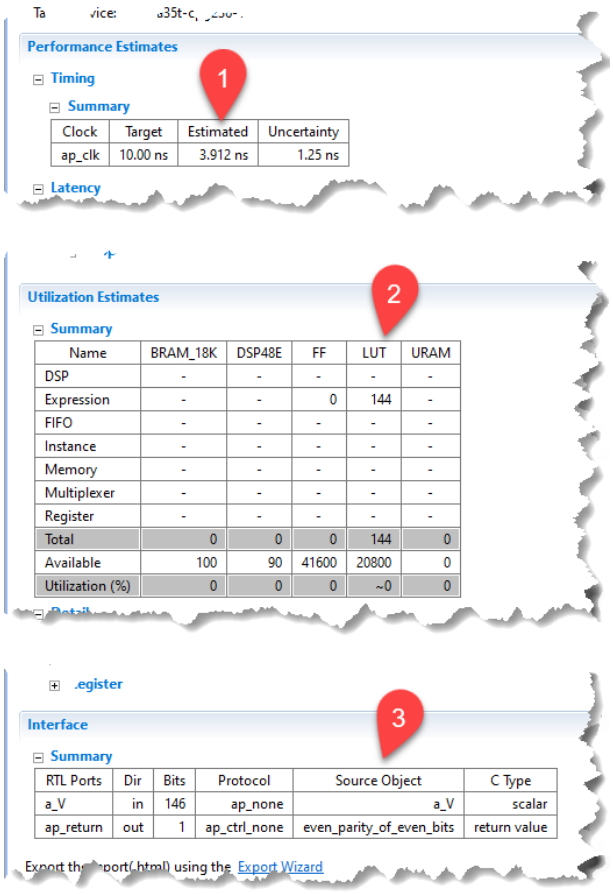
Exercise 1

The following code can implement the even parity of the even bits problem

```
#include <ap_int.h>
#define N 146

bool even_parity_of_even_bits(ap_uint<N> a) {
#pragma HLS INTERFACE ap_ctrl_none port=return
#pragma HLS INTERFACE ap_none port=a

	bool p = 0;
	for (int i = 0; i < N; i=i+2) {
#pragma HLS UNROLL
		p = p ^ a[i];
	}

	return p;
}
```

This figure shows parts of the high-level synthesis report



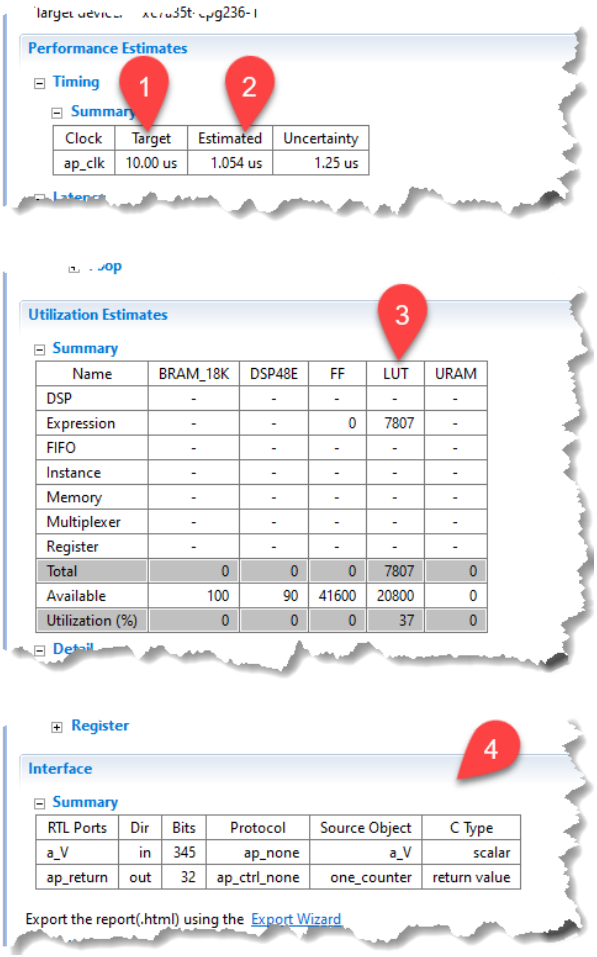
Exercise 2

The following code represents the one-counter design in HLS

```
#include <ap_int.h>

#define N 345

int one_counter(ap_uint<N> a) {
#pragma HLS INTERFACE ap_none port=a
#pragma HLS INTERFACE ap_ctrl_none port=return
	int counter = 0;

	for (int i = 0; i < N; i++) {
#pragma HLS UNROLL
		if (a[i] ==  1) {
			counter = counter + 1;
		} else {
			counter = counter + 0;
		}
	}

	return counter;
}
```

This figure shows parts of the high-level synthesis report.

Look at the design clock period constraint that it should be a large number such as *10000 ns*. The design propagation delay is about $1.054\ \mu s$. And the design utilises 7807 LUTs.



Performance Estimates

Timing

Summary

Clock	Target	Estimated	Uncertainty
ap_clk	10.00 us	1.054 us	1.25 us

Utilization Estimates

Summary

Name	BRAM_18K	DSP48E	FF	LUT	URAM
DSP	-	-	-	-	-
Expression	-	-	0	7807	-
FIFO	-	-	-	-	-
Instance	-	-	-	-	-
Memory	-	-	-	-	-
Multiplexer	-	-	-	-	-
Register	-	-	-	-	-
Total	0	0	0	7807	0
Available	100	90	41600	20800	0
Utilization (%)	0	0	0	37	0

Register

Interface

Summary

RTL Ports	Dir	Bits	Protocol	Source Object	C Type
a_V	in	345	ap_none	a_V	scalar
ap_return	out	32	ap_ctrl_none	one_counter	return value

Export the report(.html) using the Export Wizard

Exercise 3

This code shows the HLS description for a combinational circuit that finds the even-parity bit in a 234-bit unsigned integer number.

```
#include <ap_int.h>

#define N 234
bool  even_parity_234(ap_uint<N> a) {
#pragma HLS INTERFACE ap_none port=a
#pragma HLS INTERFACE ap_ctrl_none port=return

      bool p = a.xor_reduce();
      return p;


}
```