**void** **main**(**void**)

{

 WDTCTL = WDTPW | WDTHOLD; // stop watchdog timer

 RCC\_Init();

 PM5CTL0 &= ~LOCKLPM5; //PMM\_unlockLPM5();

 GPIO\_Init();

 Ecomps\_Init();

 twobus\_InitMessage();

 P1OUT|=BIT3|BIT4|BIT2; //set it to high

P1OUT&=~BIT5;

TIM\_B\_Init();

**\_\_bis\_SR\_register**(GIE);

**for** (;;)

 {

 }

}

**static** **void** **Ecomps\_Init**(**void**)

{

 GPIO\_setAsPeripheralModuleFunctionOutputPin(GPIO\_PORT\_P1,GPIO\_PIN0,GPIO\_TERNARY\_MODULE\_FUNCTION);

 GPIO\_setAsPeripheralModuleFunctionOutputPin(GPIO\_PORT\_P1,GPIO\_PIN1,GPIO\_TERNARY\_MODULE\_FUNCTION);

 EComp\_initParam param = {0};

 param.positiveTerminalInput = ECOMP\_INPUT\_0;

 param.negativeTerminalInput = ECOMP\_INPUT\_1;

 param.outputFilterEnableAndDelayLevel = ECOMP\_FILTER\_DELAY\_OFF;

 param.invertedOutputPolarity = ECOMP\_NORMAL\_OUTPUT\_POLARITY;

 EComp\_init(ECOMP0\_BASE, &param);

 EComp\_setInterruptEdgeDirection(ECOMP0\_BASE, ECOMP\_OUTPUT\_INTERRUPT\_RISING\_EDGE | ECOMP\_OUTPUT\_INTERRUPT\_FALLING\_EDGE);

 EComp\_clearInterrupt(ECOMP0\_BASE, ECOMP\_OUTPUT\_INTERRUPT\_FLAG |ECOMP\_INVERTED\_POLARITY\_INTERRUPT\_FLAG);

 EComp\_enableInterrupt(ECOMP0\_BASE, ECOMP\_OUTPUT\_INTERRUPT |ECOMP\_INVERTED\_POLARITY\_INTERRUPT);

 EComp\_selectHysteresisMode(ECOMP0\_BASE,ECOMP\_HYSTERESIS\_MODE\_DISABLE);//ECOMP\_HYSTERESIS\_MODE\_DISABLE);//ECOMP\_HYSTERESIS\_MODE\_30MV);

 EComp\_enable(ECOMP0\_BASE);

}

**static** **void** **TIM\_B\_Init**(**void**)

{

 Timer\_B\_clearTimerInterrupt(TIMER\_B0\_BASE);

 Timer\_B\_outputPWMParam param1 = {0};

 param1.clockSource=TIMER\_B\_CLOCKSOURCE\_SMCLK;

 param1.clockSourceDivider=TIMER\_B\_CLOCKSOURCE\_DIVIDER\_8;

 param1.compareOutputMode=TIMER\_B\_OUTPUTMODE\_SET\_RESET;

 param1.compareRegister=TIMER\_B\_CAPTURECOMPARE\_REGISTER\_2;

 param1.timerPeriod=330;

 param1.dutyCycle=165;

 Timer\_B\_outputPWM(TIMER\_B0\_BASE,&param1);

 GPIO\_setAsPeripheralModuleFunctionOutputPin(GPIO\_PORT\_P1, GPIO\_PIN7,GPIO\_SECONDARY\_MODULE\_FUNCTION);

}

**#if** defined(\_\_TI\_COMPILER\_VERSION\_\_) || defined(\_\_IAR\_SYSTEMS\_ICC\_\_)

**#pragma** vector=ECOMP0\_VECTOR

**\_\_interrupt** **void** **ECOMP0\_ISR**(**void**)

**#elif** defined(\_\_GNUC\_\_) **void** **\_\_attribute\_\_** ((**interrupt**(ECOMP0\_VECTOR))) ECOMP0\_ISR (**void**)

**#else**

**#error** Compiler not supported!

**#endif**

{uint8\_t wHigh\_VAL=0;

 **switch**(**\_\_even\_in\_range**(CPIV, CPIV\_\_CPIIFG))

{

**case CPIV\_\_CPIFG:**

**EComp\_clearInterrupt(ECOMP0\_BASE,ECOMP\_OUTPUT\_INTERRUPT\_FLAG);**

**break**;

**case** CPIV\_\_CPIIFG:

**EComp\_clearInterrupt(ECOMP0\_BASE,ECOMP\_INVERTED\_POLARITY\_INTERRUPT\_FLAG);**

 **break**;

**default**: **break**;

}

}