The function accumulates the numbers that it receives on each execution.

We can consider the following top function and test bench codes to examine the functionality.

```
void basic_pointer_quiz(ap_uint<16> *d) {
#pragma HLS INTERFACE mode=ap_hs port=d
#pragma HLS INTERFACE mode=ap_ctrl_hs port=return

    static ap_uint<16> s = 0;
    s += *d;
    *d = s;
}
```

```
int main() {
    int         status = 0;
    ap_uint<16> d = 0;
    int         sum_sw = 0;

    for (int i = 0; i < 10; i++) {
        d = i;
        std::cout << " Current input is = " << d ;
        basic_pointer_quiz(&d);
        std::cout << "   Total sum is = " << d << std::endl;
        sum_sw += i;
        if (sum_sw != d) {
            status = -1;
            std::cout << " Error: total sum is = " << d
                      << "should be " << sum_sw << std::endl;
            break;
        }
    }

    if (status == 0) { std::cout << "  TEST: SUCCESSFUL " <<  std::endl;}
    else { std::cout << "  TEST: FAILED     " <<  std::endl; }

    return status;
}
```

Then the output would be as follows.

```
INFO: [SIM 2] *************** CSIM start ***************
INFO: [SIM 4] CSIM will launch GCC as the compiler.
make: 'csim.exe' is up to date.
 Current input is = 0   Total sum is = 0
 Current input is = 1   Total sum is = 1
 Current input is = 2   Total sum is = 3
 Current input is = 3   Total sum is = 6
 Current input is = 4   Total sum is = 10
 Current input is = 5   Total sum is = 15
 Current input is = 6   Total sum is = 21
 Current input is = 7   Total sum is = 28
 Current input is = 8   Total sum is = 36
 Current input is = 9   Total sum is = 45
  TEST: SUCCESSFUL
INFO: [SIM 1] CSim done with 0 errors.
INFO: [SIM 3] *************** CSIM finish ***************
```

As can be seen, the function receives a number on each invocation and accumulates them in a static variable.

The following figure shows the Vivado design to represent its functionality on the actual FPGA.