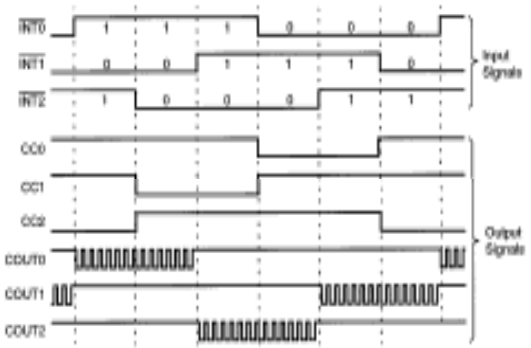
XC800 Product Presentation



May 2006

Capture Compare Unit
CC6

Never stop thinking.

Table of contents

Target Application for XC8xx – Electronic Motor Control

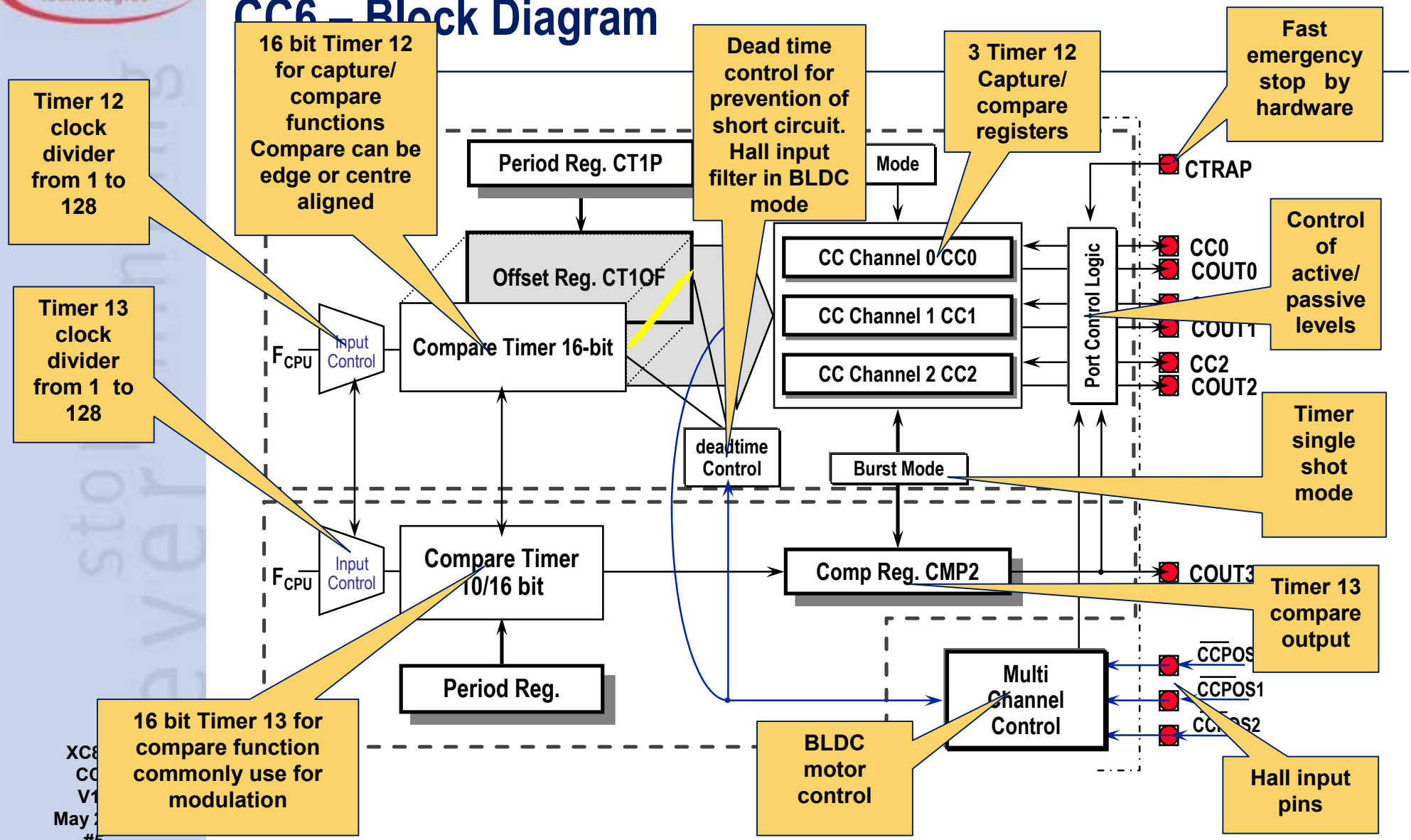
- Drives need realtime performance
  - control loop must run faster than 2-4 PWM periods (e.g. 100-200us)
  - CPU performance is valuable and must be saved for key tasks
  - Question: How to offload the CPU?
  - Answer: Build intelligent and autonomous peripherals!

- CC6 in a Drive application:
  - generate PWM patterns for all kind of motors
  - operate always in a safe state – even in an error condition
  - interact with ADC for sensorless control of motors

- CC6 is used intensively – the more it works autonomous the more CPU load can be saved for control algorithms

Table of contents

CC6 – Block Diagram



CC6 – The Register Map

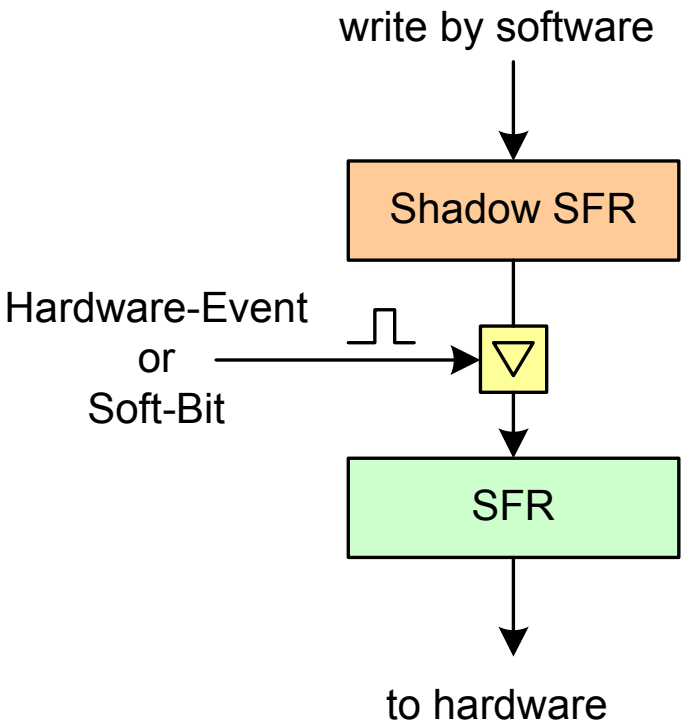
- The CC6 has 61 SFRs in 4 SFR pages
- Switching between pages for configuration
- Running from a working page after configuration

Page 0	Page 1	Page 2	Page 3
CC63SRL	CC63RL	T12MSELL	MCMOUTL
CC63SRH	CC63RH	T12MSELH	MCMOUTH
TCTR4L	T12PRL	IENL	ISL
TCTR4H	T12PRH	IENH	ISH
MCMOUTSL	T13PRL	INPL	PISEL0L
MCMOUTSH	T13PRH	INPH	PISEL0H
ISRL	T12DTCL	ISSL	PISEL2
ISRH	T12DTCH	ISSH	–
CMPMODIFL	TCTR0L	PSLR	–
CMPMODIFH	TCTR0H	MCMCTR	–
CC60SRL	CC60RL	TCTR2L	T12L
CC60SRH	CC60RH	TCTR2H	T12H
CC61SRL	CC61RL	MODCTRL	T13L
CC61SRH	CC61RH	MODCTRH	T13H
CC62SRL	CC62RL	TRPCTRL	CMPSTATL
CC62SRH	CC62RH	TRPCTRH	CMPSTATH

CC6 – Shadow Register

Shadow Register

- All SFRs which have to be updated in runtime (realtime) are shadowed
- Software (CPU) accesses are decoupled from hardware
- Safe operation in all conditions
- Fully utilized PWM (0-100%)
- Hardware event controls shadow transfer



Result Control

EX: CC6_1/2

CC6 – Single Shot and Synchronization

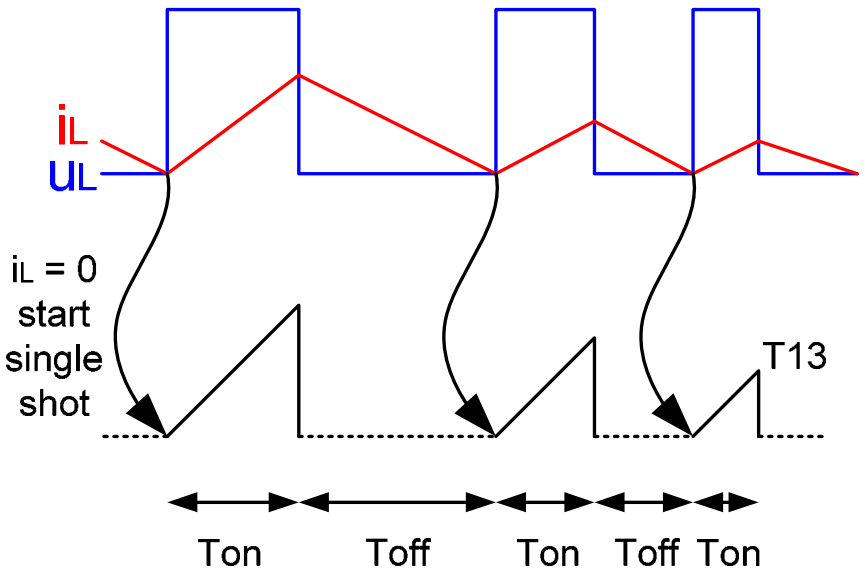
Single Shot and Synchronization

- T12 and T13 can run in single shot mode
- T12 and T13 can be started on several events
  - external start – T12HR/T13HR
  - period/compare match of the other timer

➢ Synchronization T13 on T12

➢ Inductive Load Control

➢ Trigger ADC



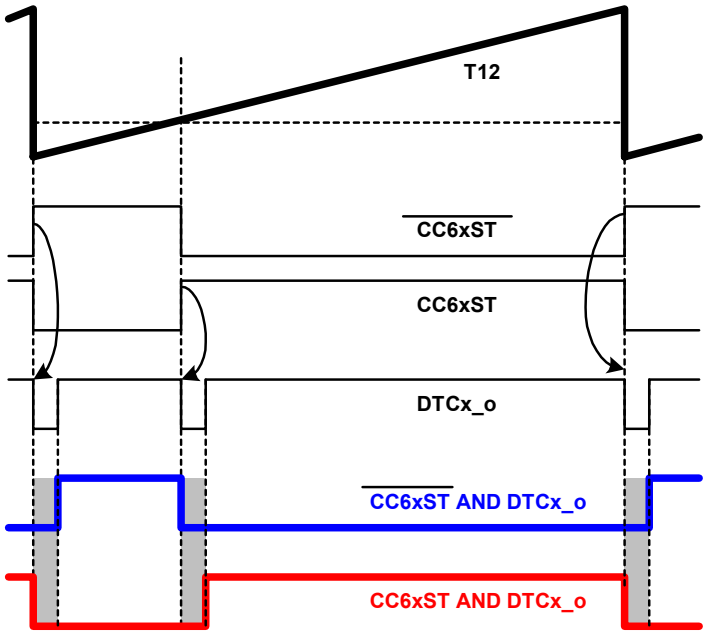
Lampballast, SMPS, PFC, Current Measurement

EX: CC6_3/4

CC6 – Deadtime Generation

Deadtime Generation

- Dead time generation on T12 Channels
  - by programmable dead time counter DTC
- Dead time feature can be used
  - in edge and center aligned mode
  - together with 0% - 100% duty cycle



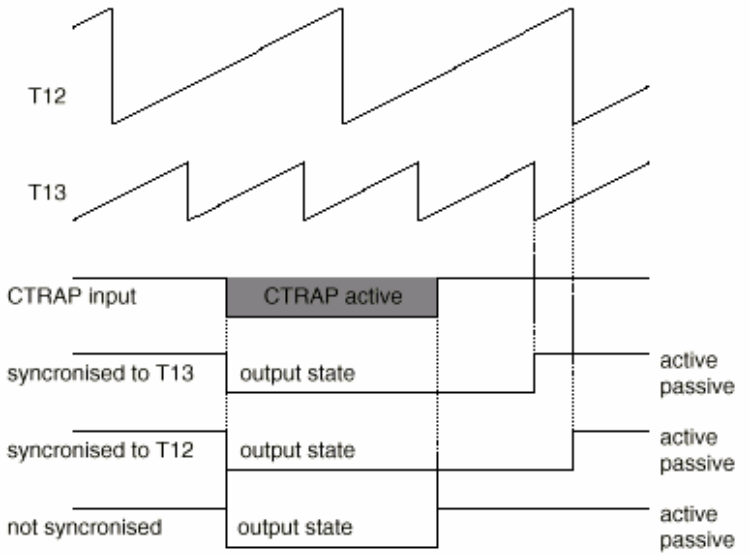
One to Three Phase - Halfbridge Control

EX: CC6_5/6

CC6 – Hardware Trap Generation

Hardware Trap Generation

- TRAP control
  - in TRAP state **all** outputs can be switched to selected passive state
  - several possibilities for leaving TRAP state, synchronization to PWM
  - TRAP state can be triggered by SW or HW



One to Three Phase - Halfbridge Control

EX: CC6_5/6

CC6 – Hysteresis Like Control Mode

Hysteresis Like Control Mode

- CCPOSx Inputs
  - phase can be switched inactive upon comparator signal, e.g. overcurrent
  - simple & ultralowcost hardware control structure possible

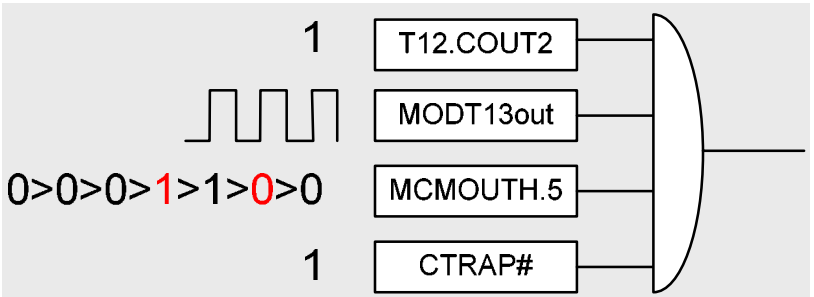
One to Three Phase - Halfbridge Control

EX: CC6_7

CC6 – Multichannel Control Mode

Multichannel Mode

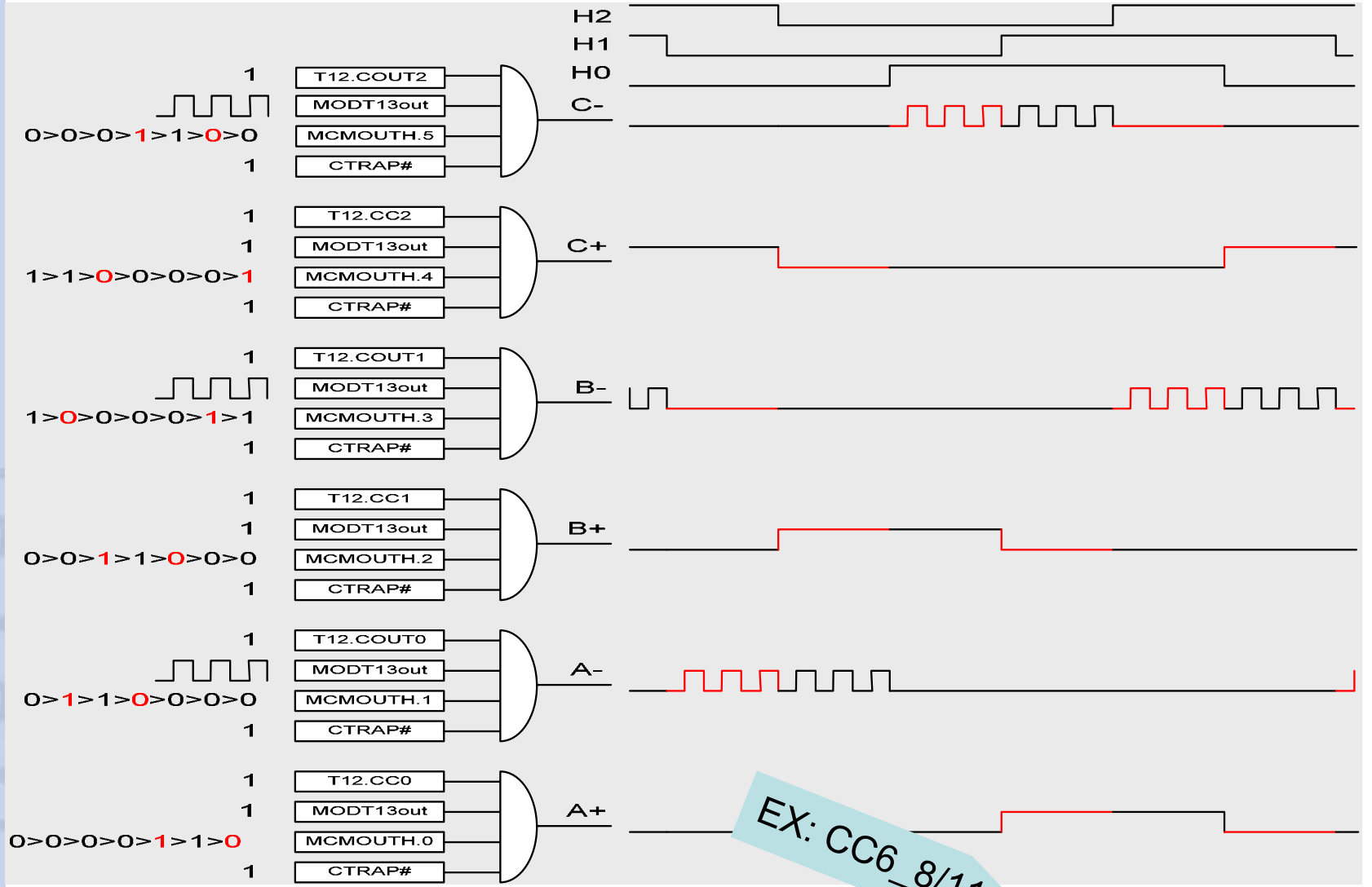
- Modulation Control
  - T12output (3 channel)
  - T13output (1 channel)
  - MCMOUT.x (6 Bits)
  - CTRAP



BLDC Motor Control

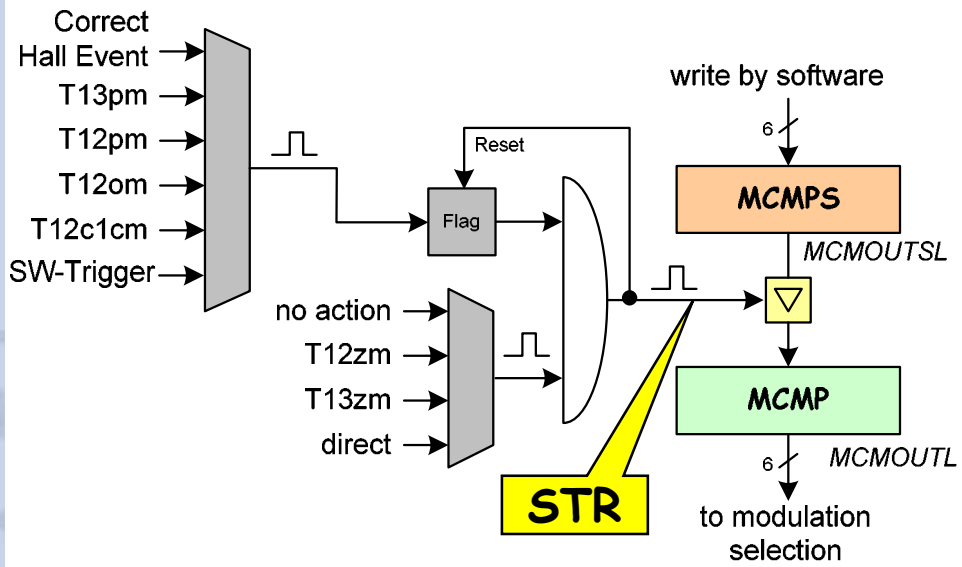
EX: CC6_8

CC6 – MCM
Generate the PWM Pattern for BLDC Motor



CC6 – Multichannel Control Mode

Multi Channel PWM Pattern Switching



- **Software Modulation Control**
  - Switching Selection
  - Switching Synchronization
  - New Bit STR „Shadow Transfer Request“ for qualified interrupt generation

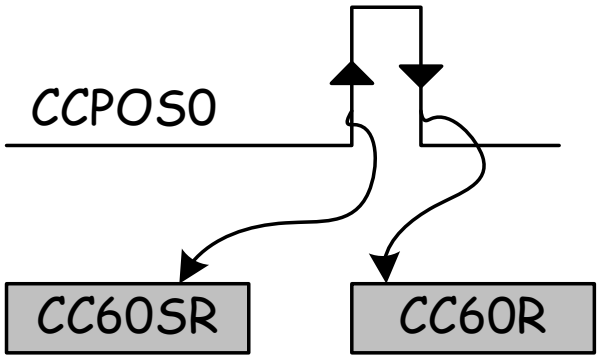
BLDC Motor Control / Sophisticated Pulse Pattern

EX: CC6_8

CC6 – Capture Modes

Double Register Capture

- Selective Capture
  - rising / falling edge
  - for period measurement
  - for duty cylce measurement



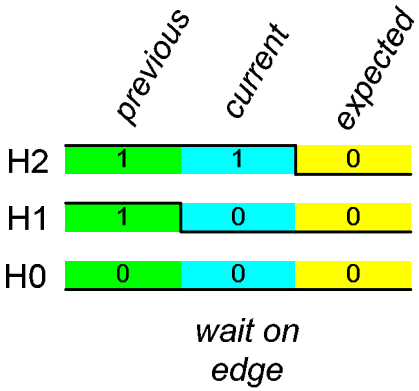
Hall Sensor „Speed“ Measurement

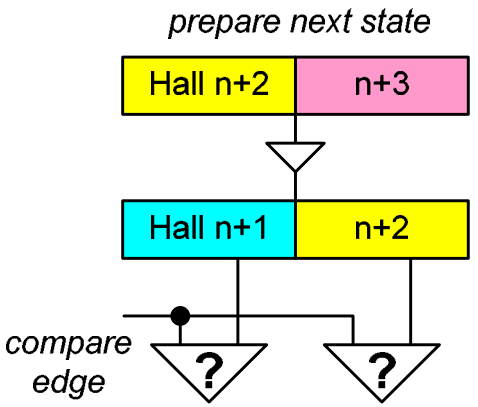
EX: CC6_9

CC6 – Hall Pattern Mode

Hall Pattern Mode

- Hall Pattern Control
  - statemachine without any timer involved
  - flexible input control





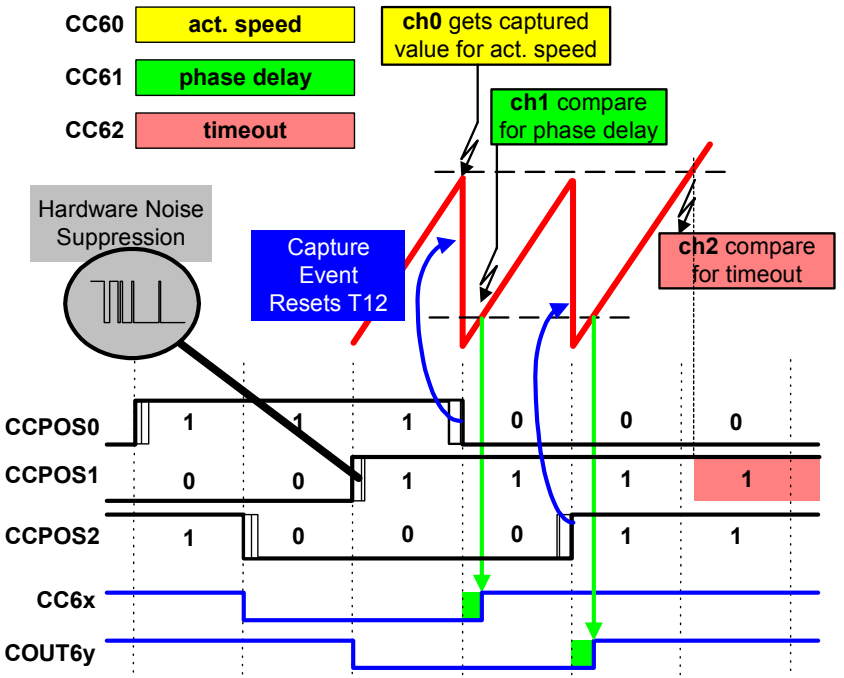
Soft-/Hardware - Statemachine

EX: CC6_10

CC6 – Hall Sensor Mode

Hall Sensor Mode

- BEMF-Detection/Hall Signals
  - HW-noise filter on CCPOSx inputs (BEMF-signals)
  - automatic reset of T12
  - actual speed by capture on ch0
  - phase delay function on ch1
  - time out function on ch2
  - qualified interrupt for next state by STR



BLDC Motor Control

EX: CC6_11

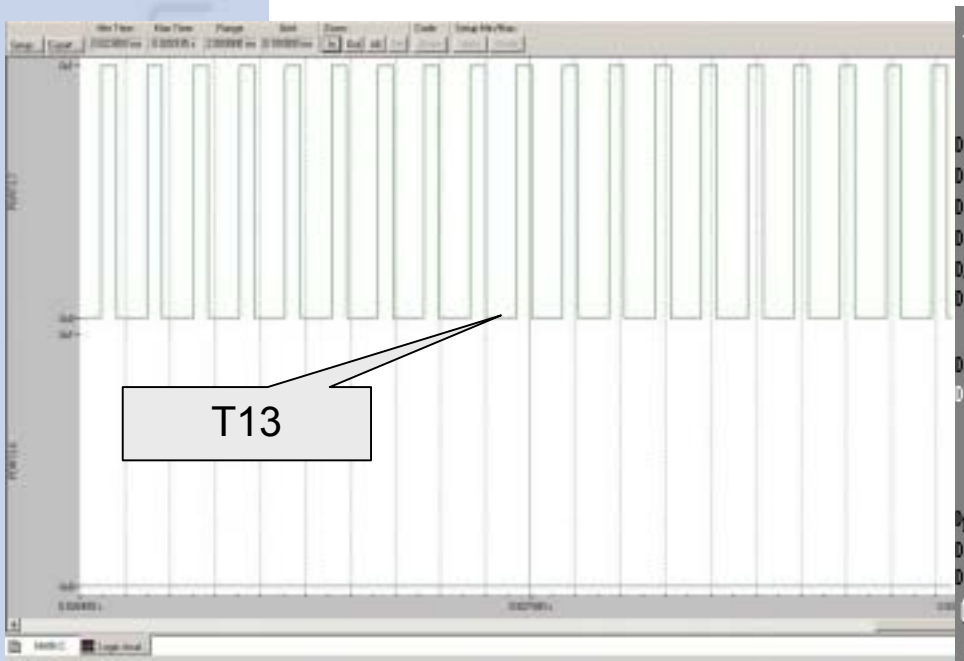

Any Questions?

Table of contents

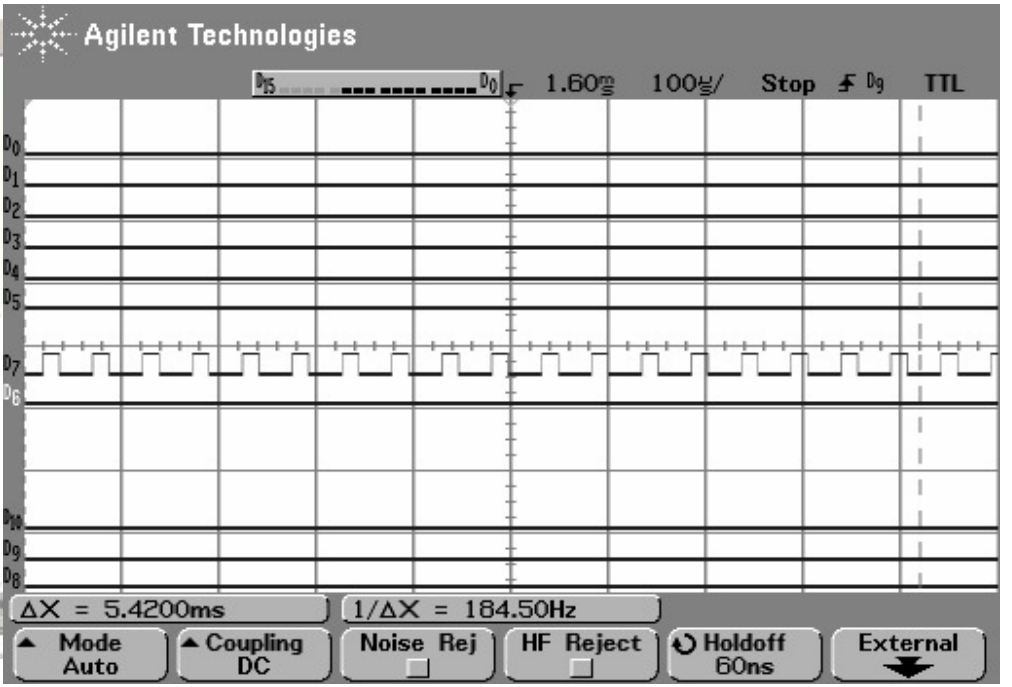
How to use the CC6 – CC6-Example 1

- CC6_1 – Single Channel Constant PWM on T13

- Logic Analyzer in µVision



- Scope Screenshot



How to use the CC6 – CC6-Example 2

- CC6_2 – Single Channel Variable PWM on T13 with CTRAP
  - interrupt routine for CTRAP
  - interrupt routine for T13PM with update of duty cycle
  - main routine forces CTRAP by pulldown on P3.6

- Copy following code
  - main.c

```
// USER CODE BEGIN (MAIN_Main,1)
uword uwBufT13PR;
// USER CODE END

void main(void)
{
 // USER CODE BEGIN (MAIN_Main,2)
 uword i = 0;
 // USER CODE END

 MAIN_vInit();

 // USER CODE BEGIN (MAIN_Main,3)
 SFR_PAGE(_cc1, SST0);       // switch to page 0
 uwBufT13PR = CCU6_T13PR;
 SFR_PAGE(_cc0, RST0);       // restore ccu_page
 // USER CODE END

 while(1)
 {
 // USER CODE BEGIN (MAIN_Main,4)
 // force CTRAP
 for (i=0x1000; i>0; i--);
 SFR_PAGE(_pp1, SST0);     // switch to page 1
 P3_PUDSEL    ^=  0x40;    // force CTRAP by pulldown on P3.6
                           // alternatively use debugger
 SFR_PAGE(_pp0, RST0);     // switch to page 1
 // USER CODE END
 }

} //  End of function main
```

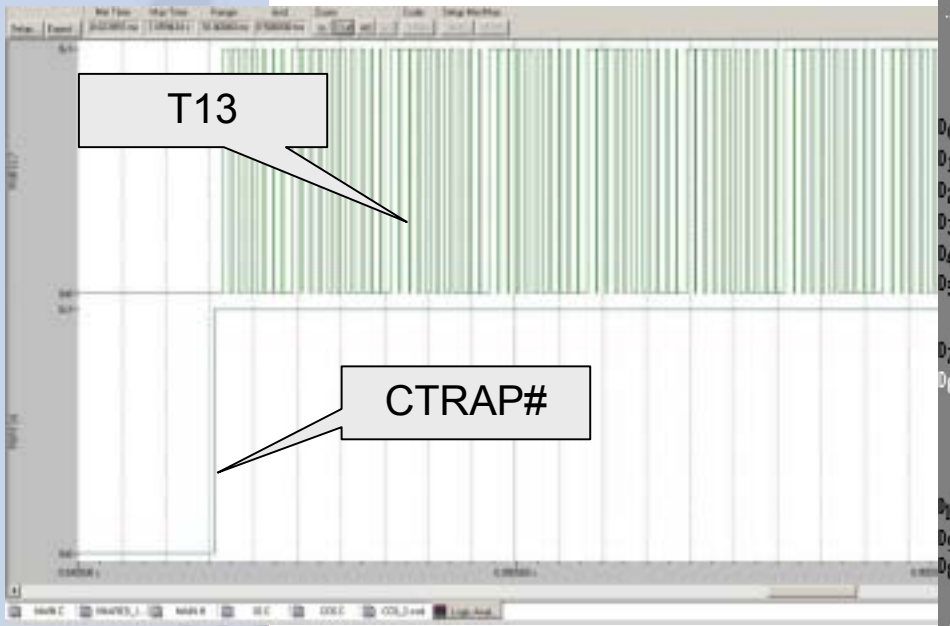
- Copy following code
  - shared_int.c

```
// USER CODE BEGIN (Node0,1)
uword uwDuty3 =0;
extern uword uwBufT13PR;
// USER CODE END

void SHINT_viXINTR10Isr(void) interrupt XINTR10INT
{
•...


if(CCU6_ISH & 0x02)   //if ISH_T13PM
      {
       //timer T13 period match detection

       SFR_PAGE(_cc0, noSST);      // switch to page 0

       CCU6_ISRH = 0x02;  //clear flag ISH_T13PM

       // USER CODE BEGIN (Node0,19)
       // update duty cycle  for Channel 3
          uwDuty3+=0x0080;
          if (uwDuty3>uwBufT13PR) uwDuty3=0;
         CCU6_CC63SR = uwDuty3;
         CC6_vEnableShadowTransfer_CC6_TIMER_13();
         SFR_PAGE(_cc1, RST0);       // restore ccu_page

       // USER CODE END
```

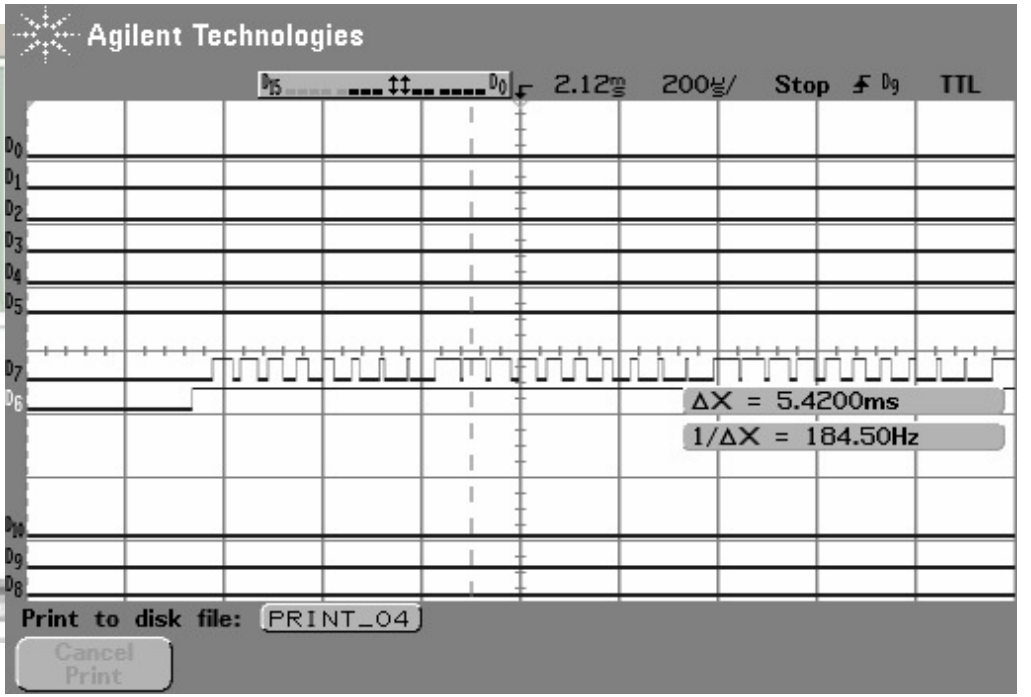
How to use the CC6 – CC6-Example 2

- CC6_2 – Single Channel Variable PWM on T13 with CTRAP

- Logic Analyzer in µVision



- Scope Screenshot

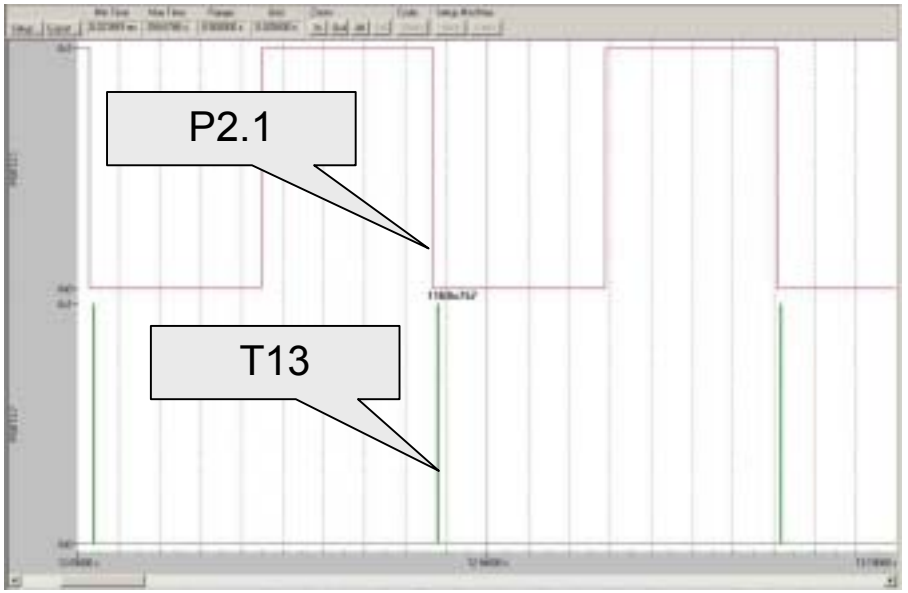


How to use the CC6 – CC6-Example 3

- CC6_3 – Single Shot Mode T13
  - **external run (T13HR) from P2.1 falling edge**
  - **interrupt service routine for T13PM**
  - **main routine forces T13HR by pulldown on P2.1**
- Copy following code
  - main.c

```
 void main(void)
{
  // USER CODE BEGIN (MAIN_Main,2)
   uword uwi = 0;
  // USER CODE END


    MAIN_vInit();

  // USER CODE BEGIN (MAIN_Main,3)

  // USER CODE END

  while(1)
  {

  // USER CODE BEGIN (MAIN_Main,4)
  // force P2.1 edge
  for (uwi=0x1000; uwi>0; uwi--);
  SFR_PAGE(_pp1, SST0);        // switch to page 1
  P2_PUDSEL     ^=  0x02;      // force T13HR on P21
  SFR_PAGE(_pp0, RST0);        // switch to page 1

  // USER CODE END

  }

} //  End of function main
```

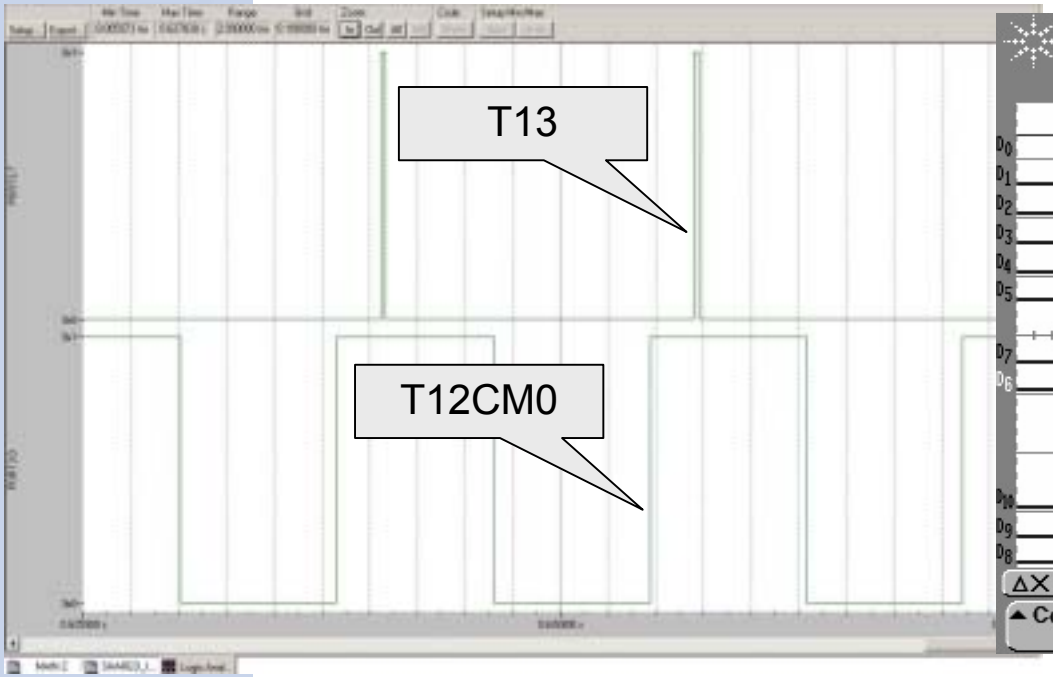
- Logic Analyzer in µVision



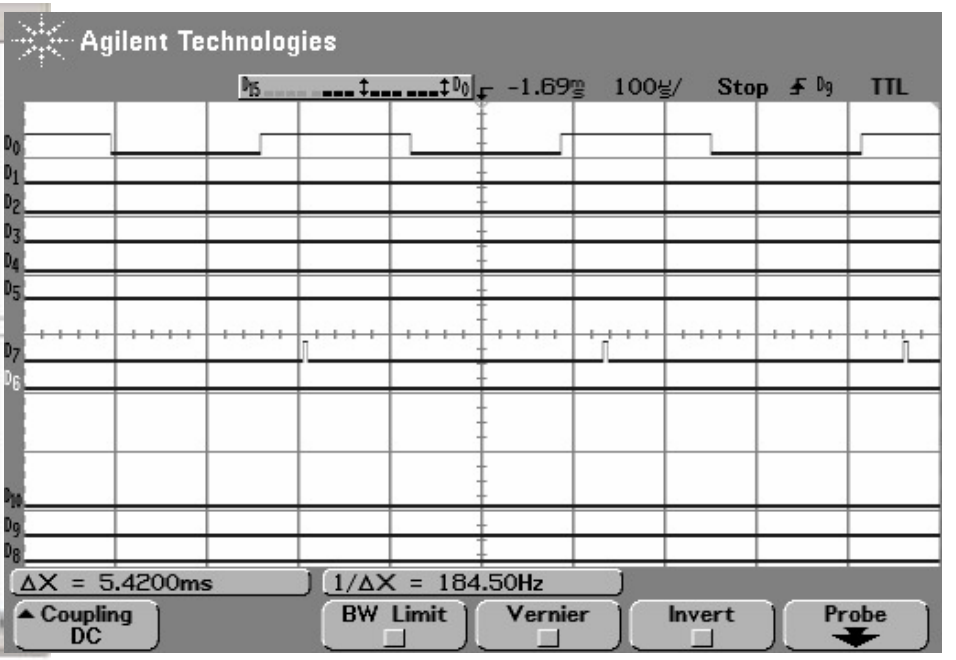
How to use the CC6 – CC6-Example 4

- CC6_4 – Single Shot Mode T13
  - **T12 in center aligned mode, PWM on ch0, 50%**
  - **interrupt service routine for T12PM**
  - **single shot mode T13, synchronize and start with T12CMch0 upcount**
  - **duty cycle for T13 10%, output on P3.7**
  - **interrupt service routine for T13PM**

- Logic Analyzer in µVision



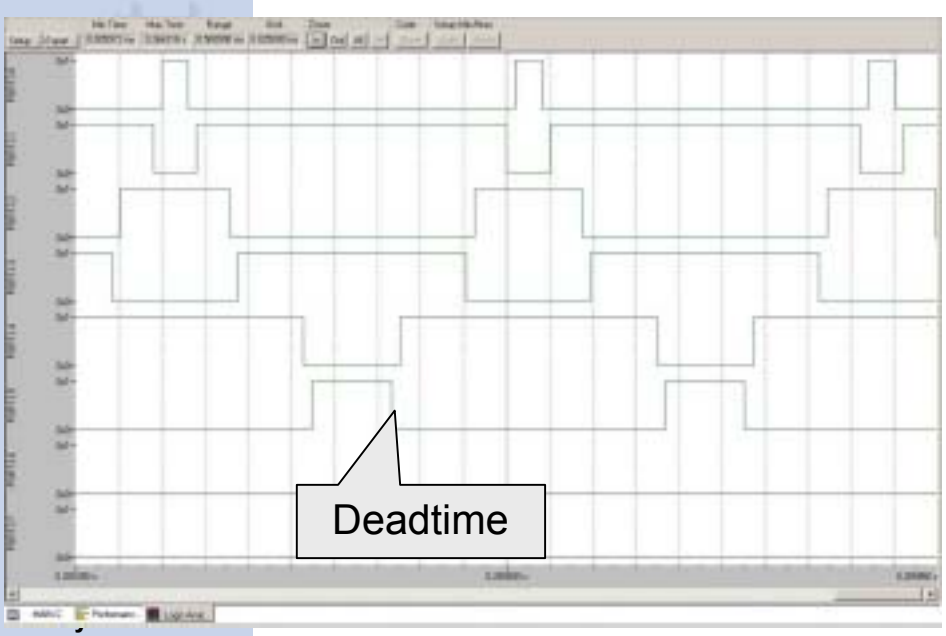
- Scope Screenshot



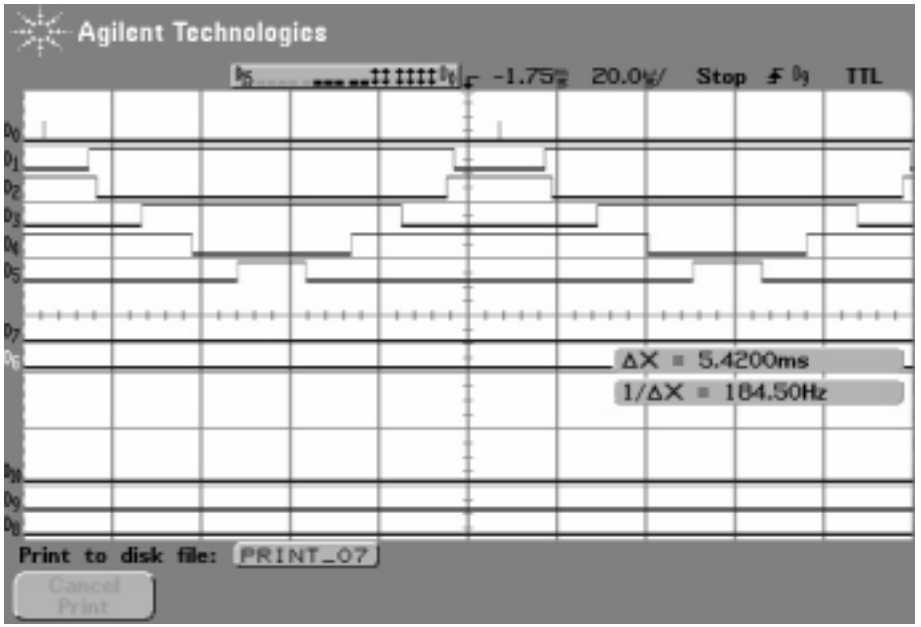
How to use the CC6 – CC6-Example 5

- CC6_5 – 3 channel PWM with deadtime on T12
  - 3ch/6outputs with inverted output level and deadtime
  - for driving stage with highside & lowside active high
  - T12 in center aligned mode
  - interrupt service routine for T12PM

- Logic Analyzer in µVision



- Scope Screenshot

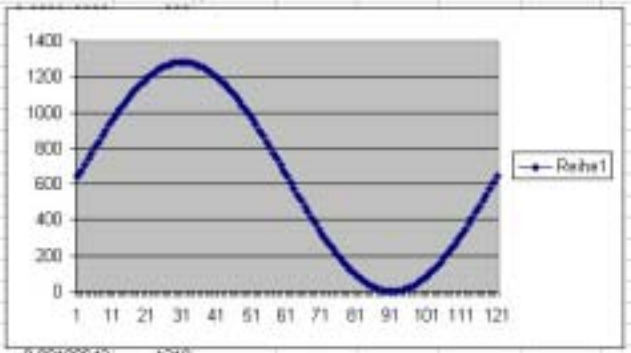


How to use the CC6 – CC6-Example 6

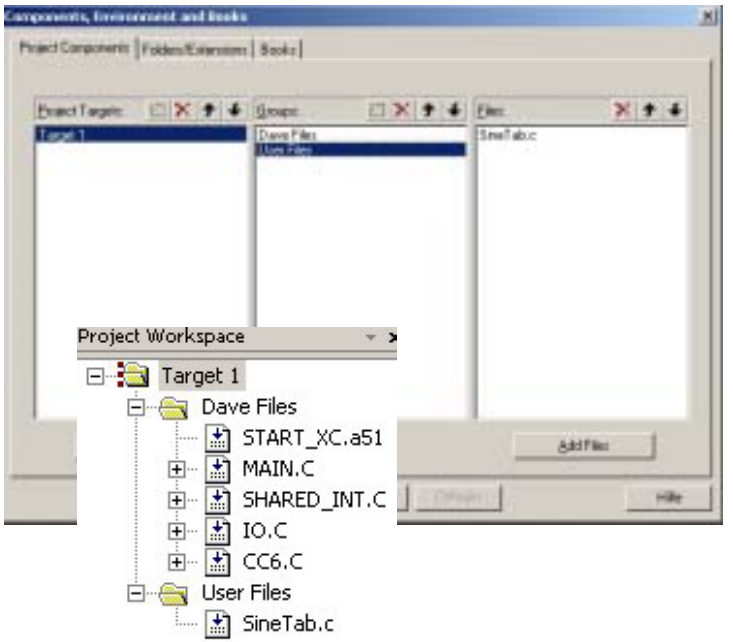
- CC6_6 – 3 channel sinusoidal PWM with deadtime and CTRAP
  - update duty cycles with every T12PM interrupt
  - simple sinusoidal table >> see excel sheet SineTab.xls
  - force CTRAP by pulldown on P3.6
  - leave trapstate synchronized to T12period

1) Generate Sine-Table from Excel

	A	B	C	D
1	XC886CLM — CC6_6-Example Sine Wave			
2	120 values for 360°			
3		hex	dec	
4	Period value	500	1280	
5				
6	rad = angle/180*PI	0.017453		scaling
7	angle	rad		period*val[x]/2 + period/2
8	0	0	0	640 ,
9	3	0.05236	0.052336945	673 ,
10	6	0.10472	0.104528442	707 ,
11	9	0.15708	0.156434433	740 ,
12	12	0.209439	0.207911648	773 ,
13	15	0.261799		
14	18	0.314159		
15	21	0.366519		
16	24	0.418879		
17	27	0.471239		
18	30	0.523599		
19	33	0.575959		
20	36	0.628318		
21	39	0.680678		
22	42	0.733038		
23	45	0.785398		
24	48	0.837758		
25	51	0.890118		
26	54	0.942478		
27	57	0.994837		
28	60	1.047197		
29	63	1.099557	0.89100642	1210 ,
30	66	1.151917	0.91354536	1225 ,



2) Import SineTab.c into project



How to use the CC6 – CC6-Example 6

3) Copy following code

– main.c

```
 void main(void)
{
  // USER CODE BEGIN (MAIN_Main,2)
   uword uwi = 0;
  // USER CODE END

  MAIN_vInit();

  // USER CODE BEGIN (MAIN_Main,3)

  // USER CODE END

  while(1)
  {

  // USER CODE BEGIN (MAIN_Main,4)
  // force CTRAP
  for (uwi=0x1000; uwi>0; uwi--);
  SFR_PAGE(_pp1, SST0);        // switch to page 1
  P3_PUDSEL     ^=  0x40;      // force CTRAP on P3.6
  SFR_PAGE(_pp0, RST0);        // switch to page 1
// USER CODE END

  }

} //  End of function main
```

How to use the CC6 – CC6-Example 6

4) Copy following code
   - shared_int.c

```
//***********************************************
// @Imported Global Variables
//***********************************************
// USER CODE BEGIN (SHARED_INT_General,6)
extern const unsigned int code SineTab[120];
// USER CODE END
```

...

```
// USER CODE BEGIN (Node10,1)
ubyte idxA = 0;
ubyte idxB = 40;
ubyte idxC = 80;
uword i;
// USER CODE END

void SHINT_viXINTR10Isr(void) interrupt XINTR10INT
{
```

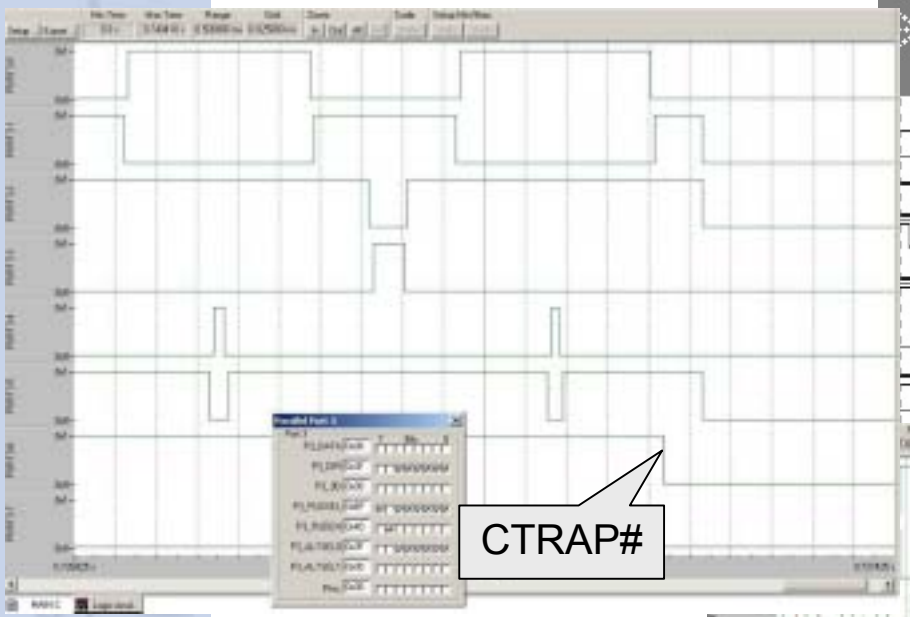
...

...

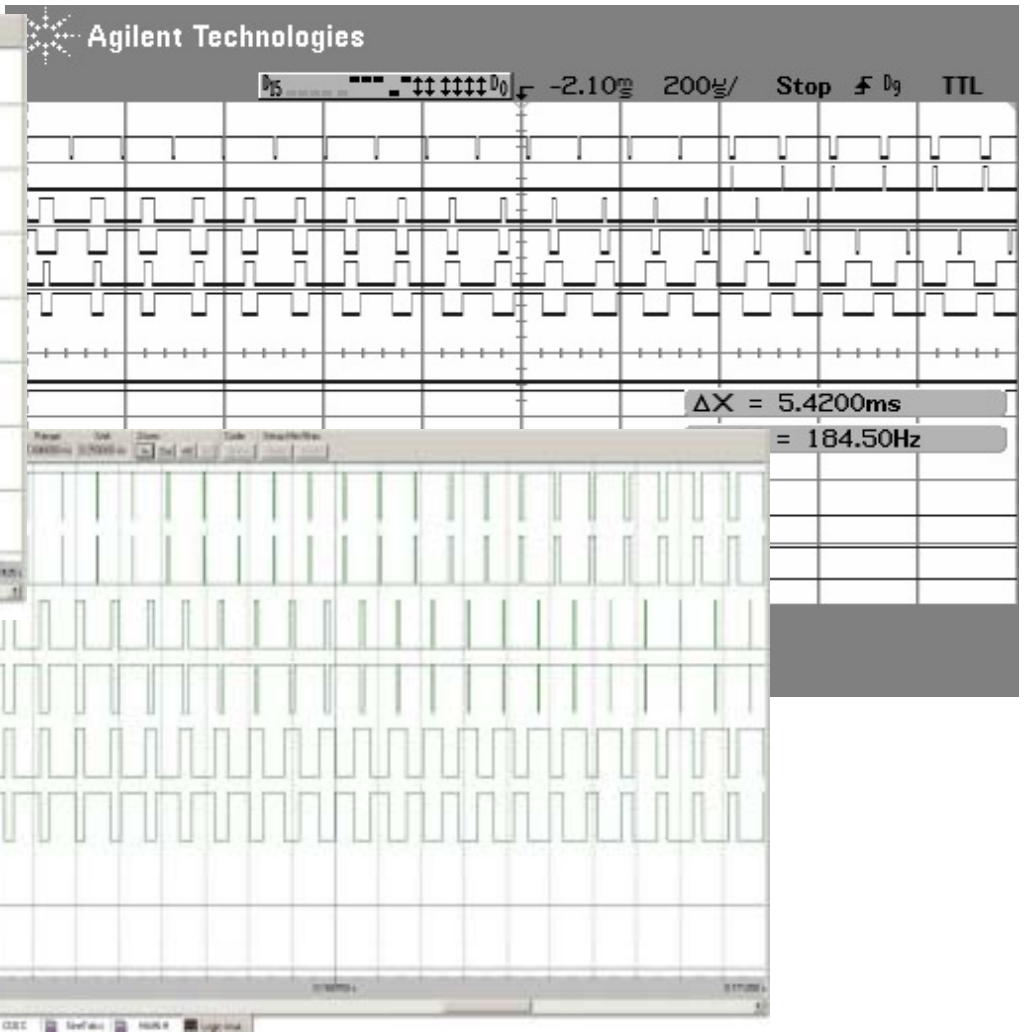
```
if(CCU6_ISL & 0x80)   //if ISL_T12PM

{
  //timer T12 period match detection
  SFR_PAGE(_cc0, noSST);      // switch to page 0
  CCU6_ISRL = 0x80;  //clear flag ISL_T12PM
  // USER CODE BEGIN (Node10,17)

  idxA++;
  if (idxA > 119) idxA=0;
  CCU6_CC60SRLH = SineTab[idxA];

  idxB++;
  if (idxB > 119) idxB=0;
  CCU6_CC61SRLH = SineTab[idxB];

  idxC++;
  if (idxC > 119) idxC=0;
  CCU6_CC62SRLH = SineTab[idxC];

  CC6_vEnableShadowTransfer_CC6_TIMER_12();

 // USER CODE END
```

How to use the CC6 – CC6-Example 6

- Logic Analyzer in µVision



- Scope Screenshot



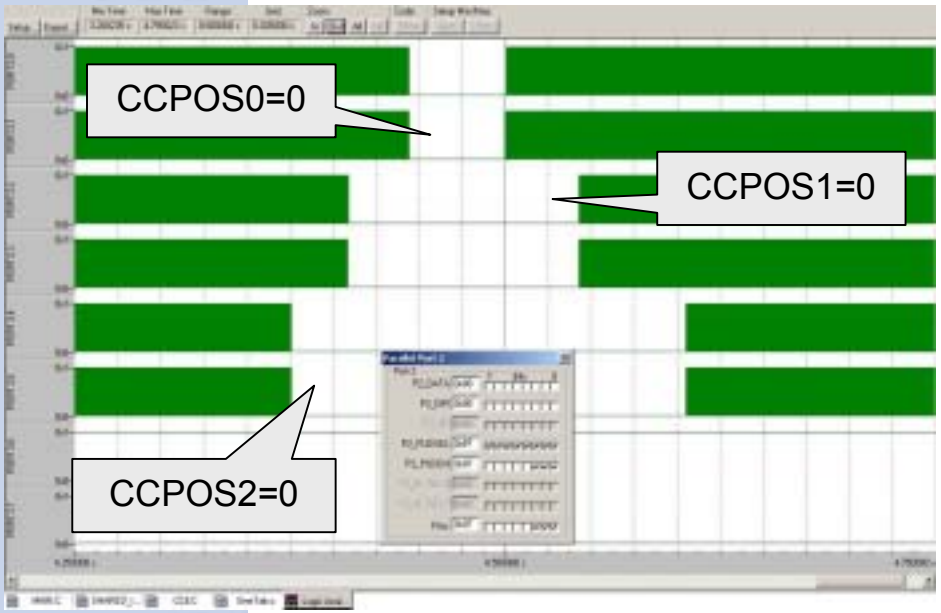
How to use the CC6 – CC6-Example 7

- CC6_7 – Hysteresis like Mode with 3 channel sinusoidal PWM on T12
  - update duty cycles with every T12PM interrupt
  - simple sinusoidal table >> see excel sheet SineTab.xls
  - force Hysteresis Control via P2.0/1/2 pulldevices

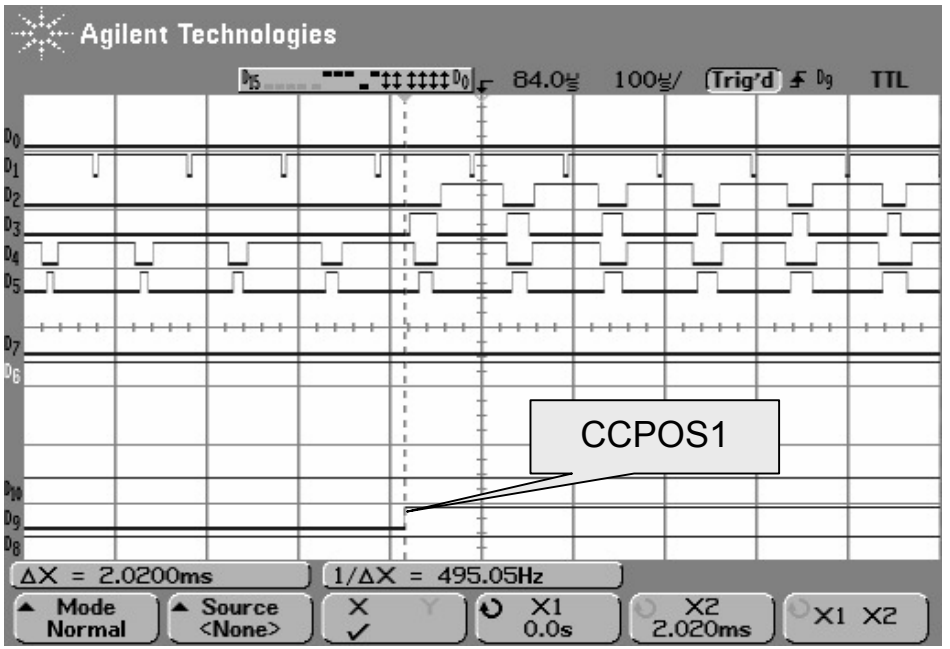
1) Follow step 1- 4 from Example 6
2) Select in Pin Control 1 setting for CCPOSx
3) Select in Channels / Mode Control >> Hysteresis like mode

How to use the CC6 – CC6-Example 7

- Logic Analyzer in µVision

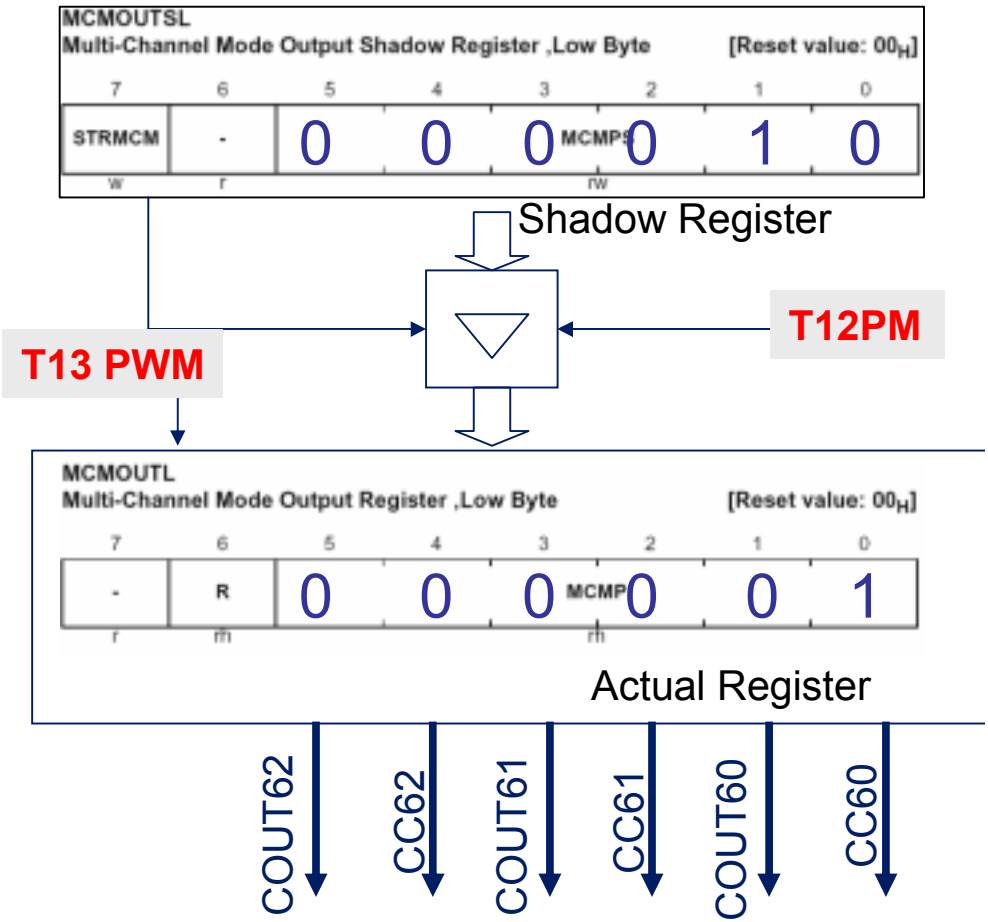


- Scope Screenshot



How to use the CC6 – CC6-Example 8

- CC6_8 – Multichannel mode T12/T13 and MCMOUTx
  - T13 modulation on CC6x/Cout6x (~20kHz) with variable duty cycle
  - T12 periodically switches to next multichannel state with var. period(~0.5-1kHz)
  - switching transfer on T12PM
  - switching synchronization on T13ZM
  - CTRAP enabled for all CC6 pins
  - interrupt for STR and CTRAP
- PWM-Pattern:
  - 0x01 >> 0x02 >> 0x04 >>
  - 0x08 >> 0x10 >> 0x20



How to use the CC6 – CC6-Example 8

- Copy following code
  - shared_int.c

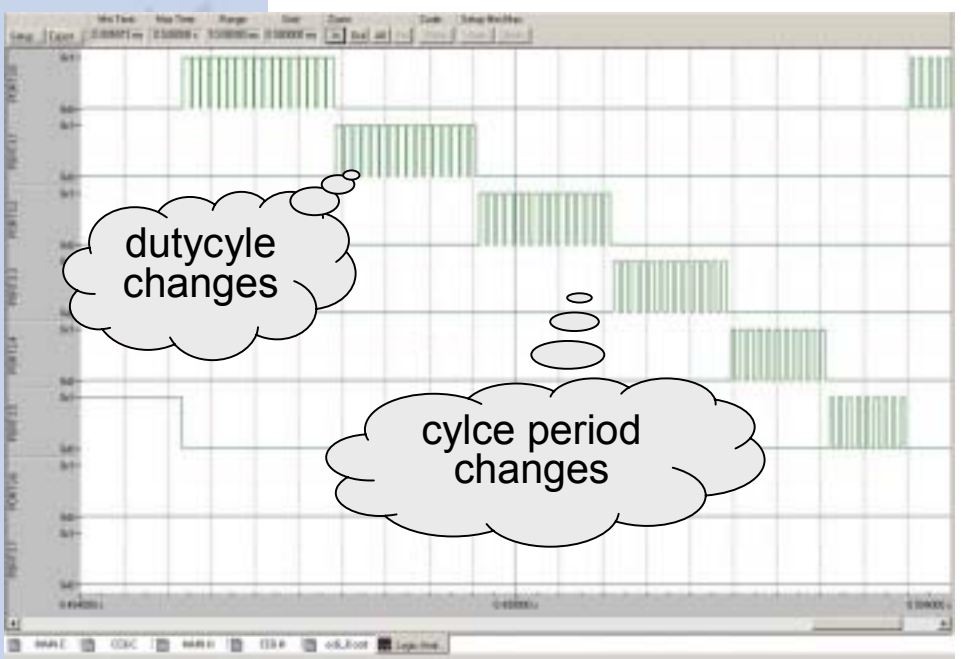
```
// USER CODE BEGIN (Node10,1)
const unsigned char code MCMstate[6] =
{0x01, 0x02, 0x04, 0x08, 0x10, 0x20};  //MCM states
unsigned char idxMCM;
// USER CODE END

void SHINT_viXINTR10Isr(void) interrupt XINTR10INT
```

...

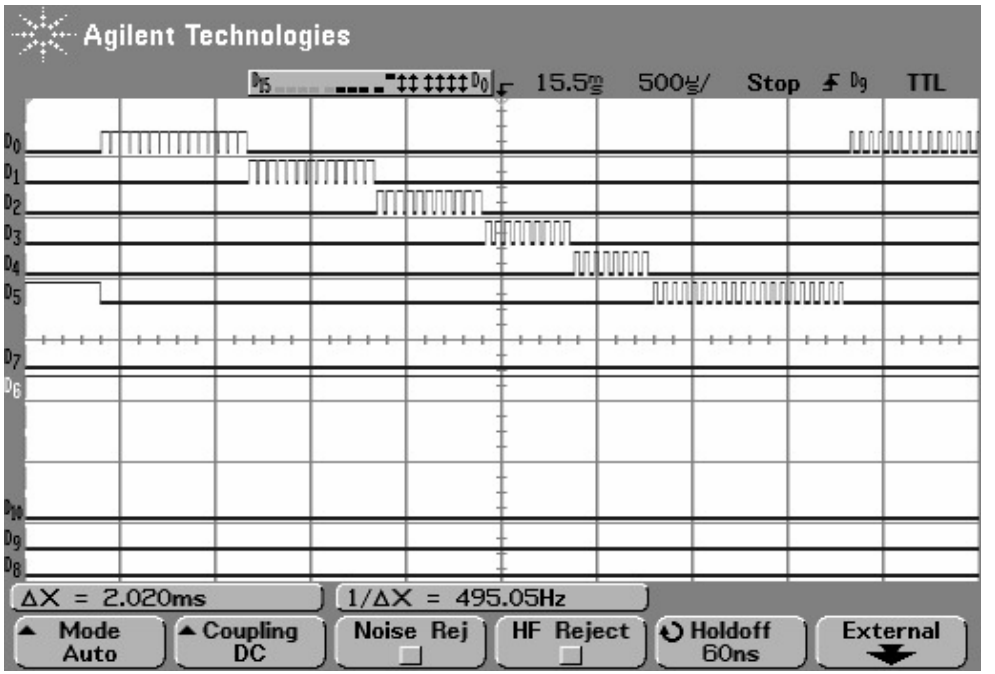
```
SFR_PAGE(_cc3, noSST);        // switch to page 3

 if(CCU6_ISH & 0x80)   //if ISH_SSTR
 {
  // shadow transfer has taken place.
  SFR_PAGE(_cc0, noSST);       // switch to page 0
  CCU6_ISRH = 0x80;  //clear flag ISH_SSTR
  // USER CODE BEGIN (Node10,24)
  // change MCM state
    idxMCM++;
    if (idxMCM>5) idxMCM=0;
    CCU6_MCMOUTSL = MCMstate[idxMCM];
    CCU6_MCMOUTSL |= 0x80;

  // change duty cycle of T13
    CCU6_CC63SRLH+=0x80;
    if (CCU6_CC63SRLH > 0x0500) CCU6_CC63SRLH=0;
                                          //check limit
    CC6_vEnableShadowTransfer_CC6_TIMER_13();

  // change period of T12
    SFR_PAGE(_cc1, SST1);                 // switch to page 1
    CCU6_T12PRLH-=0x800;
    if (CCU6_T12PRLH < 0x3000) CCU6_T12PRLH=0x6000;
                                          // check limit
    SFR_PAGE(_cc1, RST1);                 // restore page
    CC6_vEnableShadowTransfer_CC6_TIMER_12();

// USER CODE END  }
```

How to use the CC6 – CC6-Example 8

- Logic Analyzer in µVision



- Scope Screenshot



How to use the CC6 – CC6-Example 9

- CC6_9 – Double register capture with T12
  - capture on rising edge on pin CC60 to CC60RLH
  - capture on falling edge on pin CC60 to CC60SRLH

- Copy following code
  - main.c

```
void main(void)
{
 // USER CODE BEGIN (MAIN_Main,2)
 unsigned int i;
 // USER CODE END

 MAIN_vInit();

 // USER CODE BEGIN (MAIN_Main,3)

 // USER CODE END

 while(1)
 {
 // USER CODE BEGIN (MAIN_Main,4)
 for (i=0x1800;i>0;i--);
 SFR_PAGE(_pp1, SST0);        // switch port page
 P3_PUDSEL &= ~0x01;          // falling edge
 SFR_PAGE(_pp1, RST0);        // restore the old port page

 for (i=0x0800;i>0;i--);
 SFR_PAGE(_pp1, SST0);        // switch port page
 P3_PUDSEL |= 0x01;           // rising edge
 SFR_PAGE(_pp1, RST0);        // restore the old port page
 // USER CODE END
```

- Copy following code
  - shared_int.c

```
// USER CODE BEGIN (NodeI0,1)
unsigned int uiCapRise, uiCapFall;
// USER CODE END

void SHINT_viXINTR10Isr(void) interrupt XINTR10INT
.....

 if(CCU6_ISL & 0x01)   //if ISL_ICC60R
 {
   //capture, compare match rising edge detection an channel 0
   SFR_PAGE(_cc0, noSST);       // switch to page 0
   CCU6_ISRL = 0x01;            //clear flag ISL_ICC60R
   // USER CODE BEGIN (NodeI0,10)
   // double register capture mode 0100
   // rising edge >> capture to CC60RLH
   SFR_PAGE(_cc1, SST0);        // switch to page 1
   uiCapRise = CCU6_CC60RLH;
   SFR_PAGE(_cc1, RST0);        // restore page
   // USER CODE END
 }
 SFR_PAGE(_cc3, noSST);         // switch to page 3

 if(CCU6_ISL & 0x02)   //if ISL_ICC60F
 {
   //capture, compare match faling edge detection an channel 0
   SFR_PAGE(_cc0, noSST);       // switch to page 0
   CCU6_ISRL = 0x02;  //clear flag IS_ICC60F
   // USER CODE BEGIN (NodeI0,11)
   // double register capture mode 0100
   // falling edge >> capture to CC60SRLH
   uiCapFall = CCU6_CC60SRLH;
   // USER CODE END
```

How to use the CC6 – CC6-Example 10

- CC6_10 – Hallpattern and Hallogic example
  - Hall-Logic without timer involved
  - Delay Bypass
  - manual porttoggling in CHE ISR
  - CHE / WHE interrupt generation
- Copy following code
  - main.c

```
 const unsigned char code HallPatt[6] =
{0x3E, 0x34, 0x20, 0x01, 0x0B, 0x1F}; //Hall states
// USER CODE END

void main(void)
{
  // USER CODE BEGIN (MAIN_Main,2)
  unsigned char i;
  unsigned int ida;
  // USER CODE END

  MAIN_vInit();

  // USER CODE BEGIN (MAIN_Main,3)
#define MCM_transfer 0x80

i = (P2_DATA & 0x7);                    //111>110>100>000>001>011
                        case   7: i=0; break;
                        case   6: i=1; break;
                        case   4: i=2; break;
                        case   0: i=3; break;
                        case   1: i=4; break;
                        case   3: i=5; break;
                        default : i=0; break;
                        }

                        CCU6_MCMOUTSH = HallPatt[i] | MCM_transfer;
                        i++;
                        CCU6_MCMOUTSH = HallPatt[i];
  // USER CODE END
```

- Copy following code
  - main.c

```
// DBYP/HSYNC
SFR_PAGE(_cc2, noSST);        // switch to page 2
CCU6_T12MSELH  =  0x80;       // DBYP=1, any edge at CCPOSx
SFR_PAGE(_cc0, noSST);        // switch to page 0
// USER CODE END
while(1)
{
// USER CODE BEGIN (MAIN_Main,4)

for (ida=0x500; ida>0; ida--);
SFR_PAGE(_pp1, SST0);       // switch to port page 1
P2_PUDSEL = 0x07;                                       //111
SFR_PAGE(_pp1, RST0);       // restore page
for (ida=0x600; ida>0; ida--);
SFR_PAGE(_pp1, SST0);       // switch to port page 1
P2_PUDSEL = 0x06;                                       //110
SFR_PAGE(_pp1, RST0);       // restore page
for (ida=0x700; ida>0; ida--);
SFR_PAGE(_pp1, SST0);       // switch to port page
P2_PUDSEL = 0x04;                                       //100
SFR_PAGE(_pp1, RST0);       // restore page
for (ida=0x800; ida>0; ida--);
SFR_PAGE(_pp1, SST0);       // switch to port page 1
P2_PUDSEL = 0x00;                                       //000
SFR_PAGE(_pp1, RST0);       // restore page
for (ida=0x900; ida>0; ida--);
SFR_PAGE(_pp1, SST0);       // switch to port page 1
P2_PUDSEL = 0x01;                                       //001
SFR_PAGE(_pp1, RST0);       // restore page
for (ida=0xA00; ida>0; ida--);
SFR_PAGE(_pp1, SST0);       // switch to port page 1
P2_PUDSEL = 0x03;                                       //011
SFR_PAGE(_pp1, RST0);       // restore page

// USER CODE END
```

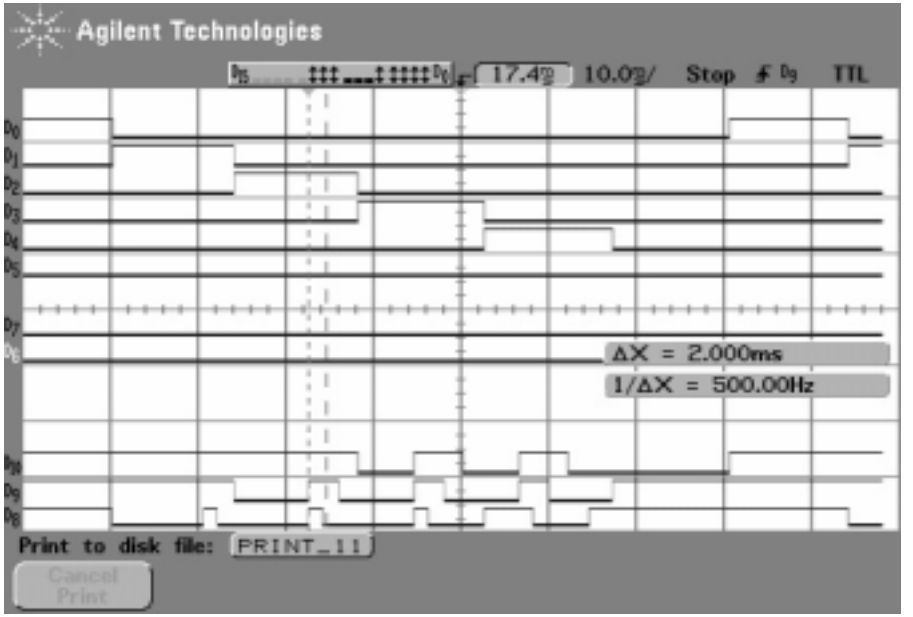
How to use the CC6 – CC6-Example 10

- Copy following code
  - shared_int.c

Simulator currently does not support this mode

```
// USER CODE BEGIN (Node10,1)
extern code unsigned char HallPatt[];
unsigned char i;
// USER CODE END

void SHINT_viXINTR10Isr(void) interrupt XINTR10INT
```

```
  if(CCU6_ISH & 0x10)   //if ISH_CHE
 {
   //correct hall event detection

   SFR_PAGE(_cc0, noSST);      // switch to page 0
   SFR_PAGE(_pp0, SST1);       // switch to port page 1
   CCU6_ISRH = 0x10;  //clear flag ISH_CHE

   // USER CODE BEGIN (Node10,23)


i = CCU6_MCMOUTSH & 0x07;          //get next hall state
                        switch (i) {           //111>110>100>000>001>011
                        case   7: i=0; P3_DATA=0x00; break;
                        case   6: i=1; P3_DATA=0x01; break;
                        case   4: i=2; P3_DATA=0x02; break;
                        case   0: i=3; P3_DATA=0x04; break;
                        case   1: i=4; P3_DATA=0x08; break;
                        case   3: i=5; P3_DATA=0x10; break;
                        default : i=0; P3_DATA=0xff; break;
                        }
CCU6_MCMOUTSH = HallPatt[i]; //the shadow transfer will be automatic

   // USER CODE END
```



How to use the CC6 – CC6-Example 11

- CC6_11 – Hallsensor Mode
  - complete statemachine for hallsensor mode
  - emultion of hallpattern using pullups on P2 in mainloop or uVision-simulator
  - T13modulation
  - capture on T12Ch0, (phase shift on T12ch1), timeout on T12ch2
  - Hallpattern on P2.0/1/2
  - PWMpattern on P3.0-6
  - interrupts for CTRAP, T12CH2, WHE, CHE, (STR)

- Emulation or Simulation of Hallpattern

How to use the CC6 – CC6-Example 11

- Copy following code
  - main.c

```
//***************************************************************************
// @Global Variables
//***************************************************************************
// USER CODE BEGIN (MAIN_General,7)
// MCMOUTSH
// STRHP  --- CURHS2 / 1 / 0 EXPHS2 / 1 / 0
// 111>110>100>000>001>011
// 11 1110        3E              111>110
// 11 0100        34              110>100
// 10 0000        20              100>000
// 00 0001        01              000>001
// 00 1011        0B              001>011
// 01 1111        1F              011>111
const unsigned char code HallPatt[6] =
{0x3E, 0x34, 0x20, 0x01, 0x0B, 0x1F}; //Hall states
const unsigned char code OutputPatt[6] =
{0x01, 0x02, 0x04, 0x08, 0x10, 0x20}; //Output states
// USER CODE END
```

```
 // USER CODE BEGIN (MAIN_Main,1)
extern const unsigned char code Hallstate[6]; //Hall states
extern const unsigned char code MCMstate[6];  //MCM states
unsigned char i;
unsigned int ida;
// USER CODE END

void main(void)
```

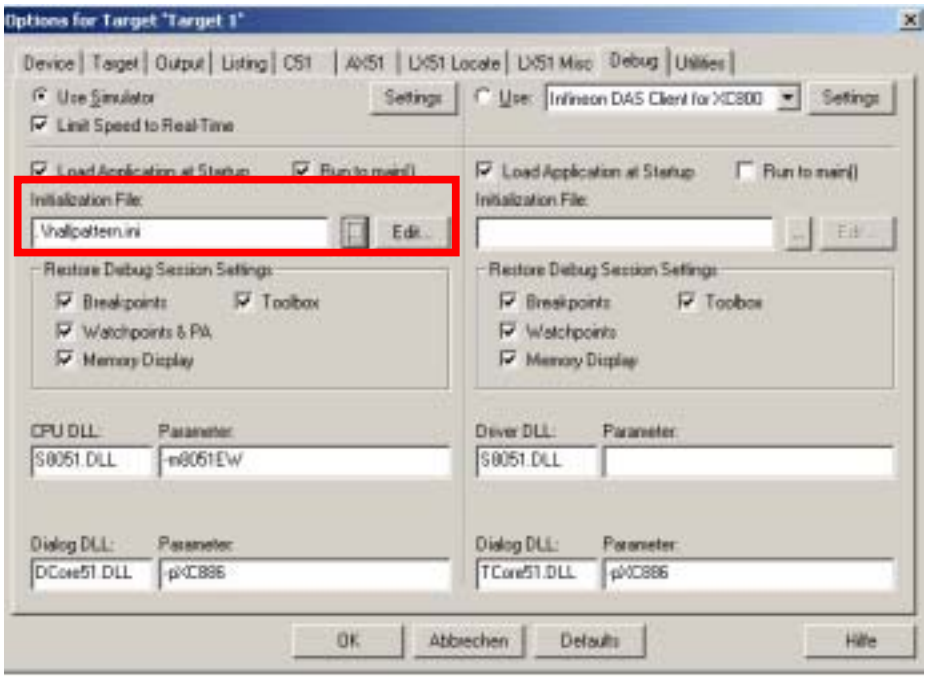
```
void main(void)
{
  // USER CODE BEGIN (MAIN_Main,2)
  // USER CODE END

  MAIN_vInit();

// USER CODE BEGIN (MAIN_Mai
#define MCM_transfer 0x80 = (P2_DATA & 0x7);
switch (i) {                          //111>110>100>000>001>011
  case   7: i=0; break;
  case   6: i=1; break;
  case   4: i=2; break;
  case   0: i=3; break;
  case   1: i=4; break;
  case   3: i=5; break;
  default : i=0; break;
}
CCU6_MCMOUTSH = HallPatt[i] | MCM_transfer;
CCU6_MCMOUTSL = OutputPatt[i] | MCM_transfer;
i++;
CCU6_MCMOUTSH = HallPatt[i];

CCU6_MCMOUTSL = OutputPatt[i];
// USER CODE END
```

How to use the CC6 – CC6-Example 11

- Copy following code
  - main.c (hall pattern emulation)

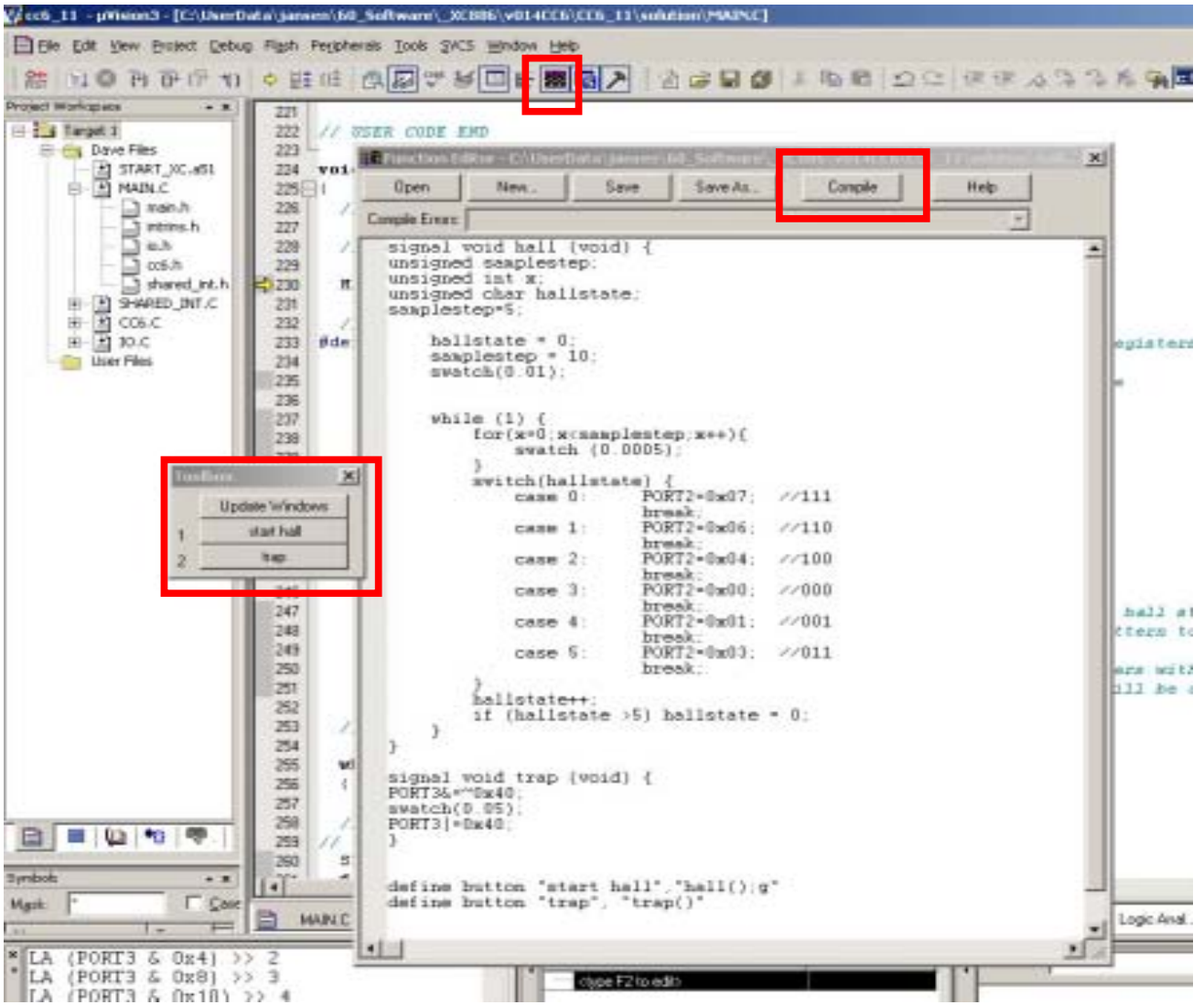
```
while(1)
 {

 // USER CODE BEGIN (MAIN_Main,4)
//  /* hall pattern emulator
  SFR_PAGE(_pp1, SST1);       // switch to port page 1
  for (ida=0x500; ida>0; ida--);
  P2_PUDSEL = 0x07;                                  //111
  for (ida=0x600; ida>0; ida--);
  P2_PUDSEL = 0x06;                                  //110
  for (ida=0x700; ida>0; ida--);
  P2_PUDSEL = 0x04;                                  //100
  for (ida=0x800; ida>0; ida--);
  P2_PUDSEL = 0x00;                                  //000
  for (ida=0x900; ida>0; ida--);
  P2_PUDSEL = 0x01;                                  //001
  for (ida=0xA00; ida>0; ida--);
  P2_PUDSEL = 0x03;                                  //011
  SFR_PAGE(_pp1, RST1);       // restore page
//  */
  // USER CODE END

 }

} //  End of function main
```

- Hall pattern simulation
  - hallpattern.ini



How to use the CC6 – CC6-Example 11

- Hall pattern simulation



How to use the CC6 – CC6-Example 11

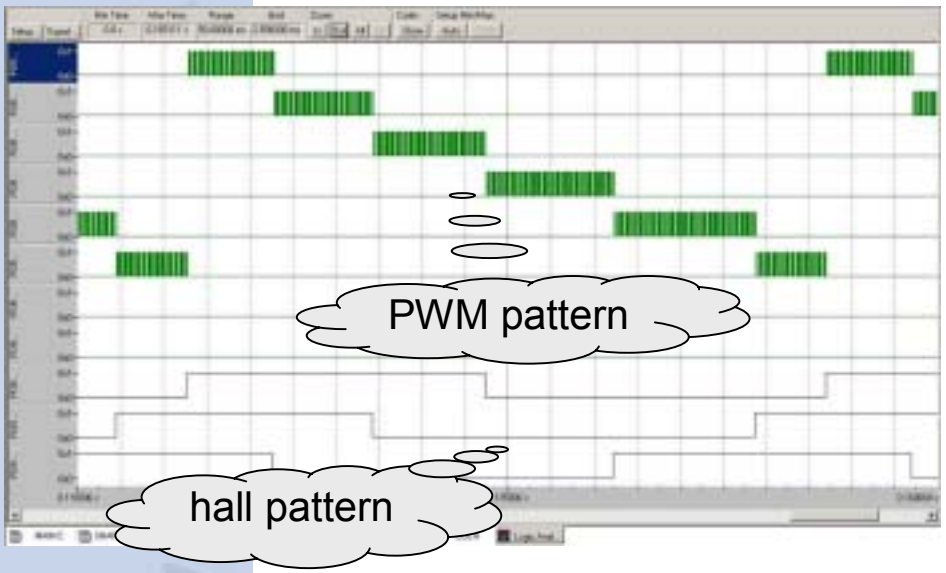
- Copy following code
  - shared_int.c

```
extern code unsigned char HallPatt[];
extern code unsigned char OutputPatt[];
unsigned char i;

 // USER CODE END
 SFR_PAGE(_su3, SST0);        // switch to page 3

if (IRCON4 & 0x10)  //  if CCU6SR3
```

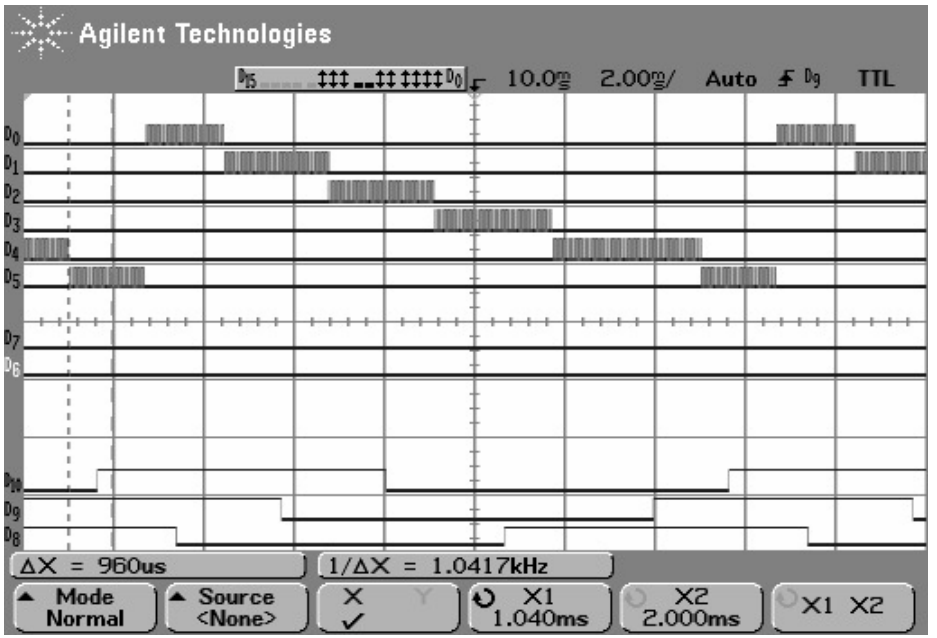
```
if (IRCON4 & 0x10)  //  if CCU6SR3
 {
    IRCON4 &= ~(ubyte)0x10;

 // USER CODE BEGIN (Nodel3,3)
 // USER CODE END

 SFR_PAGE(_cc3, noSST);        // switch to page 3

 if(CCU6_ISL & 0x04)   //if ISL_ICC61R
 {
  //capture, compare match rising edge detection an channel 1
  SFR_PAGE(_cc0, noSST);       // switch to page 0
  CCU6_ISRL = 0x04;  //clear flag ISL_ICC61R

  // USER CODE BEGIN (Nodel3,12)

  i = CCU6_MCMOUTSH & 0x07;   //get next hall state
                switch (i) {            //111>110>100>000>001>011
                case   7: i=0; break;
                case   6: i=1; break;
                case   4: i=2; break;
                case   0: i=3; break;
                case   1: i=4; break;
                case   3: i=5; break;
                default : i=0; break;
                }
                CCU6_MCMOUTSL = OutputPatt[i];
                   //program shadow registers with next state values
                CCU6_MCMOUTSH = HallPatt[i];
                //the shadow transfer will be automatic

  // USER CODE END
```

How to use the CC6 – CC6-Example 11

- Logic Analyzer in µVision



- Scope Screenshot




Infineon
technologies
Any Questions?
Never stop thinking
www.infineon.com