

МИКРОПРОЦЕССОРНА ТЕХНИКА

ЛЕКЦИЯ #13

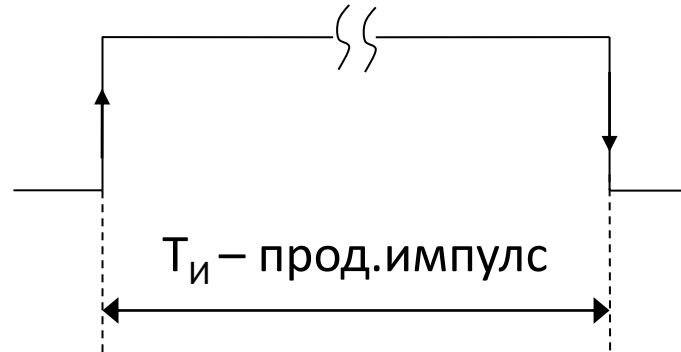
Таймерни функции – приложение (IC функции)

Определяне продължителността $T_{и}$ на единичен импулс

- дефиниране на битове **PR1, PR0** от регистър **TMSK2** така, че **overflow** периодът на таймера > очакваната продълж. на $T_{и}$ (напр. при PR1=1, PR0=0 и XTAL=8MHz --> $\approx 262\text{ms}$, $t_{1\text{отчет}} = 4\mu\text{s}$);
- инициализация на регистър **TCTRL2** (битове EDGxB, EDGxA): **EDGxB=1, EDGxA=1** /и по двата фронта/;
- инициализация на регистри TMSK1[3:0], TFLG1[3:0]:
 - ICxF \rightarrow 1 - флаг за настъпило събитие на ICx. Вдига се в “1” в случай на поява на който и да е фронт (преден или заден);
 - при ICxI=1 – заявка за прекъсване (обслужва се съответната сервизна процедура);
 - при ICxI=0 – таймерна IC функция без прекъсване (програмата продължава със следващия оператор).

Таймерни функции – приложение (IC функции)

Определяне продължителността $T_{И}$ на единичен импулс
(графично представяне)



ICxF се вдига в “1”
 $TCNT(t1) \rightarrow TICx$

ICxF се вдига в “1”
 $TCNT(t2) \rightarrow TICx$

$$T_{И} = [TCNT(t2) - TCNT(t1)] \times t_{1отчет}$$

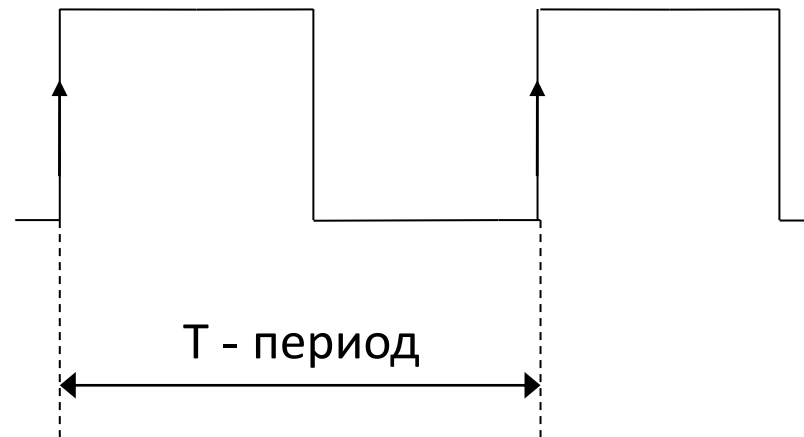
Таймерни функции – приложение (IC функции)

Определяне периода T на импулсна поредица

- дефиниране на битове **PR1, PR0** от регистър **TMSK2** така, че **overflow** периодът на таймера > очакваният период T;
- инициализация на регистър **TCTRL2** (битове **EDGxB, EDGxA**): **EDGxB=0, EDGxA=1** /по преден фронт/;
- инициализация на регистри TMSK1[3:0], TFLG1[3:0]:
ICxF → 1 - флаг за събитие на ICx. Вдига се в случая при поява на преден фронт.
- при ICxI=1 - заявка за прекъсване (обслужва се съответната сервизна процедура);
- при ICxI=0 - таймерна IC функция без прекъсване (програмата продължава със следващия оператор).

Таймерни функции – приложение (IC функции)

Определяне периода T на периодична импулсна поредица
(графично представяне)



ICxF се вдига в "1"
 $TCNT(t1) \rightarrow TICx$

ICxF се вдига в "1"
 $TCNT(t2) \rightarrow TICx$

$$T = [TCNT(t2) - TCNT(t1)] \times t_{1\text{отчет}}$$

Таймерни функции – приложение (ОС функции)

Генериране на единичен импулс с продължителност T_{OUT}

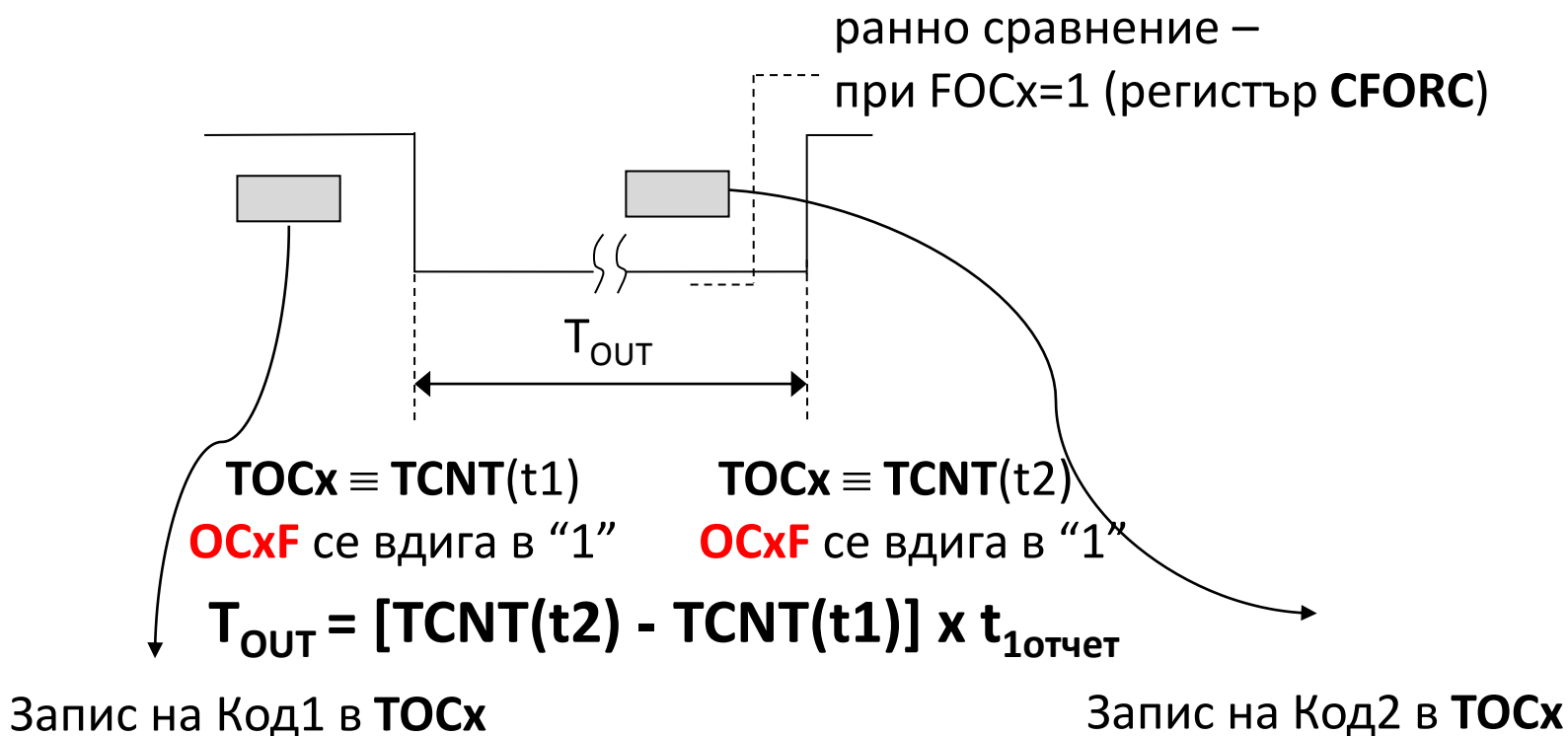
- дефиниране на битове **PR1, PR0** от регистър **TMSK2** така, че **overflow периода на таймера > желан изходен импулс T_{OUT}** ;
- инициализация на регистър **TCTRL1** (битове **OMx, OLx**). Напр.: **OMx=0, OLx=1** (изходната линия сменя състоянието си). Тъй като при Reset регистри **TOx** \rightarrow **\$FFFF** – отрицателен изходен импулс;
- инициализация на регистри **TMSK1[7:3], TFLG1[7:3]**:
 - **OSxF \rightarrow 1** - флаг за успешно сравнение в изход **OSx**.
 - при **OSxI=1** - заявка за прекъсване при успешно сравнение (обслужва се съответната сервизна процедура);
 - при **OSxI=0** - таймерна ОС функция без прекъсване (програмата продължава с изпълнение на следващия оператор).

Регистър **TMSK2** - бит **TOI=1** (разрешава прекъсване при препълване при вдигане на **TOF=1** от регистър **TFLG2**).

Таймерни функции – приложение (ОС функции)

Генериране на единичен импулс с продължителност T_{OUT} (графично представяне)

Допълнителна функционалност: инициализация на регистър **CFORC** – битове FOC[7:3]: при $FOC_x = "1"$ – за ранно сравнение: напр. реакция в съответния изход *при възникване на авария*.



Подсистема за прекъсване в реално време

Real-Time Interrupt (RTI)

- ❑ Служи за генериране на хардуерно прекъсване през фиксиран интервал от време;
- ❑ Определяне на **периода на RTI** прекъсването – от битове **RTR[1:0]** в регистъра **PACTL** (пулс-акумулатор контролен регистър);
- ❑ 4 възможни периода (интервала), определени от двоичните комбинации на битове RTR[1:0];
- ❑ RTI подсистемата се разрешава от бит **RTII="1"** от регистъра **TMSK2**.

Подсистема за прекъсване в реално време (Real-Time Interrupt, RTI)

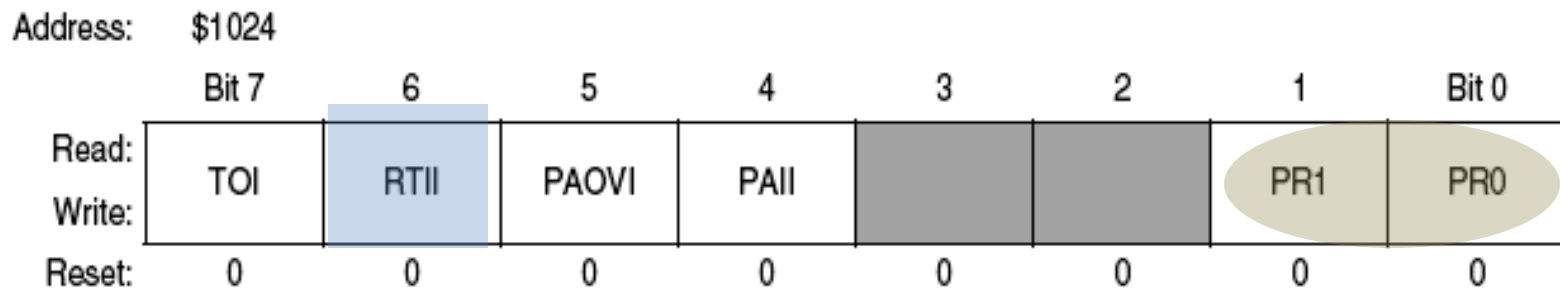
Интервали на генериране на RTI прекъсвания

| RTR[1:0] | E = 3 MHz | E = 2 MHz | E = 1 MHz | E = X MHz |
|----------|-----------|-----------|-----------|--------------|
| 0 0 | 2.731 ms | 4.096 ms | 8.192 ms | $(E/2^{13})$ |
| 0 1 | 5.461 ms | 8.192 ms | 16.384 ms | $(E/2^{14})$ |
| 1 0 | 10.923 ms | 16.384 ms | 32.768 ms | $(E/2^{15})$ |
| 1 1 | 21.845 ms | 32.768 ms | 65.536 ms | $(E/2^{16})$ |

- Тактов източник за RTI функцията – системната тактова E честота, разделена на 2^{13} ;
- Не може да бъде спряна или прекъсната (освен при Reset);
- Фиксирано време между две последователни RTI сработвания - НЕ ЗАВИСИ от софтуерни закъснения свързани с нулиране на флагове (RTIF в случая);
- При изтичане на RTI периода битът RTIF → 1 от регистъра TFLG2 (при RTII="1" от TMSK2) се генерира прекъсване.
/след Reset, RTIF се вдига след 1 пълен период/.

Подсистема за прекъсване в реално време (Real-Time Interrupt, RTI)

- **TMSK2** – Таймерен масков регистър за прекъсване 2 разрешава/забранява прекъсванията в реално време.
- TOI - бит за разрешаване на прекъсване при препълване (overflow) на таймера:
 - 0 - прекъсвания от таймера забранени;
 - 1 - заявка за прекъсвания (при TOF="1").
- RTII - бит за разрешаване на RTI прекъсванията:
 - 0 - прекъсвания от RTI подсист. забранени;
 - 1 - заявка за прекъсвания (при RTIF="1").



Подсистема за прекъсване в реално време (Real-Time Interrupt, RTI)

- **TMSK2** – Таймерен масков регистър за прекъсване 2 (продължение)
- PAOVI - бит за разрешаване на прекъсвания *при препълване на ПА*;
- PAII - бит за разрешаване на прекъсвания *при входен импулс към входа на ПА (PAI)*;
- битове [3:2] - не се използват (винаги се четат 0 от тях);
- битове PR[1:0] – Prescaler битове за таймера (основна верига).

Подсистема за прекъсване в реално време (Real-Time Interrupt, RTI)

□ **TFLG2** – Таймерен флагов регистър за прекъсване 2

Указва настъпване на събития в таймерната система. Свързани позиционно по двойки с масковите битов TMSK2 [7:4].

- TOF - флаг за прекъсване при препълване на таймера.
Установява се в “1” при промяна на TCNT \$FFFF → \$0000;
- RTIF - флаг за RTI прекъсване (периодично);
- PAOVF - флаг за прекъсване при препълване на ПА;
- PAIF - флаг за прекъсване при импулс към ПА-входа (PAI);
- битове [3:0] - не се използват (винаги се четат 0 от тях).

Address: \$1025

| | Bit 7 | 6 | 5 | 4 | 3 | 2 | 1 | Bit 0 |
|--------|-------|------|-------|------|---|---|---|-------|
| Read: | TOF | RTIF | PAOVF | PAIF | | | | |
| Write: | | | | | | | | |
| Reset: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Подсистема за прекъсване в реално време (Real-Time Interrupt, RTI)

- **РАCTL** - Пулс-акумулатор контролен регистър
 - ❖ Определя периода за RTI подсистемата- битове **RTR[1:0]** ;
 - ❖ Контрол на функциите на Пулс-акумулатора;
 - ❖ Дефинира функцията на извод **PA3 - IC4/OC5**.
- DDRA7 - определя посоката на данните на бит7 от Порт А;
- PAEN - бит за разрешаване на Пулс-акумулатора;
- PAMOD - определя режима на Пулс-акумулатора.

Address: \$1026

| | Bit 7 | 6 | 5 | 4 | 3 | 2 | 1 | Bit 0 |
|--------|-------|------|-------|-------|-------|-------|------|-------|
| Read: | DDRA7 | PAEN | PAMOD | PEDGE | DDRA3 | I4/O5 | RTR1 | RTR0 |
| Write: | | | | | | | | |
| Reset: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Подсистема за прекъсване в реално време (Real-Time Interrupt, RTI)

- **РАCTL** - Пулс-акумулатор контролен регистър
(продължение)
- PEDGE - определя фронта за Пулс-акумулатора;
- DDRA3 - определя посоката на данните на бит3 от Порт А;
- I4/O5 - бит за дефиниране функциите на PA3 (IC4/OC5);
- RTR[1:0] - дефинират периодите на RTI прекъсванията
(виж табл., дадена по-рано).

Address: \$1026

| | Bit 7 | 6 | 5 | 4 | 3 | 2 | 1 | Bit 0 |
|--------|-------|------|-------|-------|-------|-------|------|-------|
| Read: | DDRA7 | PAEN | PAMOD | PEDGE | DDRA3 | I4/O5 | RTR1 | RTR0 |
| Write: | | | | | | | | |
| Reset: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Пулс-акумулатор

- Отделна подсистема, включваща **8-битов брояч (РАСТN)**, за работа в два режима (в зависимост от стойността на бит **РАМОD** от регистър РАСТL):
 - **обикновен броячен режим** (event counting): стойността на брояча се увеличава при постъпване на сигнал на входа му РАI (максимална честота: $E\text{-clock}/2$);
 - **разрешителен акумулиращ режим** (gated accumulation) - в този случай $E\text{-clock}/64$ тактува 8-битовия брояч, но само докато външният извод РАI е в активно ниво (с активно ниско или високо ниво: задава се от бит **РЕDGE** от регистър РАСТL).
- четене/запис от Пулс-акумулаторът - по всяко време;
- контролни битове - в регистри **РАСТL**, **TMSK2** и **TFLG2**.

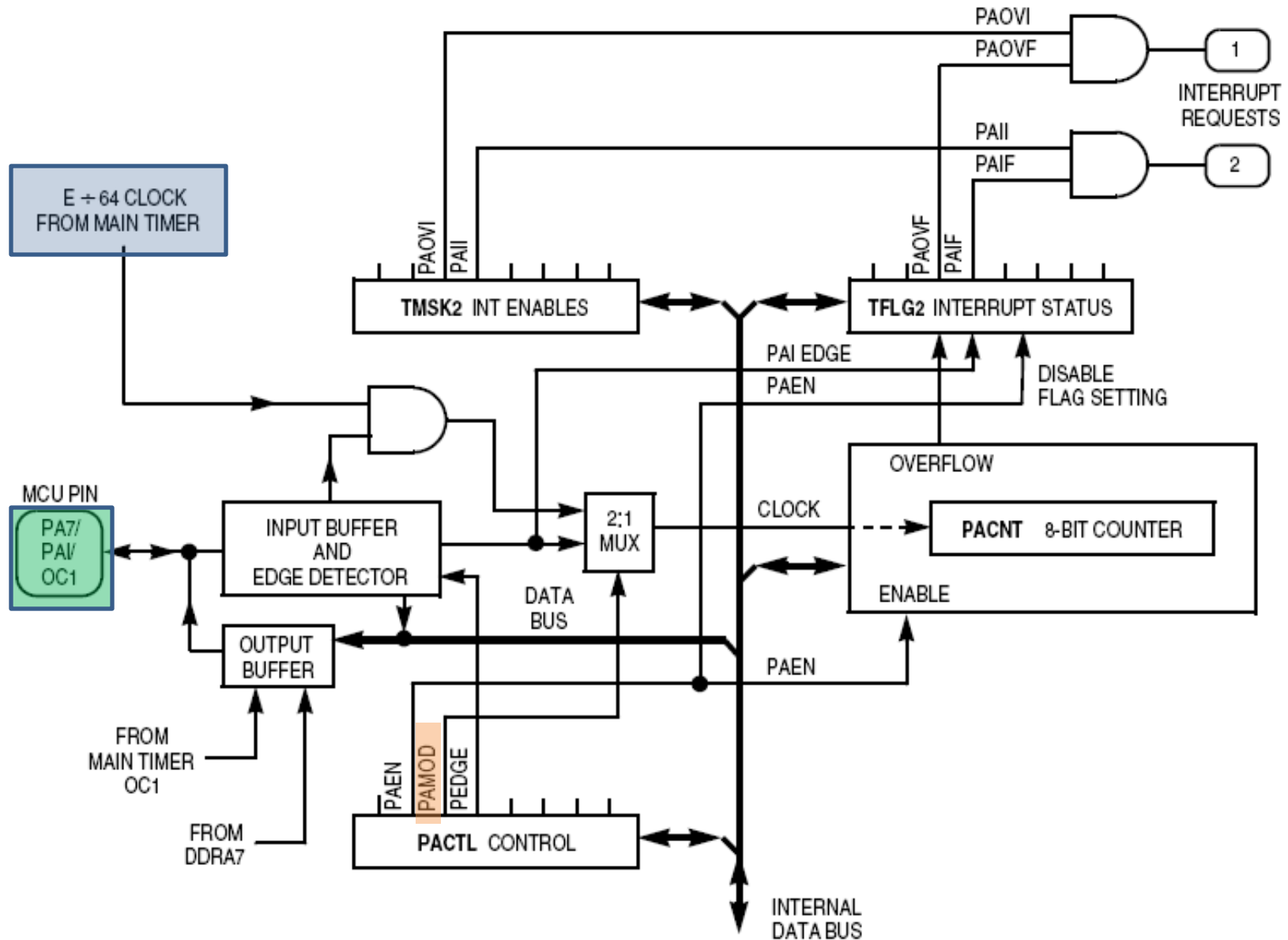
Пулс-акумулатор

- Дефиниране на циклите на Пулс-акумулатора (ПА) в двата възможни режима на работа:

$$E/2^6 = E/64 \text{ (8-битов брояч PACNT)}$$

| Crystal Frequency | E Clock | Cycle Time | E ÷ 64 | PACNT Overflow |
|-------------------|---------|------------|----------|----------------|
| 4.0 MHz | 1 MHz | 1000 ns | 64 μs | 16.384 ms |
| 8.0 MHz | 2 MHz | 500 ns | 32 μs | 8.192 ms |
| 12.0 MHz | 3 MHz | 333 ns | 21.33 μs | 5.461 ms |

Пулс-акумулатор – вътрешна структура



Пулс-акумулатор – регистри

□ РАCTL - Пулс-акумулатор контролен регистър

(тук: за контрол функциите на Пулс-акумулатора – 4 бита)

// DDRA7 - посоката на данните на бит7 от Порт А;

// I4/O5 - дефинира функциите IC4/OC;

// RTR[1:0] - задават периода на RTI прекъсването.

■ **PAEN** - бит за разрешаване на Пулс-акумулатора (ПА):

0 – забранен ПА (*по подразбиране след Reset*);

1 – разрешен ПА.

Address: \$1026

| | Bit 7 | 6 | 5 | 4 | 3 | 2 | 1 | Bit 0 |
|--------|-------|------|-------|-------|-------|-------|------|-------|
| Read: | DDRA7 | PAEN | PAMOD | PEDGE | DDRA3 | I4/O5 | RTR1 | RTR0 |
| Write: | | | | | | | | |
| Reset: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Пулс-акумулатор – регистри

- **РАCTL**- Пулс-акумулатор контролен регистър (продължение)
 - **РАMOD** - определя режима на ПА:
 - 0** - обикновен броячен режим;
 - 1** - разрешителен акумулиращ режим.
 - **PEEDGE** - бит за контрол вида на фронта за задействане брояча на ПА:

| бит РАMOD | бит PEEDGE | Реакция пр постъпване на тактов сигнал |
|--------------|---------------|--|
| 0 | 0 | Увеличаване съдържанието на брояча при постъпване на заден фронт на сигнал на входа РАI |
| 0 | 1 | Увеличаване съдържанието на брояча при постъпване на преден фронт на сигнал на входа РАI |
| 1 | 0 | Ниво "0" на вход РАI забранява броенето |
| 1 | 1 | Ниво "1" на вход РАI забранява броенето |

Пулс-акумулатор – регистри

- **PACNT** - Пулс-акумулатор броячен регистър. Функции:
 - Съхранява показанието за постъпилите към ПА външни събития през неговия вход PAI;
 - Позволява запис и четене от него по всяко време;
 - Четене: възможно е дори при неактивен вход PAI в разрешителен “gated” акумулиращ режим;
 - Съдържанието му НЕ СЕ ПРОМЕНЯ при Reset.

Address: \$1027

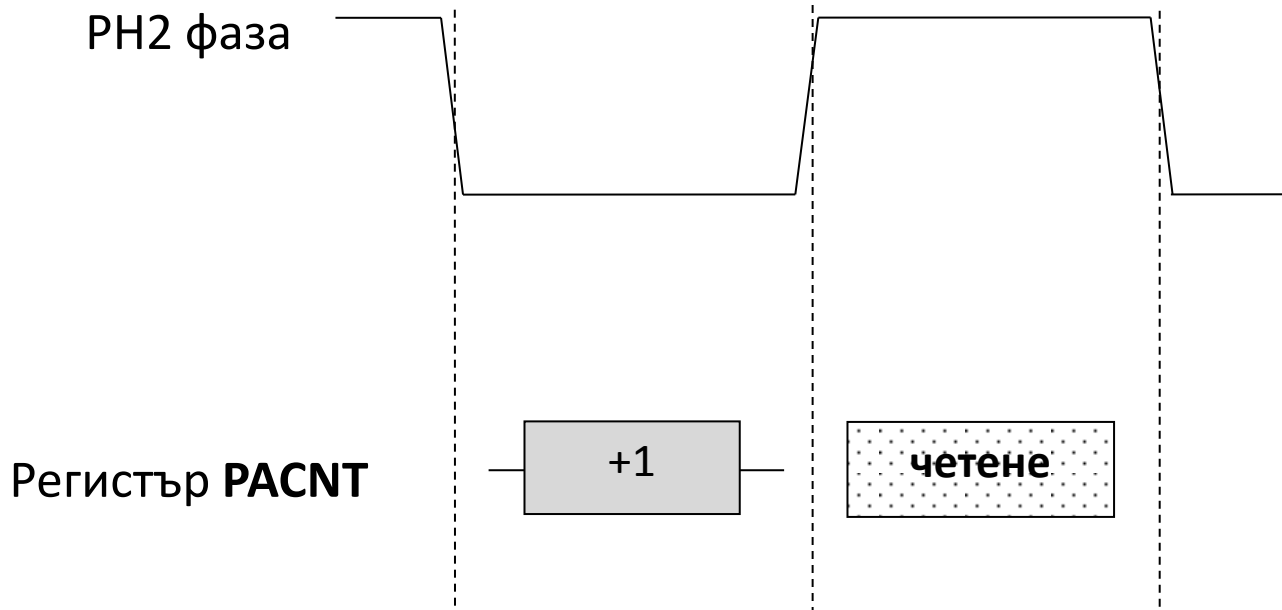
| | Bit 7 | 6 | 5 | 4 | 3 | 2 | 1 | Bit 0 |
|--------|---------------------------|-------|-------|-------|-------|-------|-------|-------|
| Read: | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Write: | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Reset: | Indeterminate after reset | | | | | | | |

Пулс-акумулатор – регистри

□ **PACNT** - Пулс-акумулатор броячен регистър
(продължение)

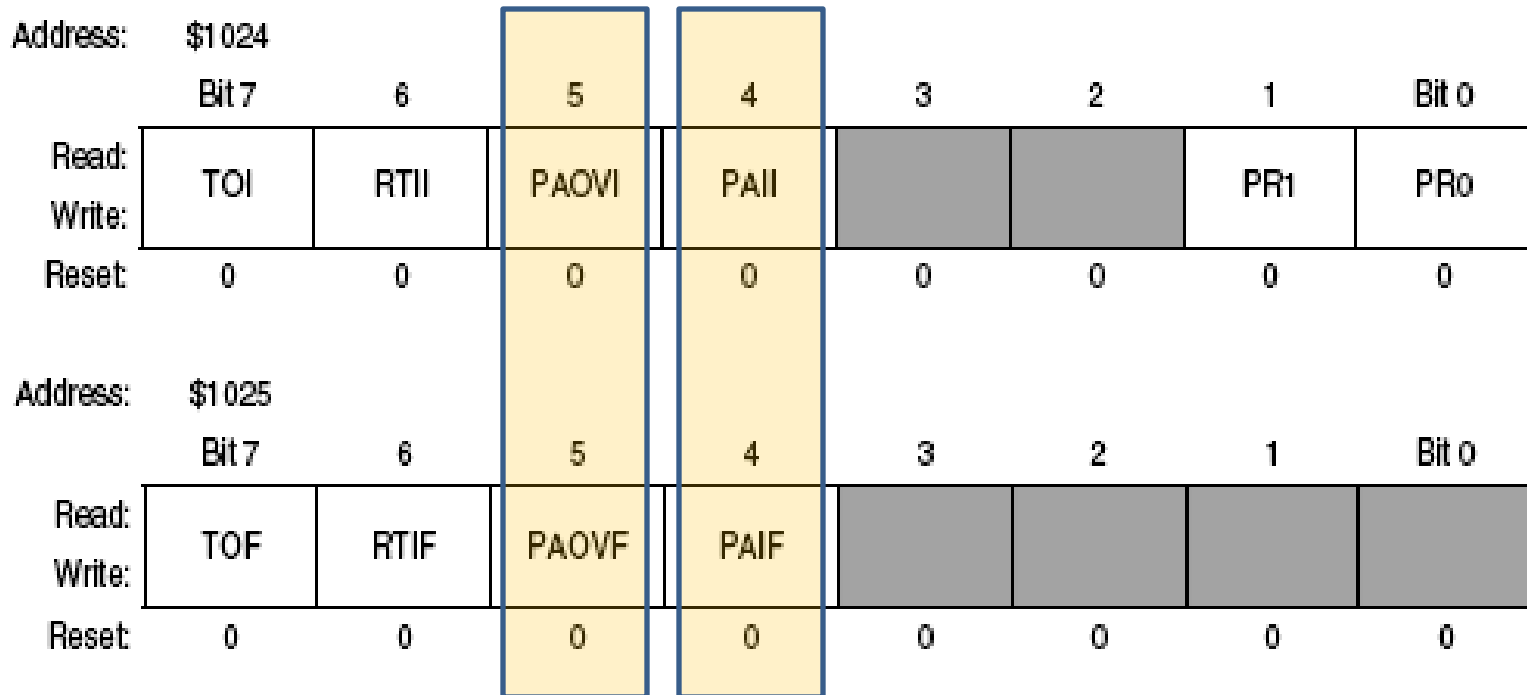
➤ синхронизация на ПА - по E-clock честота, фаза PH2:

- увеличаване на съдържанието (първа половина);
- четене (втора половина – високо ниво).



Пулс-акумулатор – регистри

- ✓ ПА – двойки статус-битове (флагове) и съответни маски за прекъсване - от таймерни регистри **TMSK2, TFLG2**.
 - Маски за прекъсване: PAOVI, PAII;
 - Флагове за прекъсване: PAOVF, PAIF.



Пулс-акумулатор – регистри

- ✓ ПА – статус-битове (флагове) и маски за прекъсване /от таймерни регистри TMSK2, TFLG2 - продължение/:
 - Маски за прекъсване: PAOVI, PAII;
 - Флагове за прекъсване: PAOVF, PAIF.
- **Флаг PAOVF** – вдига се в “1” всеки път при промяна съдържанието на ПА от \$FF в \$00 (препълване). Нулиране – запис на “1” в съответния бит (PAOVI-бит 5) от регистър TFLG2.
- **Маска PAOVI** – определя режима на работа на ПА по отношение на прекъсванията *при препълване* (не засяга установяването на флага PAOVF):
 - При PAOVI=“0” – вдигането на флага PAOVF се определя софтуерно за реализация на прекъсване при необходимост;
 - При PAOVI=“1” – генерира се хардуерно прекъсване всеки път, когато PAOVF се установи в “1”. Преди завършване сервизната процедура по обслужване на прекъсването флагът PAOVF следва да се нулира чрез запис в рег. TFLG2.

Пулс-акумулатор – регистри

- ✓ ПА – статус битове (флагове), битове (маски) за прекъсване /от таймерни регистри TMSK2, TFLG2 - продължение/:
- **Флаг PAIF** – установява се автоматично всеки път при детекция на фронт на вход PA7/PAI/OC1. Нулиране – запис на “1” в съответния бит (PAII-бит 4) от регистър TFLG2.
- **Маска PAII** – определя режима на работа на ПА по отношение на прекъсванията *при входен сигнал* (не засяга установяването на флага PAIF):
 - При PAII = “0” – вдигането на флага PAIF се определя софтуерно за реализация на прекъсване при необходимост;
 - При PAII = “1” – генерира се хардуерно прекъсване всеки път, когато PAIF се установи в “1”. Преди завършване сервизната процедура по обслужване на прекъсването флагът PAIF следва да се нулира чрез запис в регистър TFLG2.