Cyclic Redundancy Check (CRC) example project

3.0

Features

- Generation CRC input signals same as clock, reset.
- Hardware Reset generation for CRC component.
- Generation CRC sequence by CRC component and Software CRC sequence calculation.
- Software and hardware CRC sequence on LCD are displayed.
- Possibility of the CRC Component input signals handling.

General Description

This example project demonstrates the CRC component with default configuration operation. Software CRC calculation is executed for visual values comparison. Hardware CRC calculations are displayed on the LCD step by step over defined (SEED_VALUE_COUNT – default is 20) steps. Software CRC calculation is displayed on the LCD at once.

Development kit configuration

1. This project is written for 2X16 display as the one available on Cypress kit CY8CKIT-001.
2. The DVK CY8CKIT-001 board besides default configuration should have LCD power jumper (J12).
3. Build the project and program the hex file on to the target device using MiniProg3.
4. Power cycle the device and observe the results on the LCD.

Project configuration

This project consists of a 16 bits CRC, two Control Registers, Character LCD components and three digital output pins. The top design schematic is shown in Figure 1. The Clock signal is generated by the control register "Clock_Gen". The Reset signal is generated by the control register "Reset_Gen". All signals changes can be observed on an oscilloscope or a signal analyzer at PORT0 [2:0] by default.

Software Defines on main.c file:

SEED_VALUE_COUNT - set number of CRC calculate iteration (20 by default);

DISP_DELAY - set delay of information on the LCD (1000 by default);

in - input data signal (1 by default for software CRC calculation).

LCD views:

SV = Software calculated CRC value.

Calc = Hardware calculated CRC iteration at this time.

CRC = Hardware calculated CRC value for current iteration.

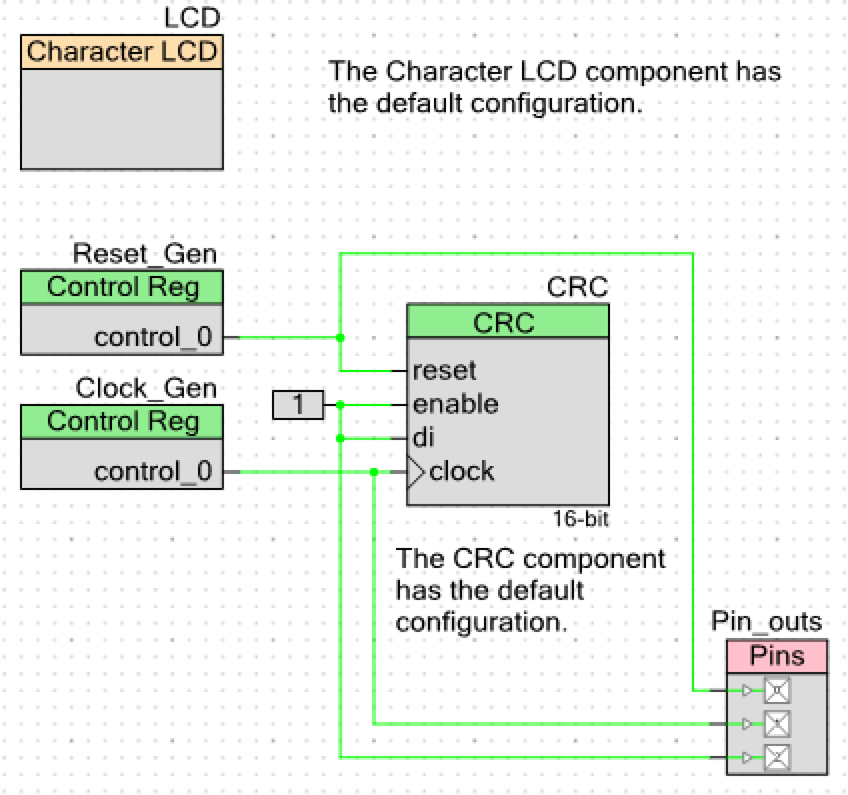
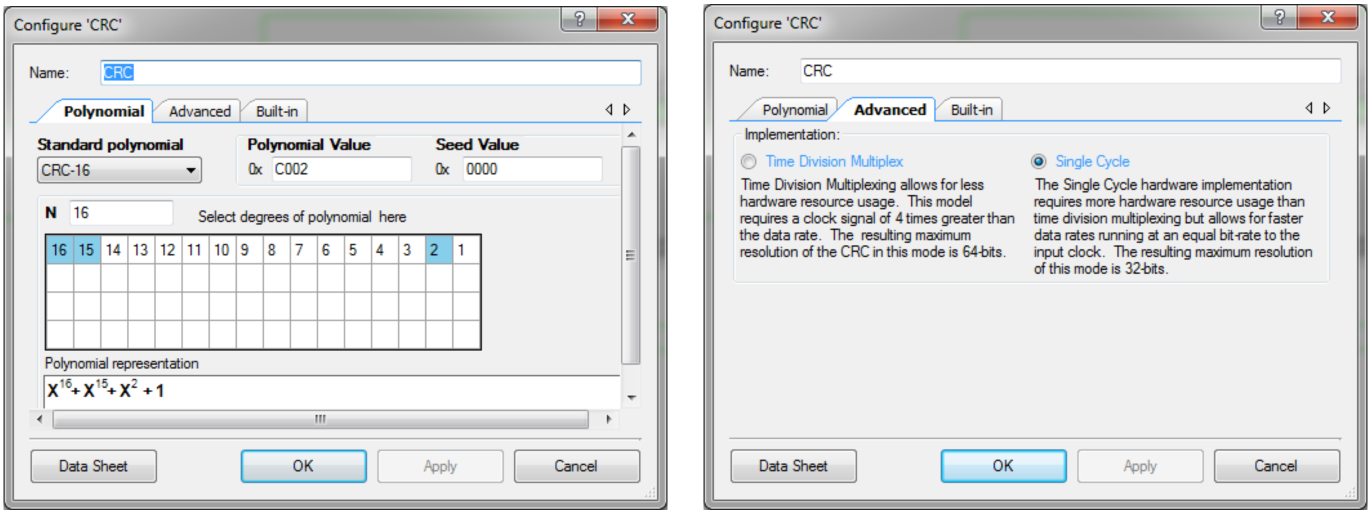


Figure 1. Top design schematic.

The Character LCD Component has the default configuration. The CRC Component configuration (Figure 2):

- Standard Polynomial - CRC-16
- Resolution - 16 bits
- Implementation - Single Cycle
- Polynomial Value - 0xC002
- Seed Value - 0x0000



Name: CRC

Standard polynomial: CRC-16

Polynomial Value: 0x C002

Seed Value: 0x 0000

N: 16

Select degrees of polynomial here

16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

Polynomial representation

$X^{16}+X^{15}+X^{2}+1$

Name: CRC

Implementation:

○ Time Division Multiplex

Time Division Multiplexing allows for less hardware resource usage. This model requires a clock signal of 4 times greater than the data rate. The resulting maximum resolution of the CRC in this mode is 64-bits.

● Single Cycle

The Single Cycle hardware implementation requires more hardware resource usage than time division multiplexing but allows for faster data rates running at an equal bit-rate to the input clock. The resulting maximum resolution of this mode is 32-bits.

Figure 2. CRC Component Configuration Window

Project description

The CRC Component consistently executes some calculation (SEED_VALUE_COUNT define). Its value is displayed on the LCD and it is named CRC. The software calculated value outputs on the LCD for verification (SV). After calculation is finished the reset is executed. The values after reset are displayed on LCD and are named CRC.

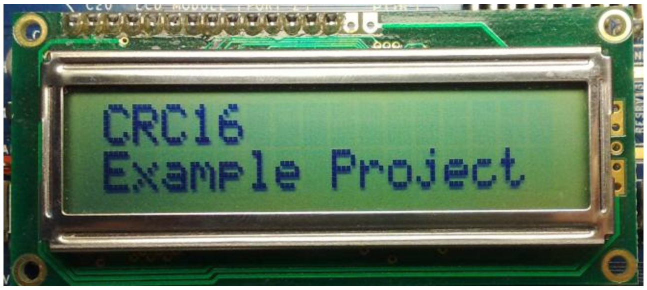
Expected results

Display legend is described below:

First string: “SV=XXXX Calc=YY”
SV – Software calculated, XXXX – Software calculated CRC value, Calc – The current hardware CRC calculation iteration, YY – value of the current hardware CRC calculation iteration.

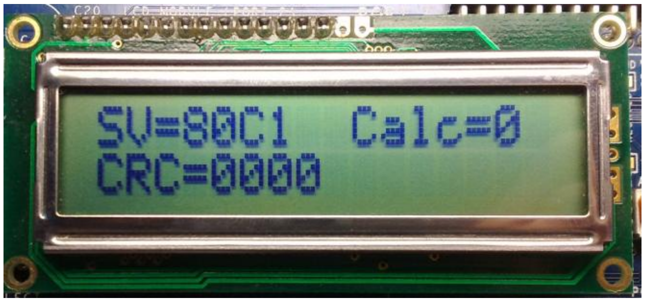
Second string: “CRC=ZZZZ ”
CRC – CRC component, ZZZZ – current iteration hardware calculated value of the CRC component.

1)



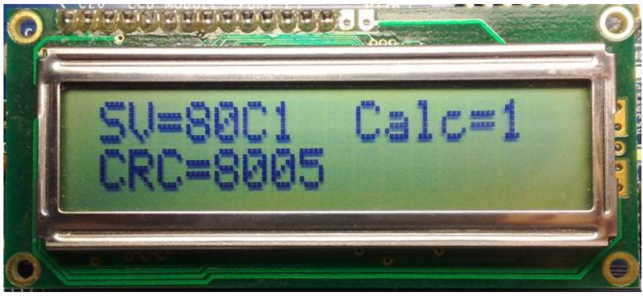
```
CRC16
Example Project
```

2)



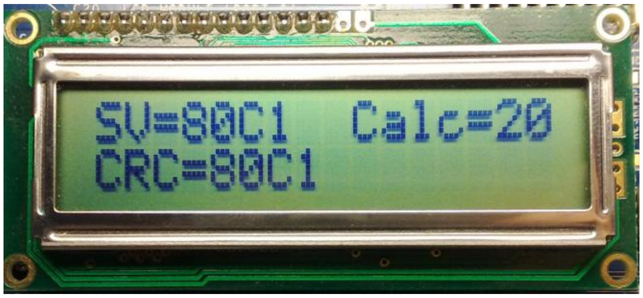
```
SV=80C1   Calc=0
CRC=0000
```

3)



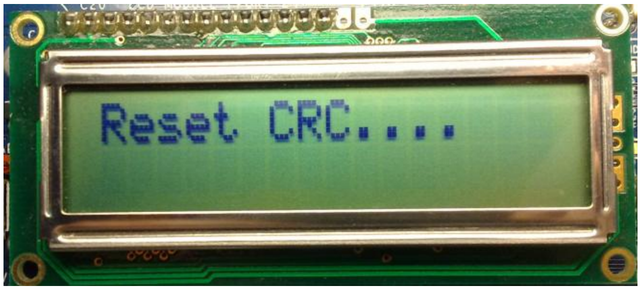
```
SV=80C1   Calc=1
CRC=8005
```

4)



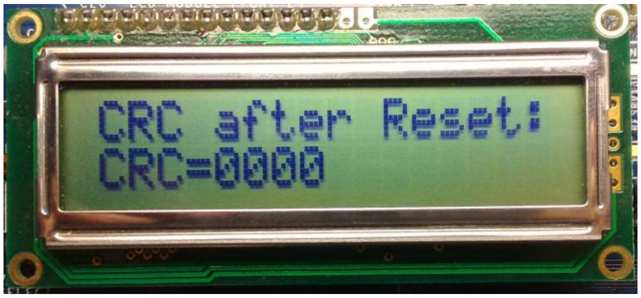
```
SV=80C1   Calc=20
CRC=80C1
```

5)



```
Reset CRC....
```

6)



```
CRC after Reset:
CRC=0000
```

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.