## Design Project:

# Implementation of a JPEG Encoding / Decoding Scenario in SystemC

**Joachim Gerlach** <gerlach@informatik.uni-tuebingen.de>

**University of Tübingen**
**Wilhelm-Schickard-Institut**
**Department of Computer Engineering**

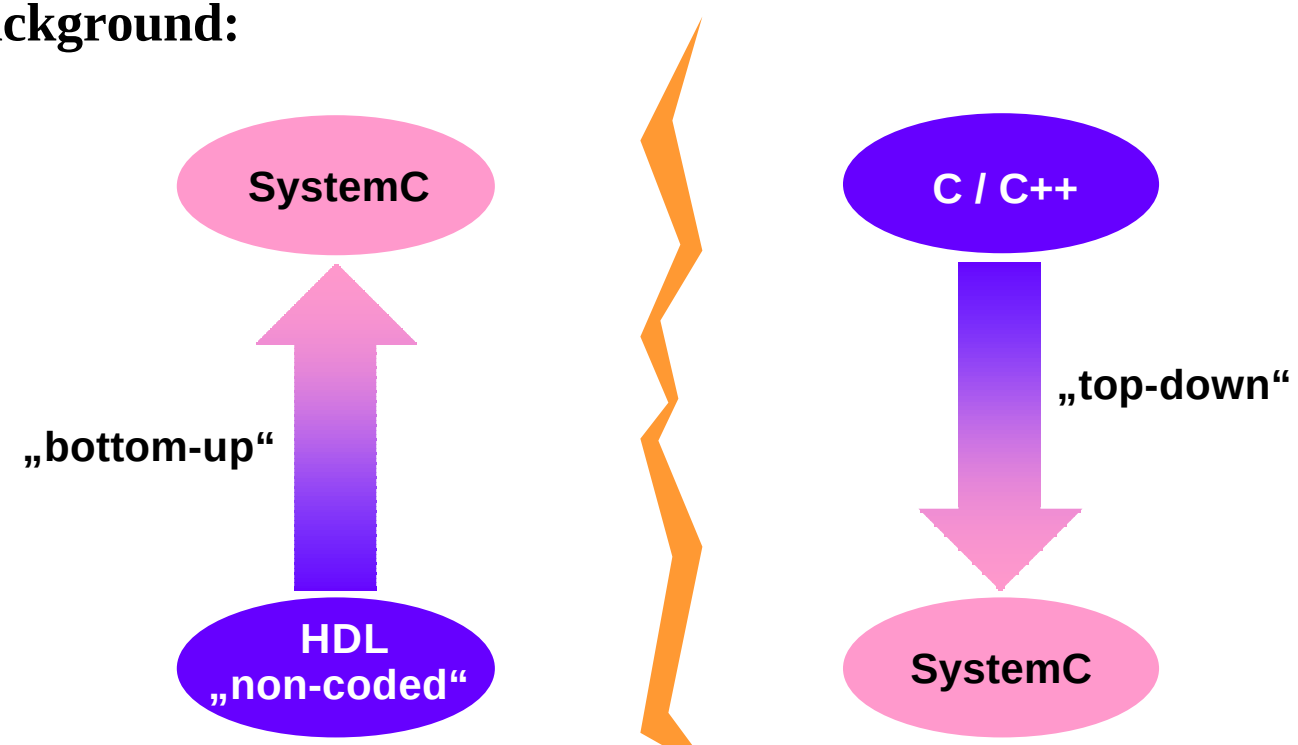
University of Tübingen

Department of Computer Engineering

Joachim Gerlach

# SystemC Design Example

- **Background:**




**Transforming HDL into SystemC**
**Creating new designs in SystemC**

**Transforming C/C++ into SystemC**
- **Many algorithms exist in C/C++**
- **Many standardization committees (e.g., ISO) use C specifications**

University of Tübingen

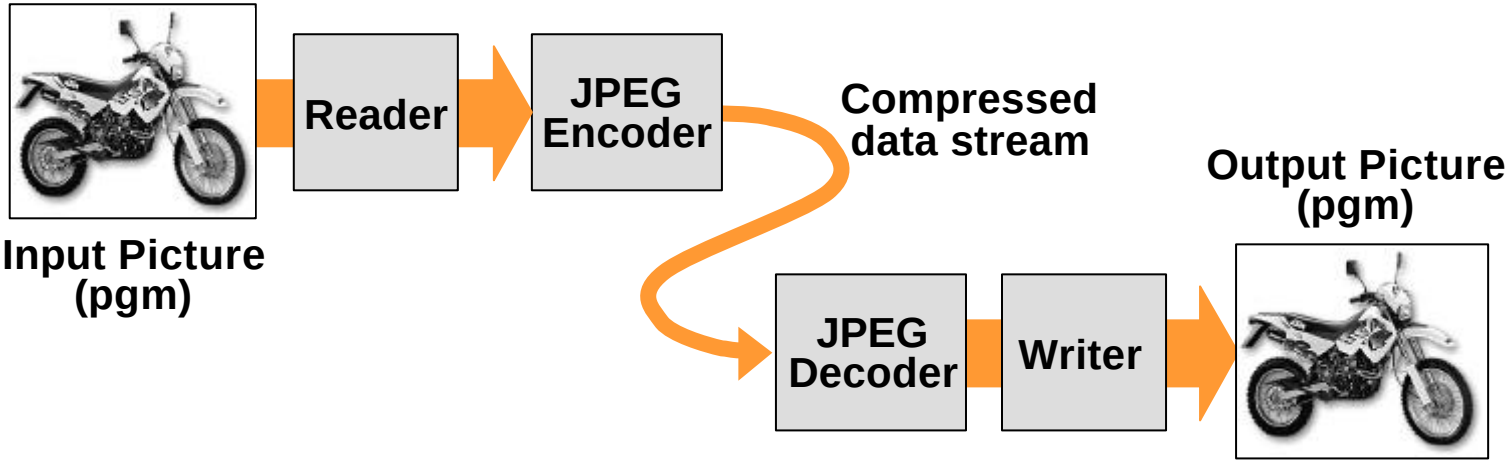
Department of Computer Engineering

Joachim Gerlach

# SystemC Design Example

- **Application:**

  **JPEG compression and decompression stream**



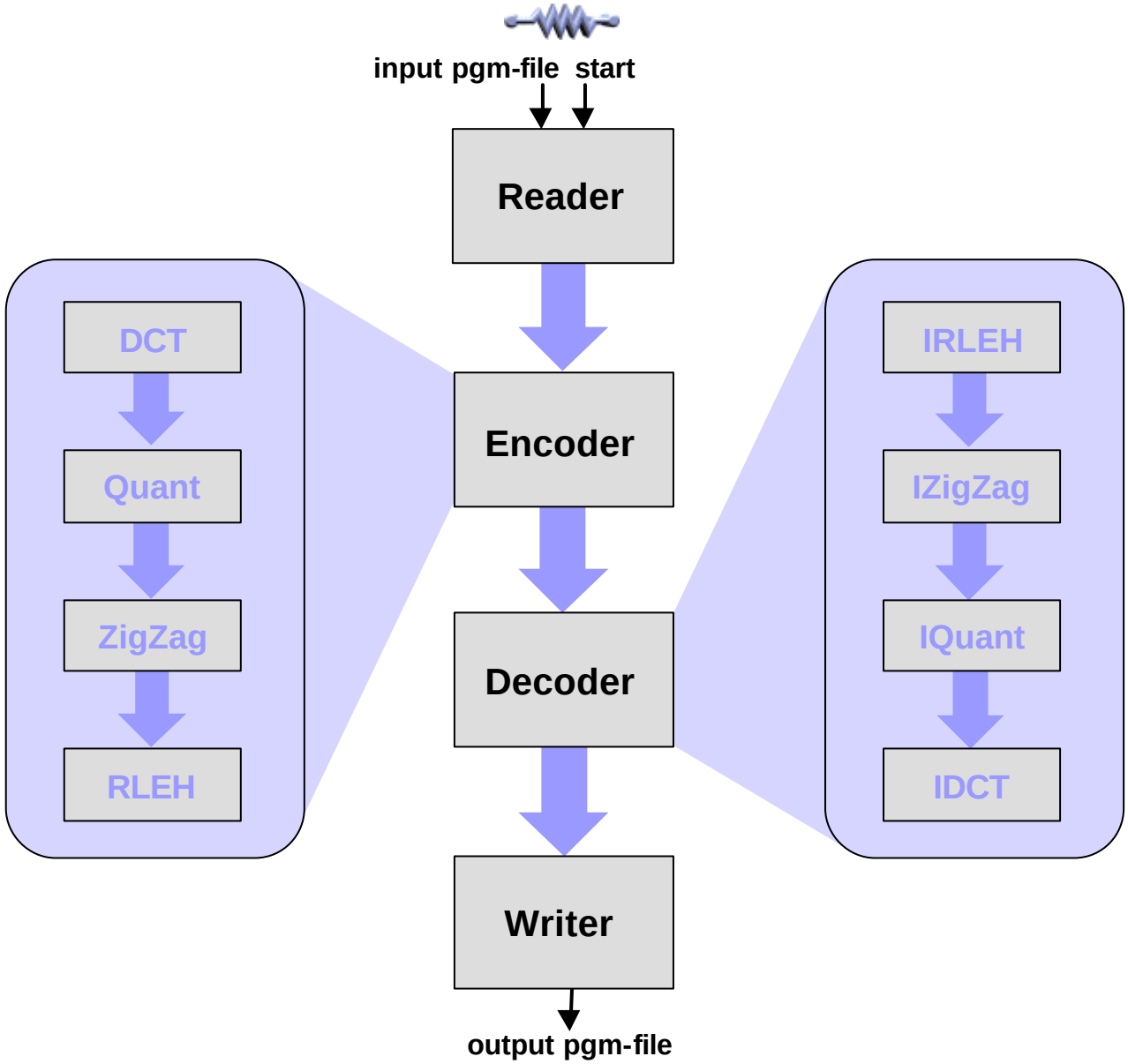

- **Reference implementation:**
  - **16 modules, approx. 950 lines of C++ code**
  - **by T. Thissenhusen, TU Dresden, Germany**

University of Tübingen

Department of Computer Engineering

Joachim Gerlach

# SystemC Design Example

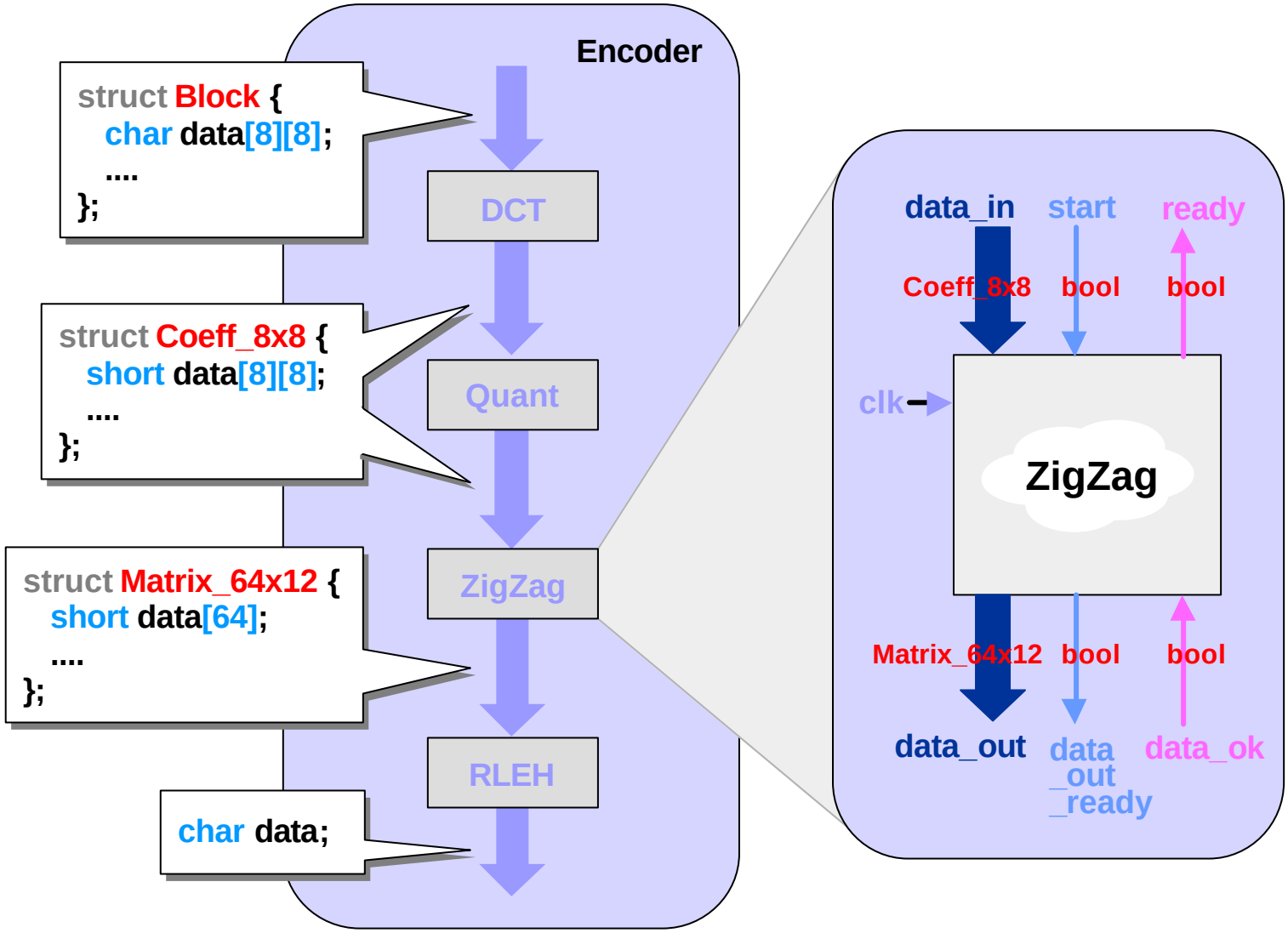


University of Tübingen

Department of Computer Engineering

Joachim Gerlach

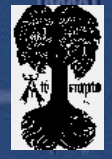
# SystemC Design Example



University of Tübingen

Department of Computer Engineering

Joachim Gerlach



# SystemC Design Example

```
#include <systemc.h>
#include "global.h"


SC_MODULE(zigzag)  {
  sc_in_clk                 clk;
  sc_in<Coeff_8x8>          data_in;
  sc_in<bool>               start;
  sc_in<bool>               data_ok;
  sc_out<Matrix_64x12>      data_out;
  sc_out<bool>              ready;
  sc_out<bool>              data_out_ready;
  void do_zigzag();
  SC_CTOR(zigzag) {
    SC_CTHREAD(do_zigzag,clk.pos());
  }
};
```

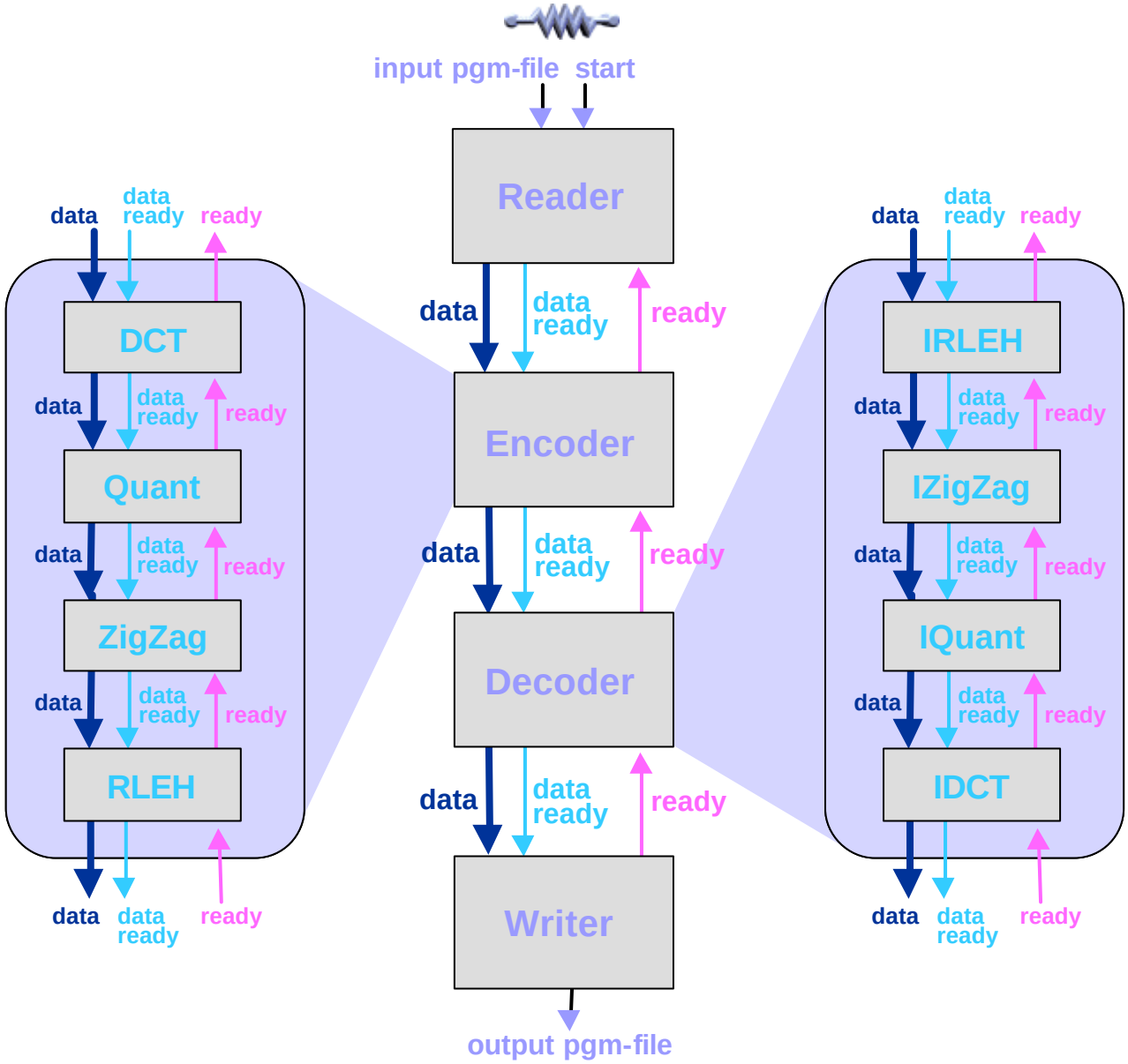
```
void zigzag::do_zigzag() {
  Coeff_8x8         fuv;
  Matrix_64x12      result;
  unsigned char     u, v, a, dir;
  while(true)  {
    ready.write(true);
    data_out_ready.write(false);
    wait_until(start.delayed()==true);
    ready.write(false);
    fuv = data_in.read();
    // zigzag
    u = 0; v = 0;
    dir = 1;    // dir == 1: upwards, dir == 0: downwards
    for ( a = 0; a < 64; a++ )  {
      result.put ( a, (WORD) (fuv.get (v,u) ) );
      if ( v == 0 )
        if ( dir )   { u++; dir = 0; }
        else         { u--; v++; }
      else if ( v == 7 )
        if ( !dir )  { u++; dir = 1; }
        else         { u++; v--; }
      else if ( u == 0 )
        if ( !dir )  { v++; dir = 1; }
        else         { u++; v--; }
      else if ( u == 7 )
        if ( dir )   { v++; dir = 0; }
        else         { u--; v++; }
      else
        if ( dir )   { u++; v--; }
        else         { u--; v++; }
    }
    data_out.write(result);
    data_out_ready.write(true);
    wait_until(data_ok.delayed()==true);
  }
}
```

University of Tübingen

Department of Computer Engineering

Joachim Gerlach

# SystemC Design Example



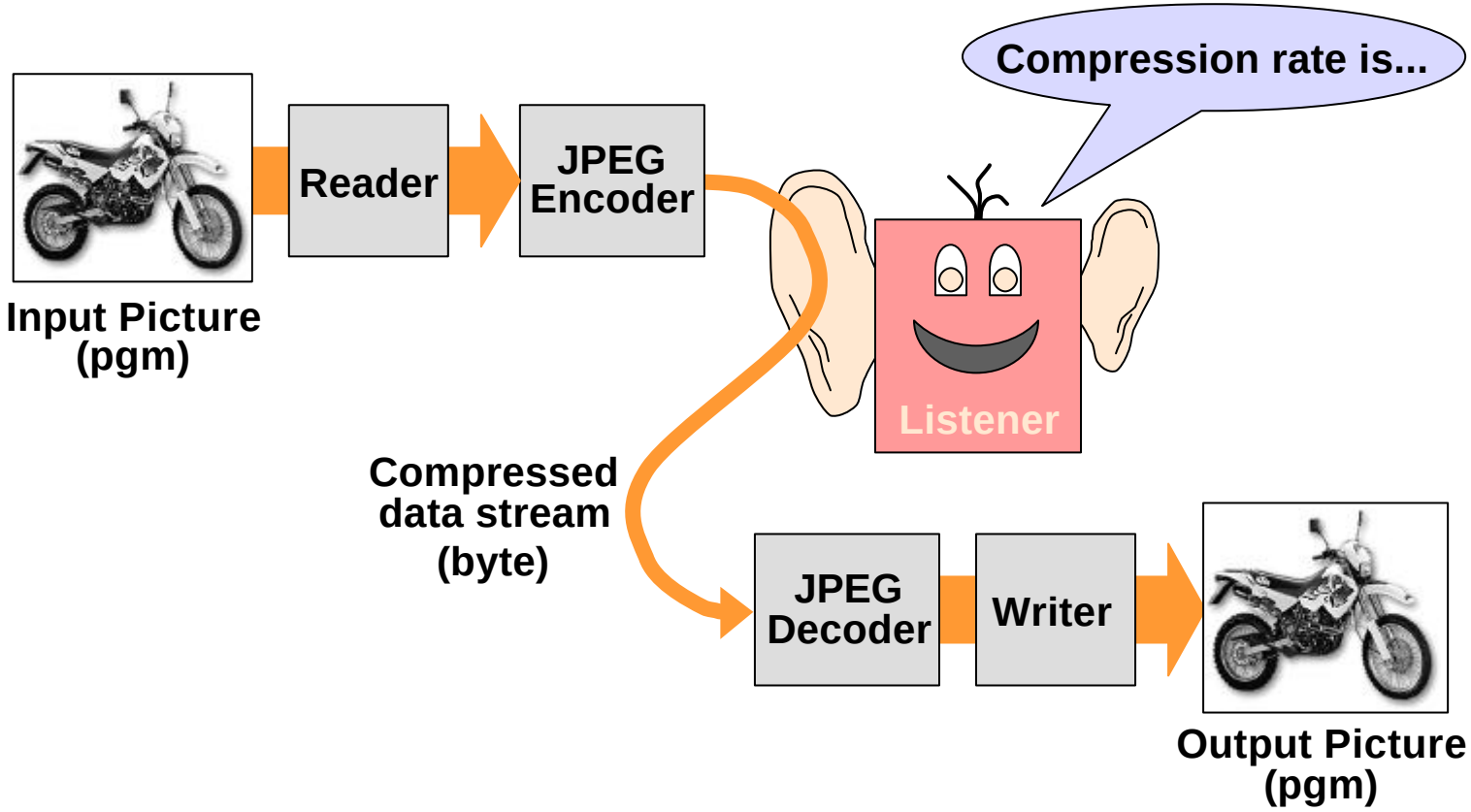
University of Tübingen

Department of Computer Engineering

Joachim Gerlach

# SystemC Design Example

- Adding a "Listener" component:



University of Tübingen

Department of Computer Engineering

Joachim Gerlach

# SystemC Design Example

```
#include "global.h“
#include <systemc.h>


SC_MODULE ( listener ) {
  sc_in_clk          clk;
  sc_in<BYTE>        data_in;
  sc_in<bool>        start;
  sc_in<unsigned>    xsize_in;
  sc_in<unsigned>    ysize_in;
  void do_listener();
  SC_CTOR (listener)  {
    SC_CTHREAD ( do_listener, clk.pos() );
  }
};
```

```
void channel::do_listener()  {
  static int no_bytes = 0;
  static int no_packages = 0;
  BYTE data;
  while ( true )  {
    wait_until ( start.delayed() == true );
    data = data_in.read();
    no_bytes += (int) data;
    no_bytes++;  // length info bit
    no_packages++;
    if ( no_packages == ( xsize_in.read() * ysize_in.read() ) )  {
      cout << endl << "  input picture [bytes]     : “;
      cout << xsize_in.read() * ysize_in.read() * 64 << endl;
      cout << "  compressed stream [bytes] : “;
      cout << no_bytes << endl << endl;
    }
  }
}
```