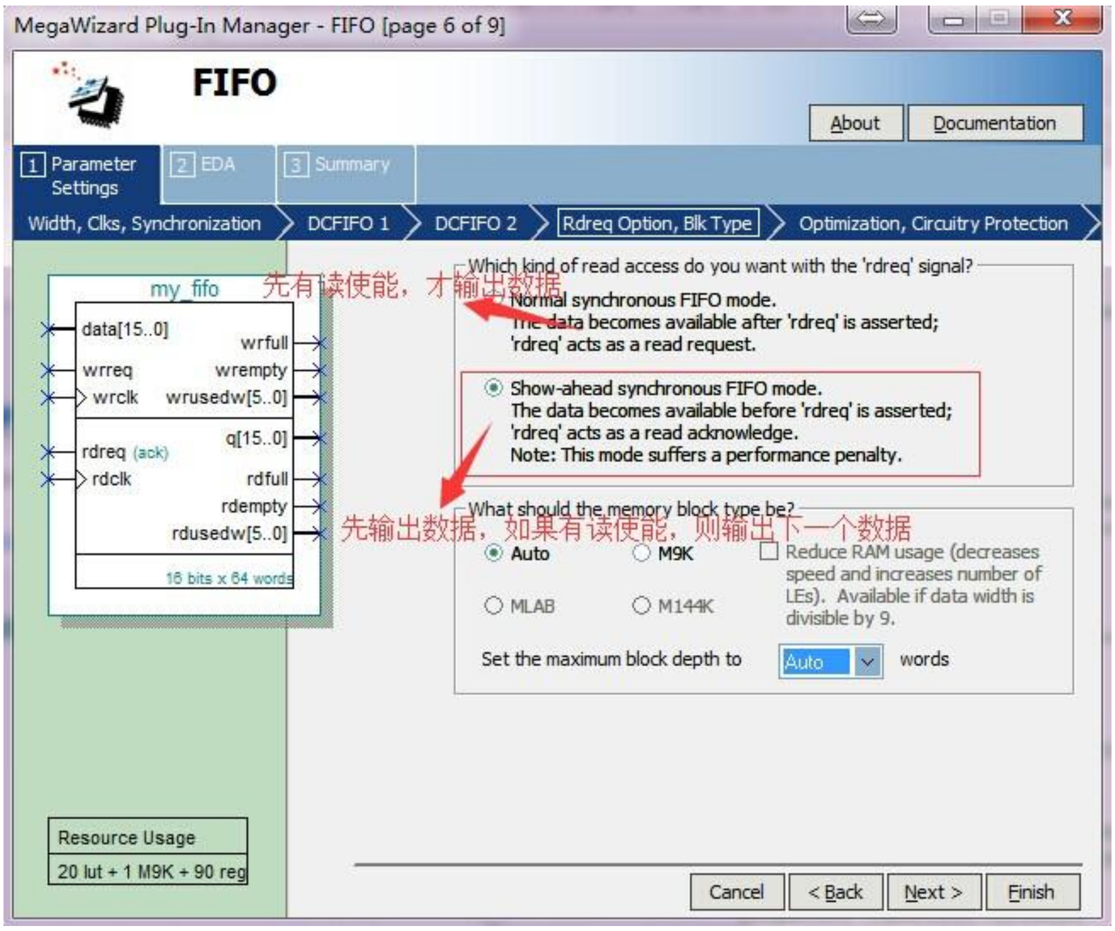
usedw[] is the number of words in the FIFO. Note: You can use the MSB to generate a half-full flag.

- [ ] Add an extra MSB to usedw port(s)
- [ ] Asynchronous clear

Note: For more accurate timing analysis, use the TimeQuest Timing Analyzer. If you are using the Classic Timing Analyzer, turn on the Enable Recovery/Removal analysis option.

- [ ] Add circuit to synchronize 'aclr' input with 'wrclk'

Resource Usage
20 lut + 1 M9K + 90 reg



先有读使能，才输出数据

先输出数据，如果有读使能，则输出下一个数据

Which kind of read access do you want with the 'rdreq' signal?

○ Normal synchronous FIFO mode.
The data becomes available after 'rdreq' is asserted;
'rdreq' acts as a read request.

● Show-ahead synchronous FIFO mode.
The data becomes available before 'rdreq' is asserted;
'rdreq' acts as a read acknowledge.
Note: This mode suffers a performance penalty.

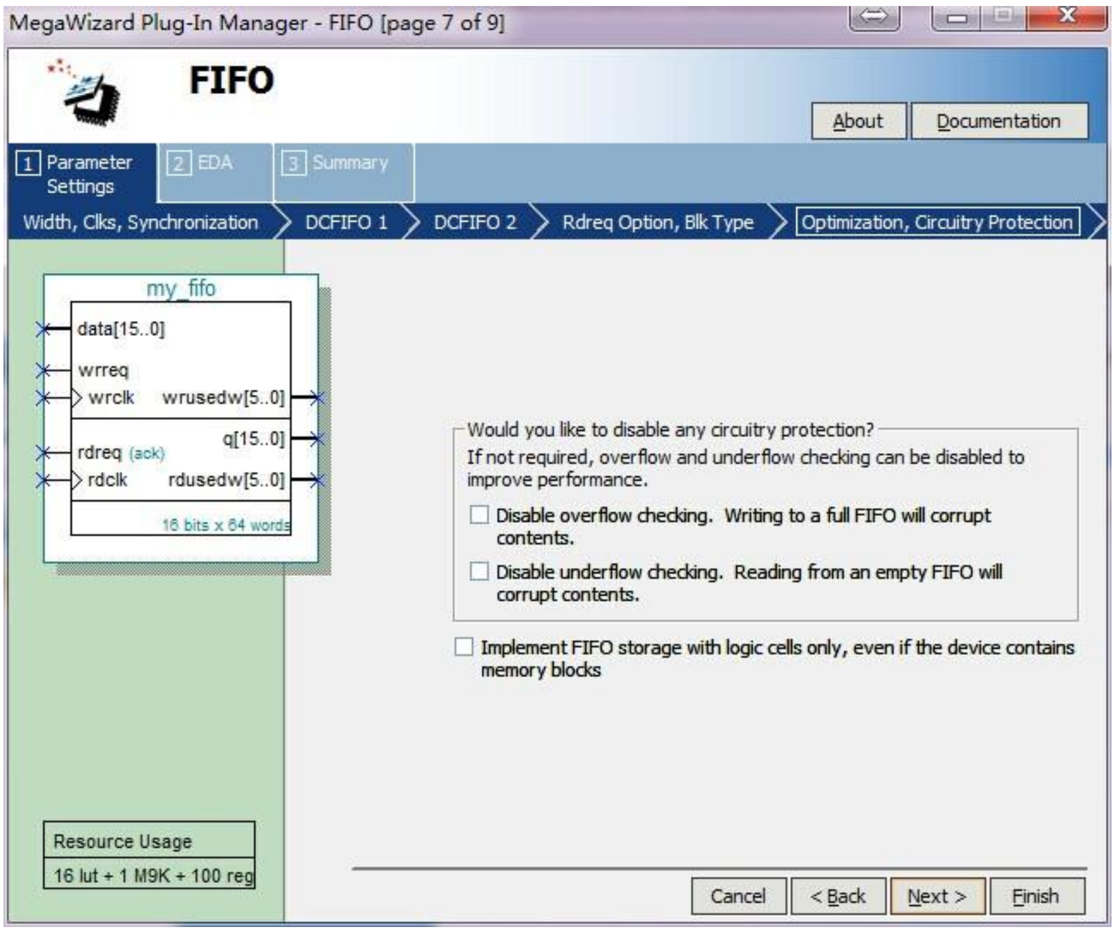
What should the memory block type be?

● Auto ○ M9K ☐ Reduce RAM usage (decreases speed and increases number of LEs). Available if data width is divisible by 9.

○ MLAB ○ M144K

Set the maximum block depth to Auto words

Resource Usage
20 lut + 1 M9K + 90 reg



MegaWizard Plug-In Manager - FIFO [page 7 of 9]

FIFO

Resource Usage
16 lut + 1 M9K + 100 reg

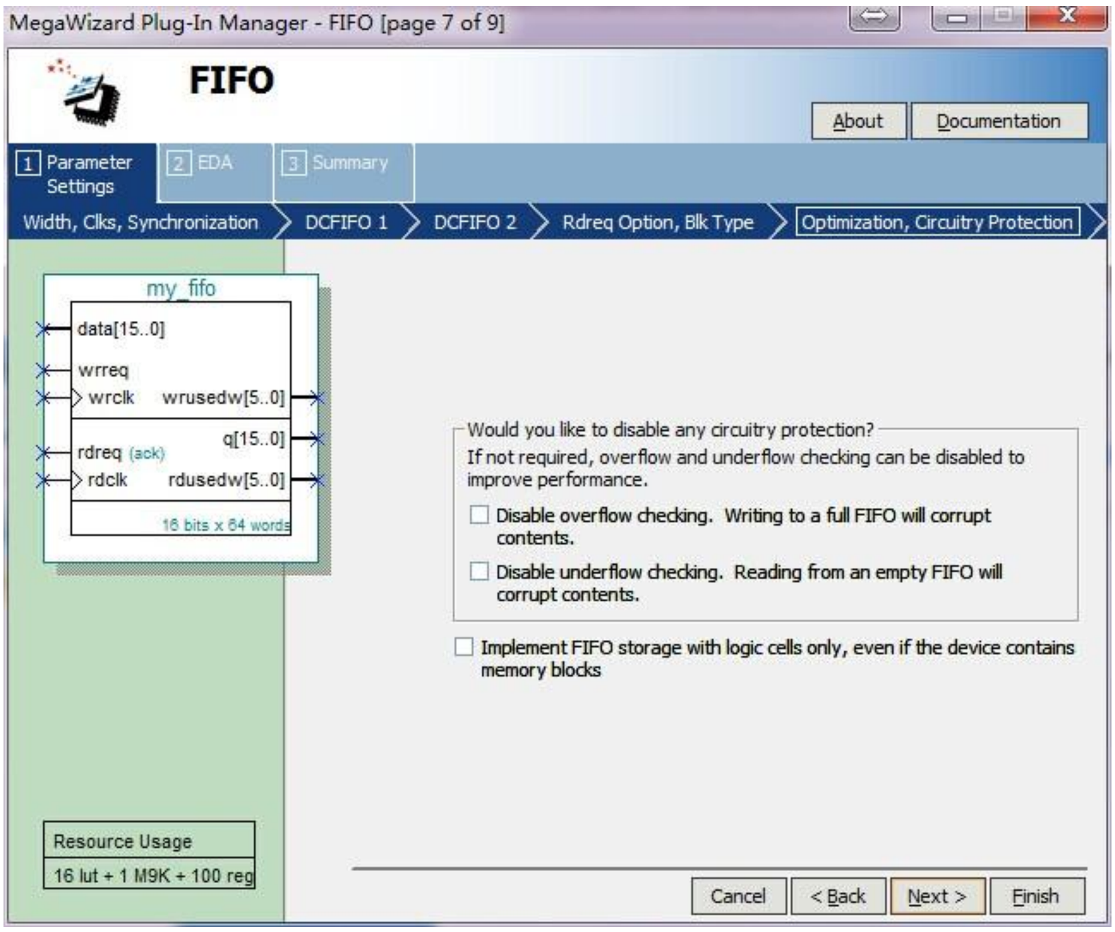
Would you like to disable any circuitry protection?

If not required, overflow and underflow checking can be disabled to improve performance.

☐ Disable overflow checking. Writing to a full FIFO will corrupt contents.

☐ Disable underflow checking. Reading from an empty FIFO will corrupt contents.

☐ Implement FIFO storage with logic cells only, even if the device contains memory blocks



MegaWizard Plug-In Manager - FIFO [page 7 of 9]

FIFO

About | Documentation

1 Parameter Settings | 2 EDA | 3 Summary

Width, Clks, Synchronization > DCFIFO 1 > DCFIFO 2 > Rdreq Option, Blk Type > Optimization, Circuitry Protection

Would you like to disable any circuitry protection?

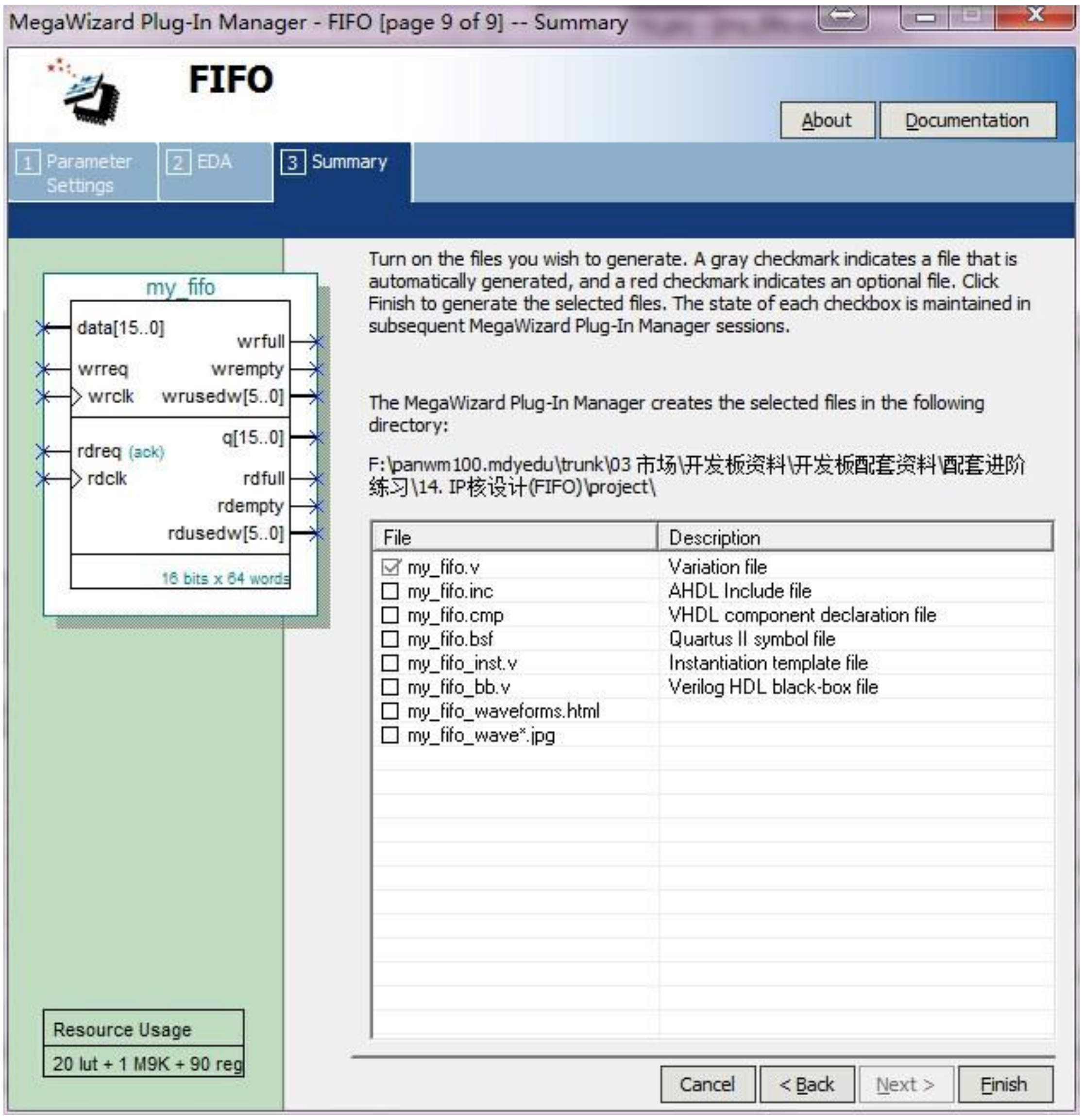
If not required, overflow and underflow checking can be disabled to improve performance.

- [ ] Disable overflow checking. Writing to a full FIFO will corrupt contents.
- [ ] Disable underflow checking. Reading from an empty FIFO will corrupt contents.

- [ ] Implement FIFO storage with logic cells only, even if the device contains memory blocks

Resource Usage
16 lut + 1 M9K + 100 reg

Cancel | < Back | Next > | Finish



MegaWizard Plug-In Manager - FIFO [page 9 of 9] -- Summary

FIFO

Turn on the files you wish to generate. A gray checkmark indicates a file that is automatically generated, and a red checkmark indicates an optional file. Click Finish to generate the selected files. The state of each checkbox is maintained in subsequent MegaWizard Plug-In Manager sessions.

The MegaWizard Plug-In Manager creates the selected files in the following directory:

F:\panwm100.mdyedu\trunk\03 市场\开发板资料\开发板配套资料\配套进阶练习\14. IP核设计(FIFO)\project\

File	Description
✓ my_fifo.v	Variation file
☐ my_fifo.inc	AHDL Include file
☐ my_fifo.cmp	VHDL component declaration file
☐ my_fifo.bsf	Quartus II symbol file
☐ my_fifo_inst.v	Instantiation template file
☐ my_fifo_bb.v	Verilog HDL black-box file
☐ my_fifo_waveforms.html	
☐ my_fifo_wave*.jpg	

Resource Usage
20 lut + 1 M9K + 90 reg