SHA-256

PYNQ-Z2

Livoroi Arsal-Hanif

SHA-256

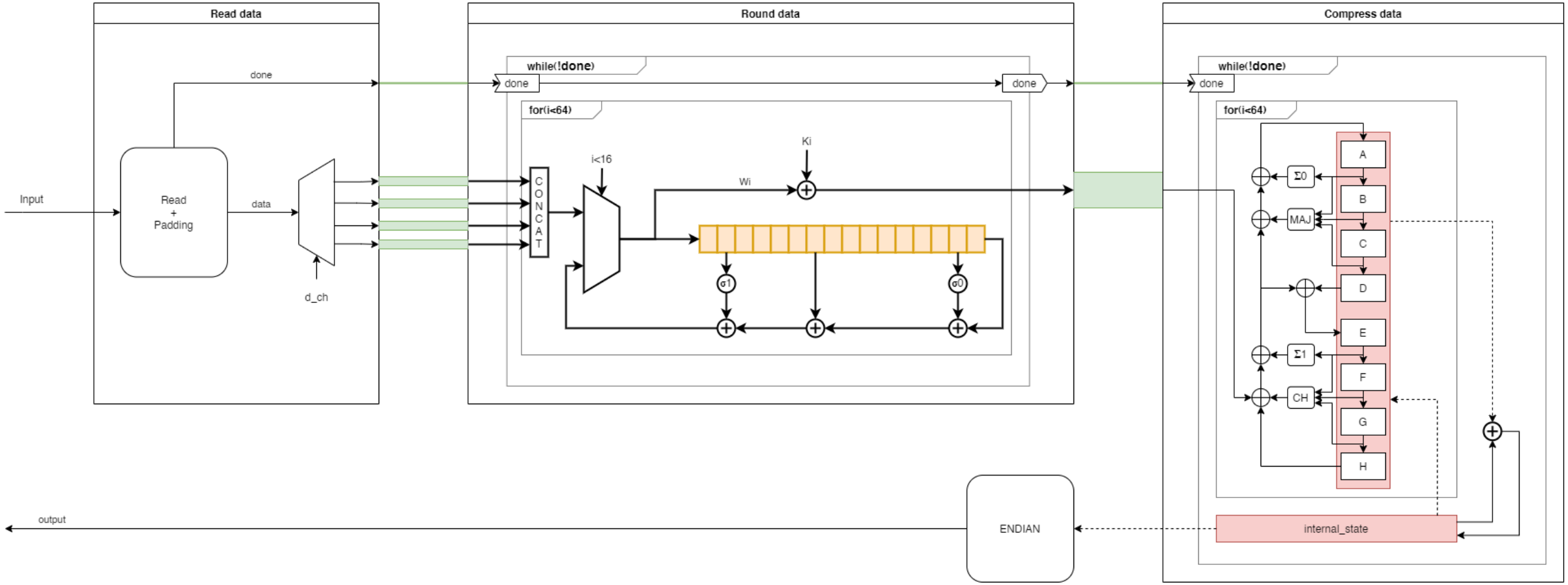
- Input: Messaggio di lunghezza qualsiasi
- Output: HASH da 256 bit

Fasi principali

- Padding: In: Msg | Out: N chunk da 16 uchar
- Espansione: In: 16 uchar | Out: 64 uint
- Compressione: In: 64 uint | Out: HASH

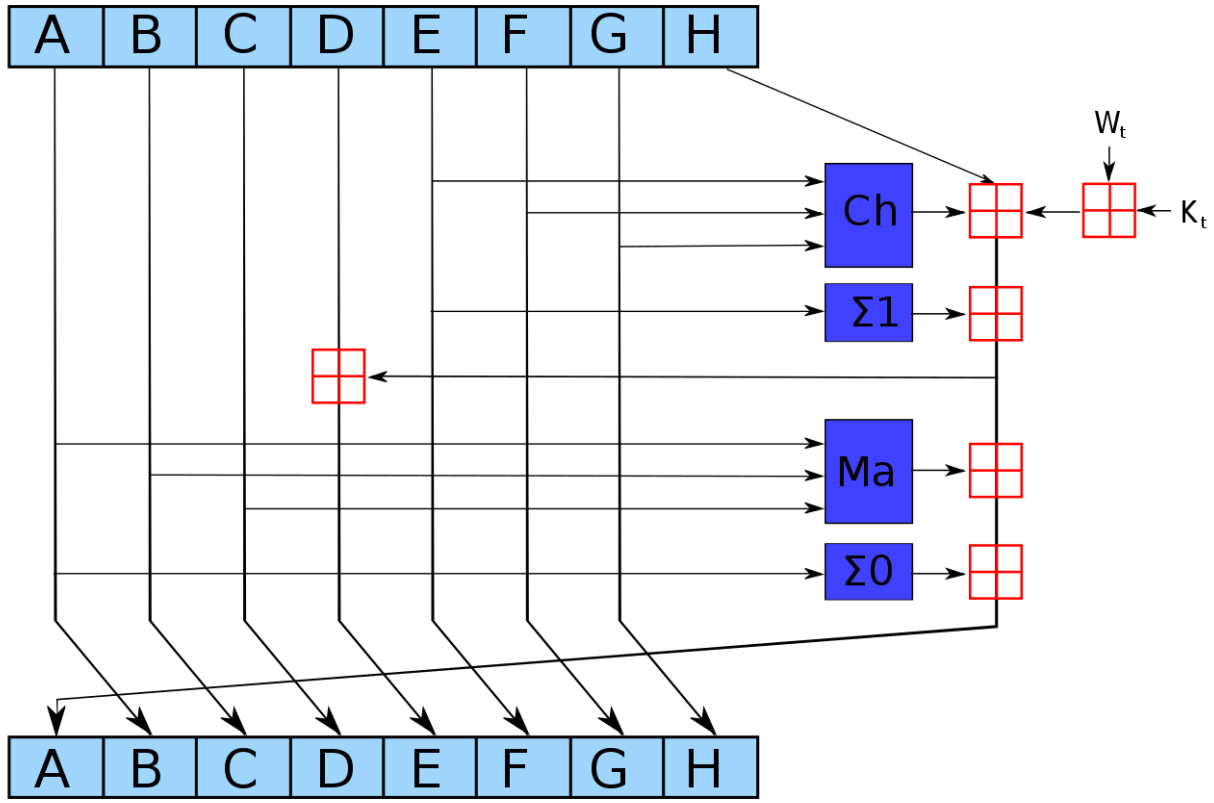
HLS

- Direttiva Dataflow
- Direttiva Unroll
- Direttiva Pipeline
- Stream
- Shift register
- Segnale di Sync



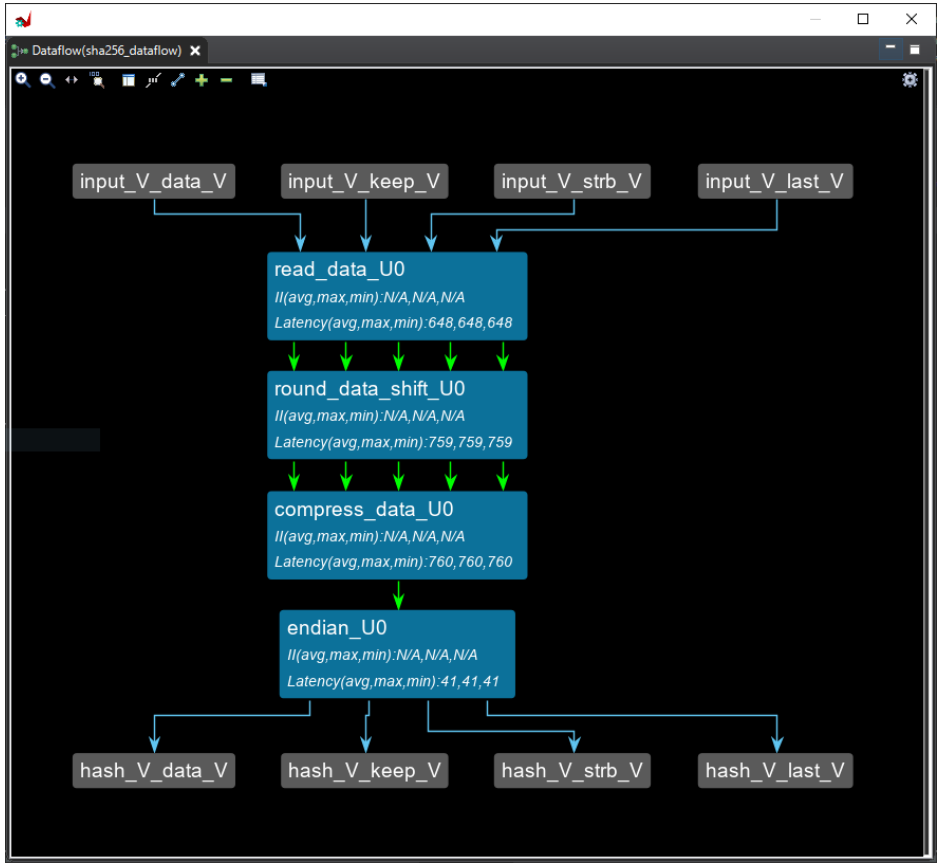
Sympy

- Libreria python
- Manipolazione simbolica
- Utilizzato per la fase di compressione
- Risultato simile alla direttiva Unroll

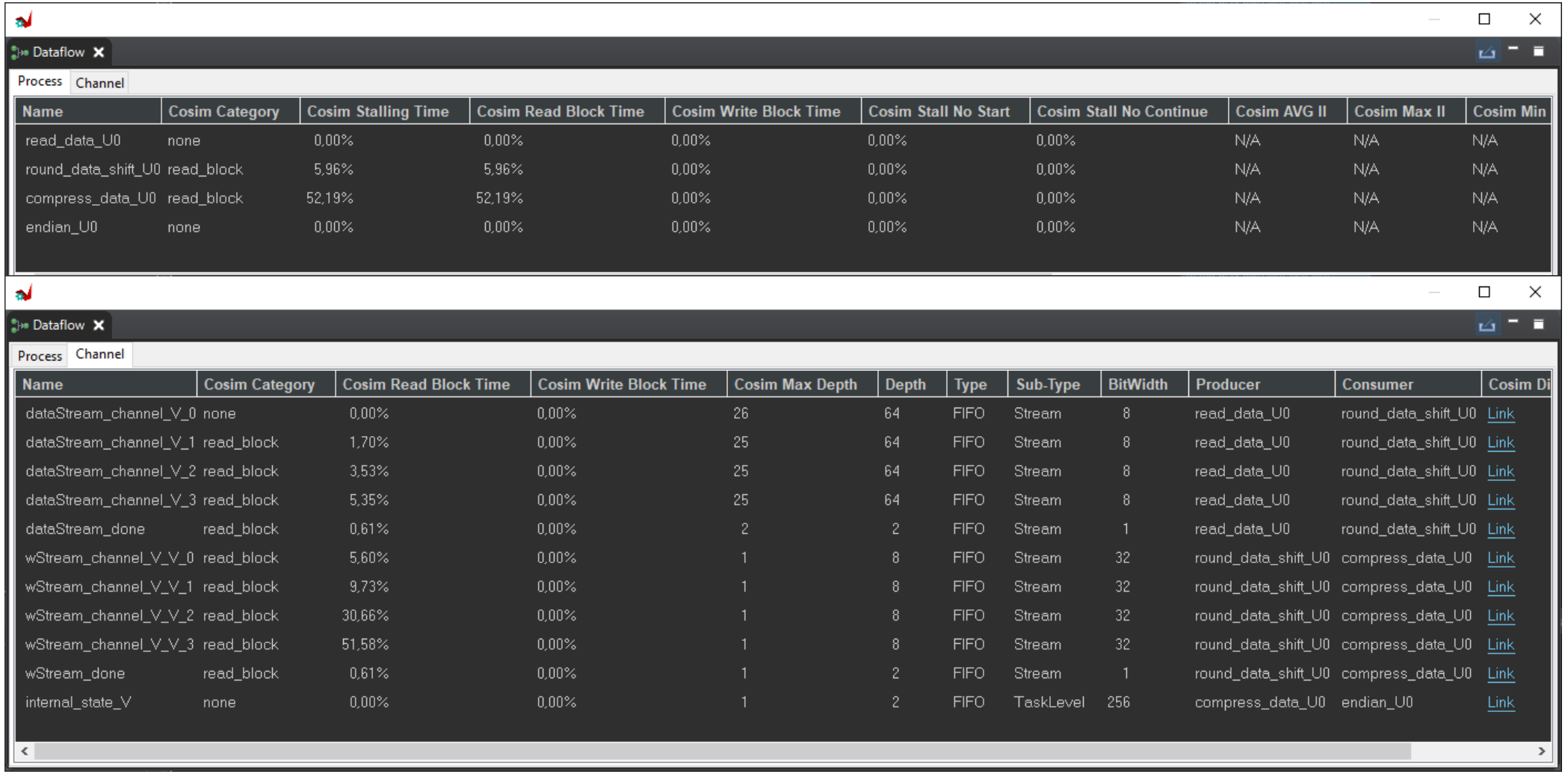


DATAFLOW VIEW

- Strumento fondamentale
- Visualizza i canali
- Analizza eventuali congestioni
- Permette di dimensionare gli stream



Processi e Canali sintetizzati



Dataflow

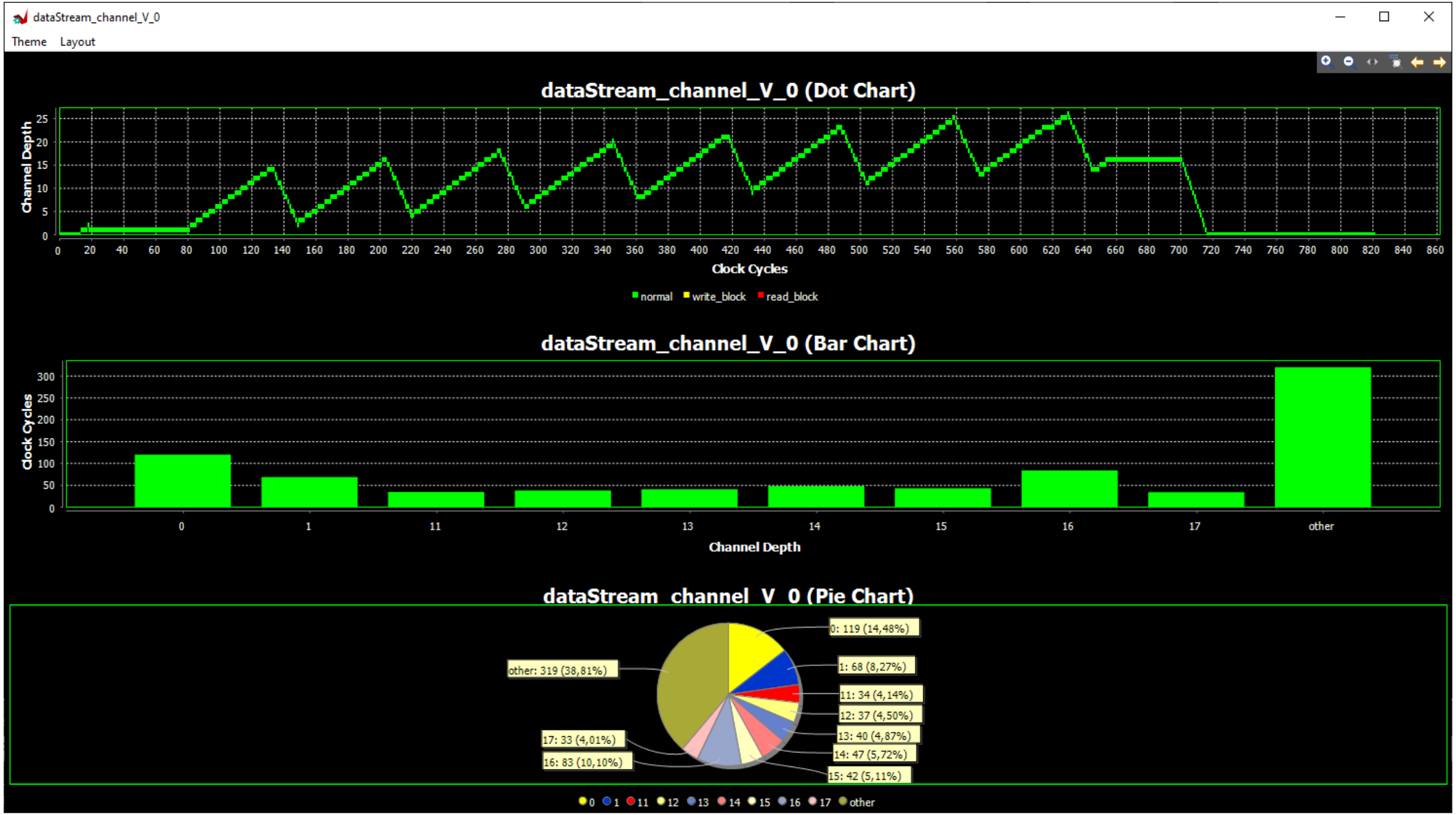
Process | Channel

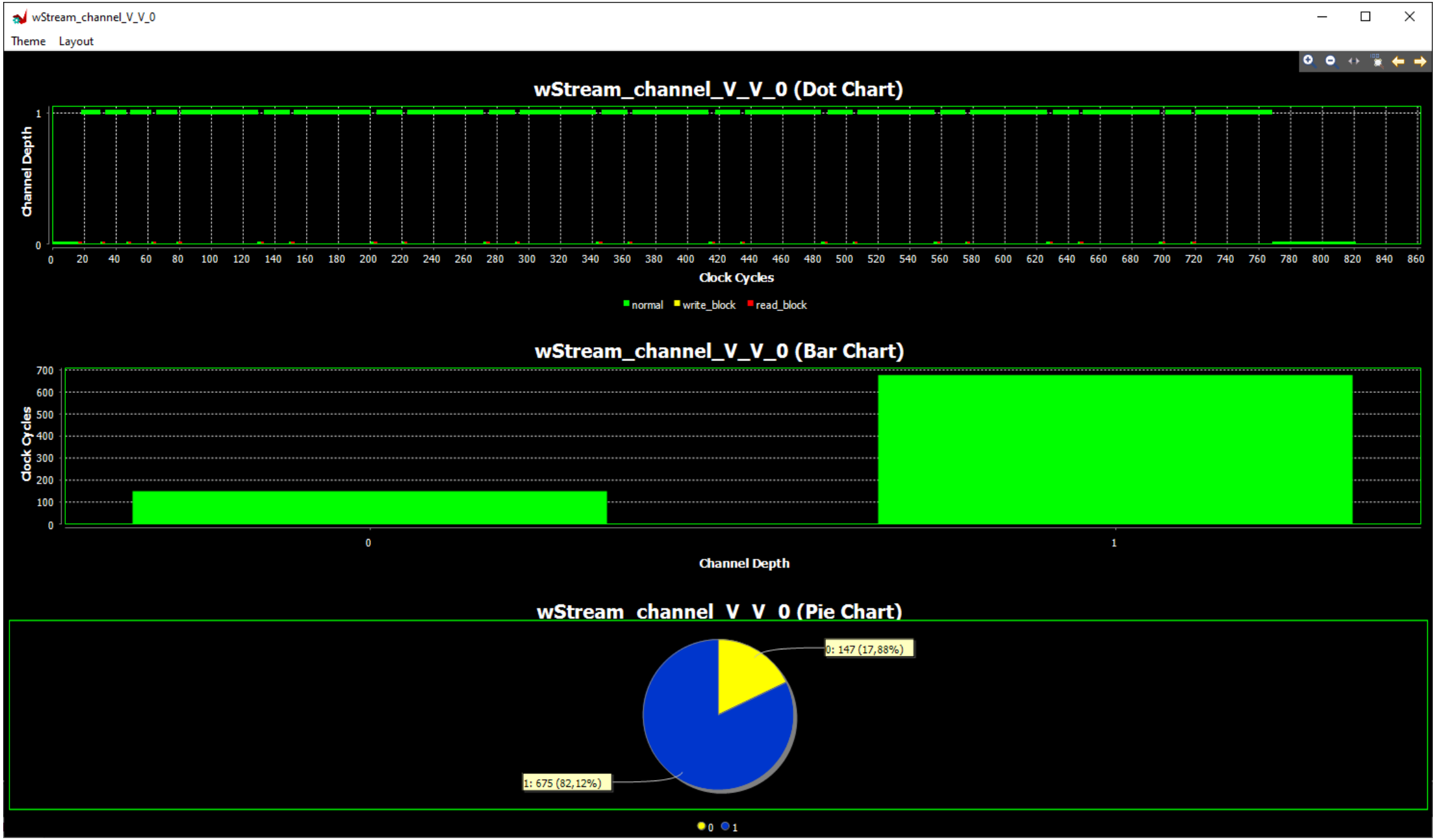
Name	Cosim Category	Cosim Stalling Time	Cosim Read Block Time	Cosim Write Block Time	Cosim Stall No Start	Cosim Stall No Continue	Cosim AVG II	Cosim Max II	Cosim Min
read_data_U0	none	0,00%	0,00%	0,00%	0,00%	0,00%	N/A	N/A	N/A
round_data_shift_U0	read_block	5,96%	5,96%	0,00%	0,00%	0,00%	N/A	N/A	N/A
compress_data_U0	read_block	52,19%	52,19%	0,00%	0,00%	0,00%	N/A	N/A	N/A
endian_U0	none	0,00%	0,00%	0,00%	0,00%	0,00%	N/A	N/A	N/A

Dataflow

Process | Channel

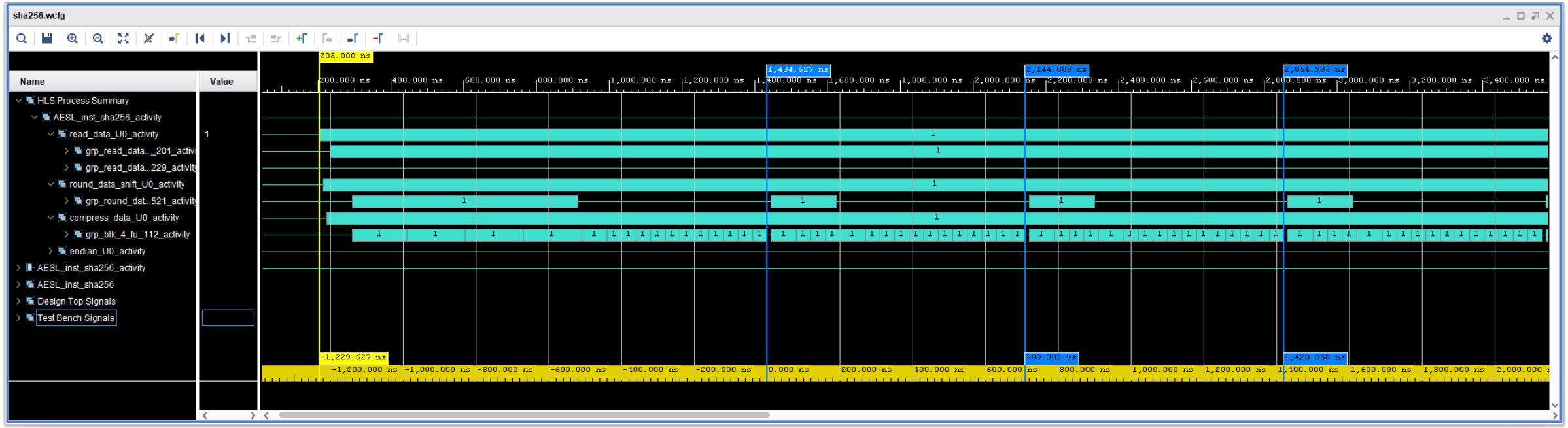
Name	Cosim Category	Cosim Read Block Time	Cosim Write Block Time	Cosim Max Depth	Depth	Type	Sub-Type	BitWidth	Producer	Consumer	Cosim Di
dataStream_channel_V_0	none	0,00%	0,00%	26	64	FIFO	Stream	8	read_data_U0	round_data_shift_U0	Link
dataStream_channel_V_1	read_block	1,70%	0,00%	25	64	FIFO	Stream	8	read_data_U0	round_data_shift_U0	Link
dataStream_channel_V_2	read_block	3,53%	0,00%	25	64	FIFO	Stream	8	read_data_U0	round_data_shift_U0	Link
dataStream_channel_V_3	read_block	5,35%	0,00%	25	64	FIFO	Stream	8	read_data_U0	round_data_shift_U0	Link
dataStream_done	read_block	0,61%	0,00%	2	2	FIFO	Stream	1	read_data_U0	round_data_shift_U0	Link
wStream_channel_V_V_0	read_block	5,60%	0,00%	1	8	FIFO	Stream	32	round_data_shift_U0	compress_data_U0	Link
wStream_channel_V_V_1	read_block	9,73%	0,00%	1	8	FIFO	Stream	32	round_data_shift_U0	compress_data_U0	Link
wStream_channel_V_V_2	read_block	30,66%	0,00%	1	8	FIFO	Stream	32	round_data_shift_U0	compress_data_U0	Link
wStream_channel_V_V_3	read_block	51,58%	0,00%	1	8	FIFO	Stream	32	round_data_shift_U0	compress_data_U0	Link
wStream_done	read_block	0,61%	0,00%	1	2	FIFO	Stream	1	round_data_shift_U0	compress_data_U0	Link
internal_state_V	none	0,00%	0,00%	1	2	FIFO	TaskLevel	256	compress_data_U0	endian_U0	Link


dataStream_channel_V_0
Theme
Layout
dataStream_channel_V_0 (Dot Chart)
Channel Depth
Clock Cycles
normal
write_block
read_block
dataStream_channel_V_0 (Bar Chart)
Clock Cycles
Channel Depth
dataStream_channel_V_0 (Pie Chart)
0: 119 (14,48%)
1: 68 (8,27%)
11: 34 (4,14%)
12: 37 (4,50%)
13: 40 (4,87%)
14: 47 (5,72%)
15: 42 (5,11%)
16: 83 (10,10%)
17: 33 (4,01%)
other: 319 (38,81%)


wStream_channel_V_V_0
Theme
Layout
wStream_channel_V_V_0 (Dot Chart)
Channel Depth
1
0
0 20 40 60 80 100 120 140 160 180 200 220 240 260 280 300 320 340 360 380 400 420 440 460 480 500 520 540 560 580 600 620 640 660 680 700 720 740 760 780 800 820 840 860
Clock Cycles
normal
write_block
read_block
wStream_channel_V_V_0 (Bar Chart)
Clock Cycles
700
600
500
400
300
200
100
0
0
1
Channel Depth
wStream_channel_V_V_0 (Pie Chart)
0: 147 (17,88%)
1: 675 (82,12%)
0
1

Analisi dei Task (Simulazione)

- Messaggio da 600 caratteri
- Primo chunk impiega più tempo (1229 ns)
- I chunk restanti impiegano mono tempo (709 ns)



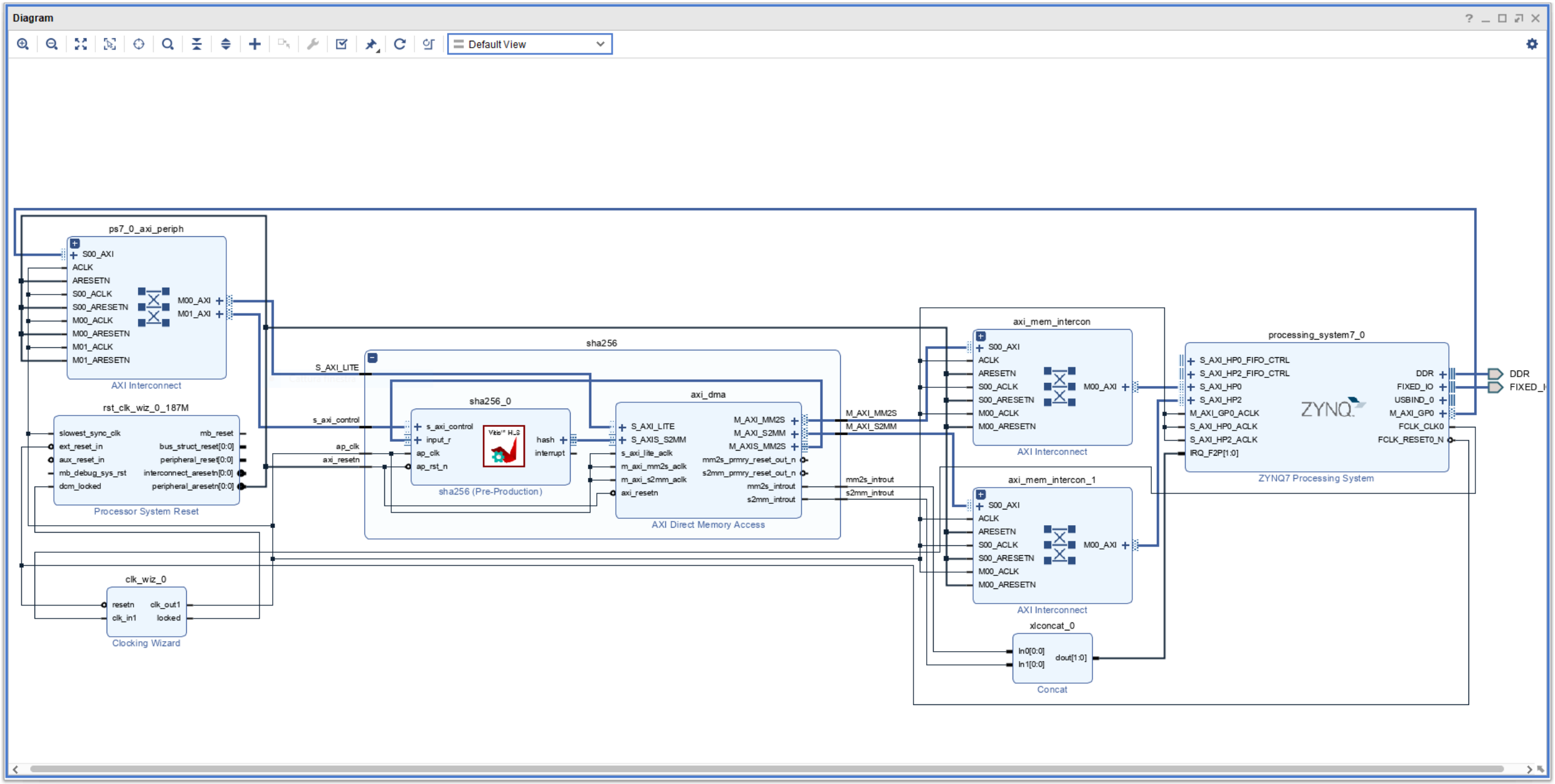


Diagram
Default View
ps7_0_axi_periph
S00_AXI
ACLK
ARESETN
S00_ACLK
S00_ARESETN
M00_ACLK
M00_ARESETN
M01_ACLK
M01_ARESETN
M00_AXI
M01_AXI
AXI Interconnect
rst_clk_wiz_0_187M
slowest_sync_clk
ext_reset_in
aux_reset_in
mb_debug_sys_rst
dcm_locked
mb_reset
bus_struct_reset[0:0]
peripheral_reset[0:0]
interconnect_aresetn[0:0]
peripheral_aresetn[0:0]
Processor System Reset
clk_wiz_0
resetn
clk_in1
clk_out1
locked
Clocking Wizard
sha256
S_AXI_LITE
s_axi_control
ap_clk
axi_resetn
sha256_0
input_r
ap_rst_n
hash
interrupt
sha256 (Pre-Production)
axi_dma
S_AXIS_S2MM
s_axi_lite_aclk
m_axi_mm2s_aclk
m_axi_s2mm_aclk
M_AXI_MM2S
M_AXI_S2MM
M_AXIS_MM2S
mm2s_prmry_reset_out_n
s2mm_prmry_reset_out_n
mm2s_introut
s2mm_introut
AXI Direct Memory Access
axi_mem_intercon
AXI Interconnect
axi_mem_intercon_1
AXI Interconnect
xlconcat_0
In0[0:0]
In1[0:0]
dout[1:0]
Concat
processing_system7_0
S_AXI_HP0_FIFO_CTRL
S_AXI_HP2_FIFO_CTRL
S_AXI_HP0
S_AXI_HP2
M_AXI_GP0_ACLK
S_AXI_HP0_ACLK
S_AXI_HP2_ACLK
IRQ_F2P[1:0]
DDR
FIXED_IO
USBIND_0
M_AXI_GP0
FCLK_CLK0
FCLK_RESET0_N
ZYNQ7 Processing System
DDR
FIXED_I

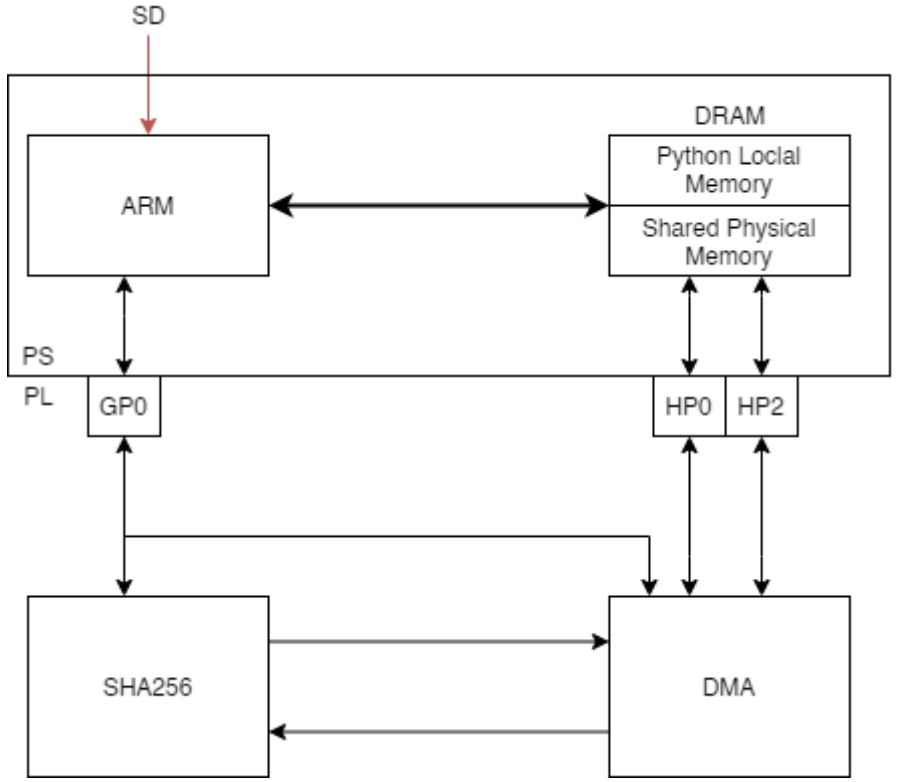
PYNQ
Caratteristiche Overlay

PS

1. Attivo Overlay tramite GP0
2. Caricamento file da microSD su memoria locale (lettura a blocchi)
3. Per ogni blocco
    1. Trasferimento da memoria locale a memoria contigua condivisa con DMA
    2. Trasmissione tramite HP0 alla DMA
    3. Attesa fine trasmissione
4. Dopo l'ultimo blocco invio un ultimo blocco contenette solo uno 0 segnalando la fine del file
5. Attesa output dalla porta HP2

PL

1. Per ogni blocco
    1. DMA reindirizza il blocco verso SHA256
    2. SHA256 aggiorna lo stato del hash e attende il blocco successivo
2. SHA256 - Attende lo 0 di fine messaggio
3. SHA256 invia l'hash al DMA che lo reindirizza verso la memoria condivisa tramite HP2



Driver Python - Prestazioni

File: 99 MB

Nome	Time [s]	Hash
Hashlib	2.8261	00A36691822E1309F95DF1283C4C26E351661943BC4F7E83323E53248776E9F6
mySha256 class	2.4248	00A36691822E1309F95DF1283C4C26E351661943BC4F7E83323E53248776E9F6
mySha256 fun	2.4081	00A36691822E1309F95DF1283C4C26E351661943BC4F7E83323E53248776E9F6
mySha256 2buf	1.7823	00A36691822E1309F95DF1283C4C26E351661943BC4F7E83323E53248776E9F6

File: 511 MB

function	Time [s]	Hash
Hashlib	24.7823	21BE956416A1669FA79B498827FB4D3F24F3CA4DA611523459C5166518459B4F
mySha256 class	25.5942	21BE956416A1669FA79B498827FB4D3F24F3CA4DA611523459C5166518459B4F
mySha256 fun	25.0967	21BE956416A1669FA79B498827FB4D3F24F3CA4DA611523459C5166518459B4F
mySha256 2buf	23.1919	21BE956416A1669FA79B498827FB4D3F24F3CA4DA611523459C5166518459B4F

Versione C - Produttore/Consumatore

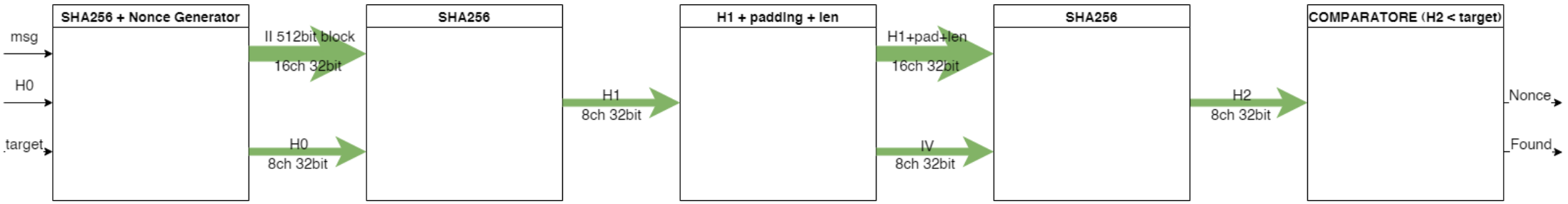
- Separazione dei compiti del PL
  - Produttore: Lettura file da microSD
  - Consumatore: Invio dati alla DMA
- Utilizzo di un Ring Buffer per ridurre ulteriormente la latenza tra i due
- Evito il passaggio per la memoria locale

Nome	T Produttore [s]	T Consumatore [s]
SHA256 – C++	22.8234	6.873

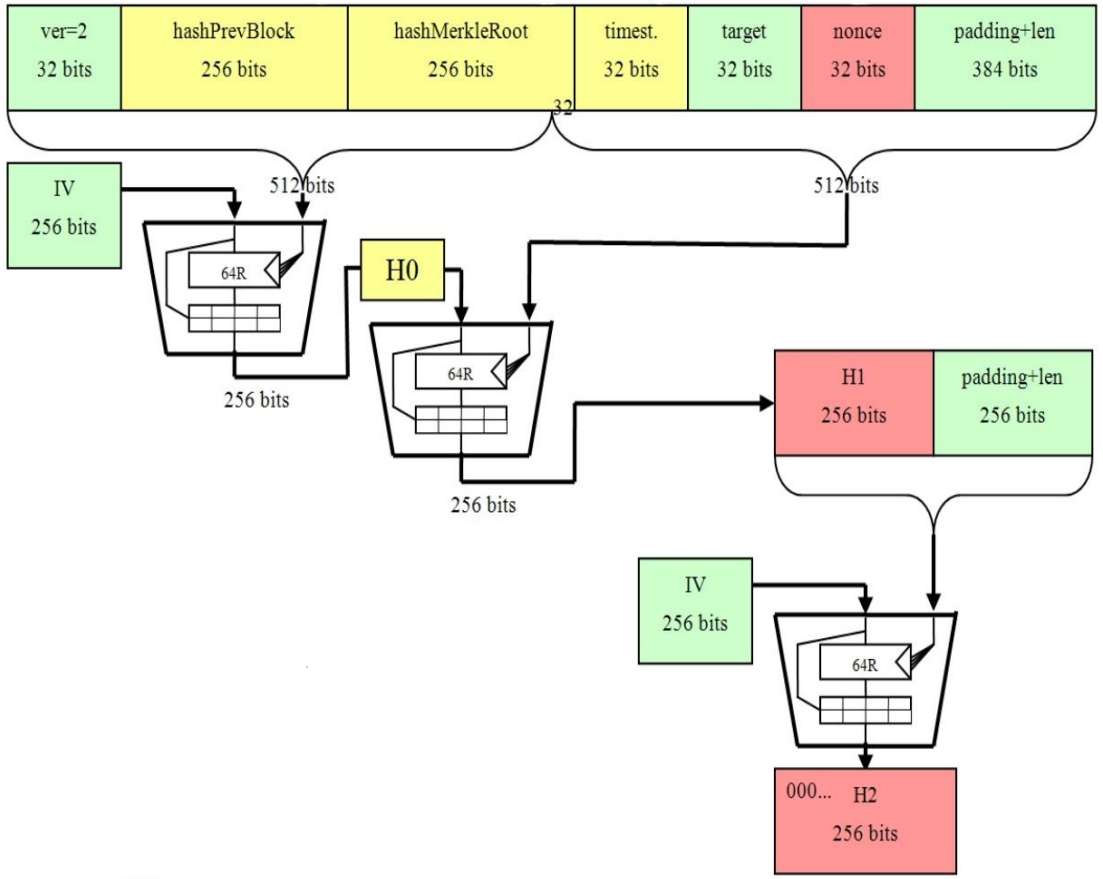
Attività progettuale

SHA256 per il mining

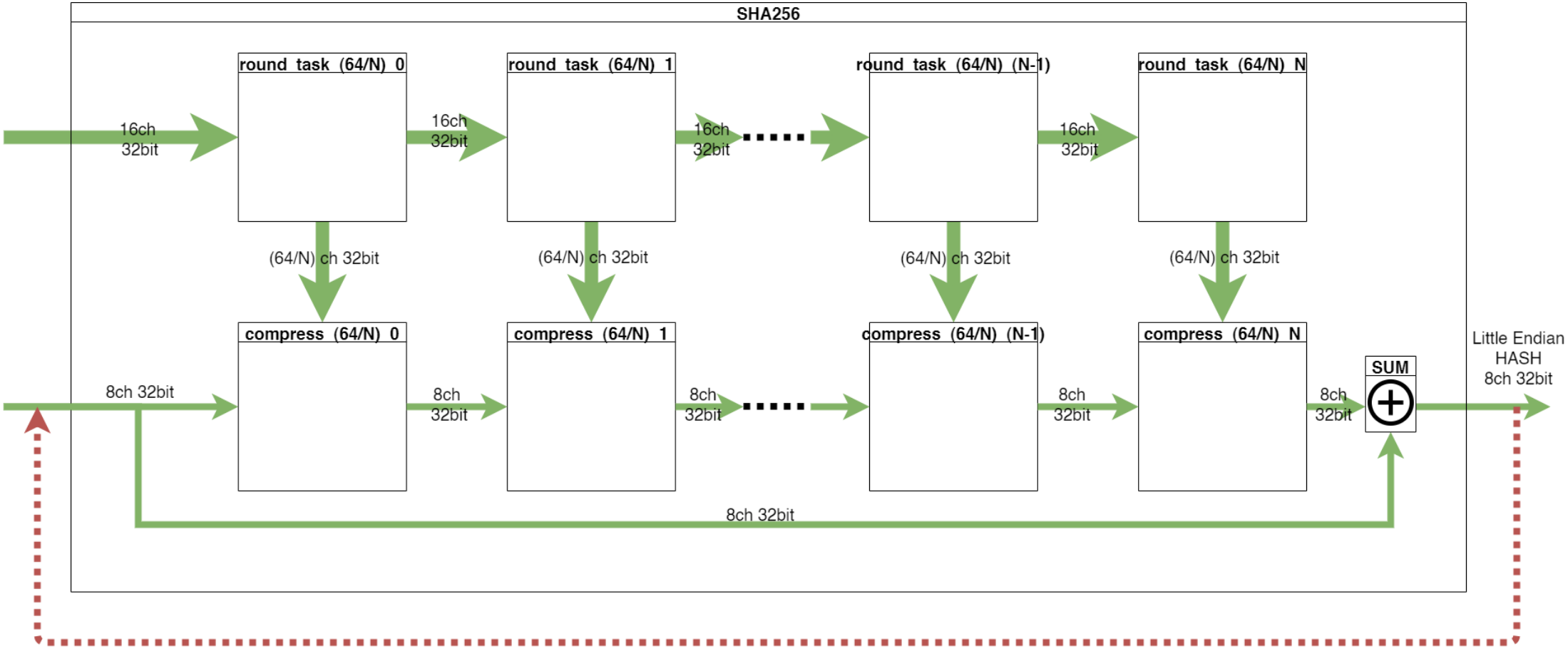
SHA256 Double



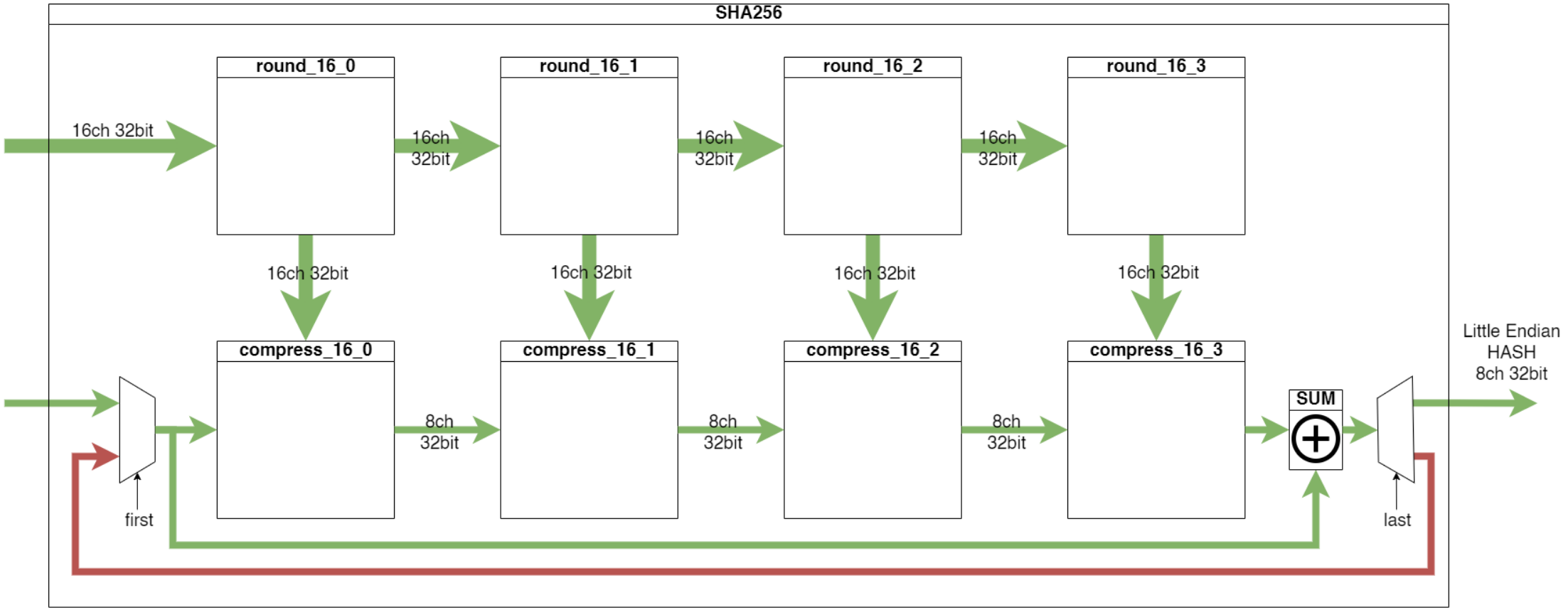
Ottimizzazione nota



Scomposizione in Sub-Task



Scomposizione in Sub-Task



Versione ad area minore