

Микропроцесорски системи

4. блок: Дигитални улази и излази

др Срђан Митровић, дипл. инж.
e-mail: srdjan.mitrovic@va.mod.gov.rs

зимски семестар, 2016./2017. год.

Дигитални улази/излази

Чему служе?

Омогућавају управљање хардвером.
Омогућавају надгледање (*monitoring*) хардвера.

Да ли их имају сви микроконтролери?

ДА! 1–2 I/O пина најмање.
Уобичајено 8–32 пина.
Motorola HCS12 преко 90 I/O пинова.

Како су организовани?

- Најчешће као портови са 8 пинова. **Због чега баш 8?**
- Могу бити улазни или излазни.
- Најчешће су бидирекционални (двосмерни) или вишенаменски.

Управљање пиновима

Три регистра управљају пиновима за дигиталне улазе/излазе:

DDR – *Data Direction Register*. Сваки бидирекционални порт има свој DDR регистар са по једним битом за сваки бит порта. Намена пина (улаз или излаз) је одређена вредношћу бита у регистру.

PORT – *Port Register*. Овим регистром се управља напонским нивоом на пину.

PIN – *Port Input Register*. Најчешће је омогућено само читање овог регистра (*read-only*) и користи као провера стања улазних или излазних пинова.

Пример – Atmega16

DDxn	PORTxn	PUD (in SFIOR)	I/O	Pull-up	Comment
0	0	X	Input	No	Tri-state (Hi-Z)
0	1	0	Input	Yes	Pxn will source current if ext. pulled low.
0	1	1	Input	No	Tri-state (Hi-Z)
1	0	X	Output	No	Output Low (Sink)
1	1	X	Output	No	Output High (Source)

Read-Modify-Write Access

- Обратити пажњу код мултитаскинг система
- Један процес не сме приступити регистру који користи други процес
- Уколико желимо изменити само један бит у регистру, остали битови морају остати непромењени!
- Користити инструкције за операцију над битом!

Дигитални улази

Када се користе?

Када улазни сигнал може имати само две вредности.

Наравно, у једном тренутку времена. Сигнал се у општем случају мења са временом.

Логички нивои:

- логичка јединица – “high”
- логичка нула – “low”

Atmega16

- $V_{CC} = [4.5, 5.5]V$
- 1: $[0.6V_{CC}, V_{CC} + 0.5]V$
- 0: $[-0.5, 0.2V_{CC}]V$

Atmega16

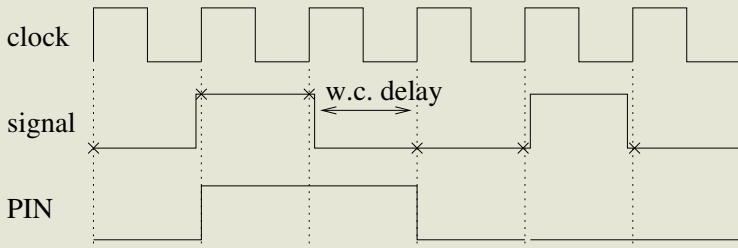
- $V_{CC} = [4.5, 5.5]V$
- 1: $[0.6V_{CC}, V_{CC} + 0.5]V$
- 0: $[-0.5, 0.2V_{CC}]V$
- интервал $(0.2V_{CC}, 0.6V_{CC})$ није дефинисан

Узорковање сигнала

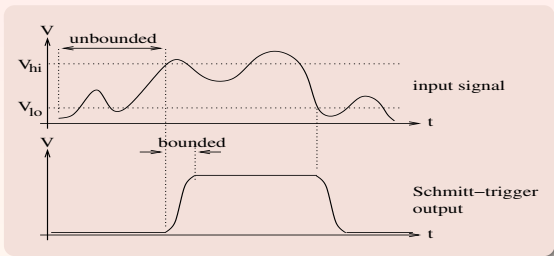
Једноставно решење

- “лечујемо” вредност PIN регистра
- са сваким доласком такта читамо “леч”
- промене читамо са закашњењем
- **спора промена сигнала?** (у односу на F такта)

Пример 2

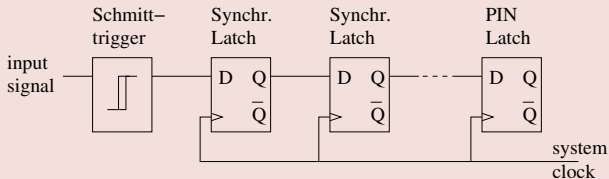


Узорковање сигнала



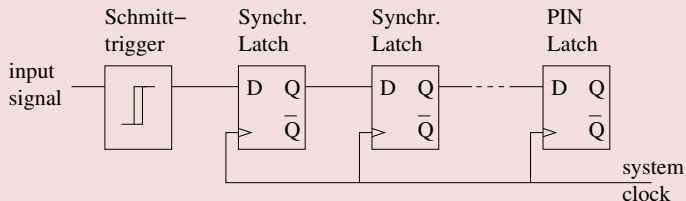
ШМИТОВО КОЛО

Решење #2



Узорковање сигнала

- Метастабилност (*meta-stability*)
- Вероватноћа одабирања недефинисаног стања је p по лечу
- Коришћењем k лечева смањује се на $p^k \ll 1$



- Недостатак? Време пропагације сигнала (кашњење)
- Пример Atmega16: $d_{in}^{min} = 0.5clk$, $d_{in}^{max} = 1.5clk$

Избегавање шума

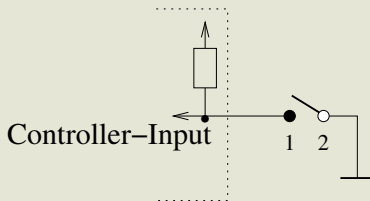
Неки контролери имају уграђен механизам за избегавање шума (\neq филтрирање). Уколико је ова опција сетована, контролер више пута читава вредност, нпр. k пута, и прихвата нову вредност ако је свих k одбирака идентично. Недостатак је утицај на укупно кашњење које је сада:

$$d_{in} = d_{latch} + d_{sync} + d_{ncanc}$$

Pull Resistors – Pull Отпорници

У контролерима се користе на два начина, као:

- Pull-UP
- Pull-Down



Чему служе?

Пинови контролера не смеју да „висе“, јер то узрокује шум („у великим количинама“).

Дигитални излаз

- Омогућава постављање напона пина на жељени ниво.
- **Atmega16** „low“ = max 0.7V; „high“ = min 4.2V
- Све док је DDR подешен за излаз, пинови се сетују према PORT регистру.
- Излазни пин може да „даје“ или „вуче“ струју
- Вредности струје за излазне пинове микτροконтролера су [4 – 20]mA¹, највише.

¹20mA је довољно за напајање стандардне LED диоде

Излазни пинови су критичнији од улазних.

Уколико излазни пин физички спојимо на GND и сетујемо га на 1 долази до **кратког споја!**

- Произвођачи дозвољавају краткотрајан кратак спој.
- Обратити пажњу при пројектовању хардвера!
- Програмска провера?
 - Очитавањем PIN регистра, али ...
 - Због кашњења, КС је могуће детектовати тек пошто је настао.
 - Ако се детектује КС, одговарајући пин се сетује као излаз и обавештава се корисник.

Који регистар прво поставити

- DDR?
 - PORT?
-
- Ако контролер не користи PORT битове за друге намене (као *Atmega16*), прво сетујемо PORT, а затим DDR.
 - Код *Atmega16* је мало компликованије, јер PORT управља „Pull-Up“ отпорницима.
 - Желимо да прво поставимо „1“, и сетујемо PORT па DDR, краткотрајно ћемо укључити „Pull-Up“ отпорнике, пре него што порт постане излазни.

Закључак

Ако контролер дозвољава, а хардвер не смета, сетујемо прво PORT па DDR. Увек треба проверити могуће нежељене ефекте.

порт В

пинови 0 и 1 „high“

пинови 2 и 3 „low“

пинови 4–7 улазни

пинови 6 и 7 pull-up

С кôд

```
unsigned char i;
```

```
...
```

```
/* definisati pull-ups i postaviti izlaze na "high" */
```

```
/* definisati smer za pinove porta */
```

```
PORTB = (1<<PB7)|(1<<PB6)|(1<<PB1)|(1<<PB0);
```

```
DDRB = (1<<DDB3)|(1<<DDB2)|(1<<DDB1)|(1<<DDB0);
```

```
/* Ubaciti nop za sinhronizaciju*/
```

```
_NOP()
```

```
/* Očitati vrednost pinova*/
```

```
i = PINB;
```

```
...
```

порт В

пинови 0 и 1 „high“

пинови 2 и 3 „low“

пинови 4–7 улазни

пинови 6 и 7 pull-up

Асемблерски кôд

...

```
; definisati pull-ups i postaviti izlaze na "high"
```

```
; definisati smer za pinove porta
```

```
ldi r16,(1<PB7)|(1<PB6)|(1<PB1)|(1<PB0)
```

```
ldi r17,(1<DDB3)|(1<DDB2)|(1<DDB1)|(1<DDB0)
```

```
out PORTB,r16
```

```
out DDRB,r17
```

```
; Ubaciti nop za sinhronizaciju
```

```
nop
```

```
; Očitati vrednost pinova
```

```
in r16,PINB
```

...

порт B

пинови 0 и 1 „high“

пинови 2 и 3 „low“

пинови 4–7 улазни

пинови 6 и 7 pull-up

Asembler Atmega16

...

```
; definisati pull-ups i postaviti izlaze na "high"
```

```
; definisati smer za pinove porta
```

```
ldi r16,0xC3
```

```
ldi r17,0x0F
```

```
out PORTB,r16
```

```
out DDRB,r17
```

```
; Ubaciti nop za sinhronizaciju
```

```
nop
```

```
; Očitati vrednost pinova
```

```
in r16,PINB
```

...

Систематизација

- 1 **Увод**
 - Digital I/O
- 2 **Дигитални улази**
 - Digital Input
- 3 **Дигитални излази**
 - Digital Output
 - Digital Sampling
 - Noise Cancellation

Задатак: Микроконтролер има задатак да надгледа дигитални сигнал чији су импулси трајања $\geq 1\mu s$. Ово се обавља непрекидним читавањем стања пина. Читање траје 1 циклус такта, обрада траје 3 циклуса. Колико треба да буде брз микроконтролер како не би пропустио ниједан импулс?

Микропроцесорски системи

4. блок: Дигитални улази и излази

др Срђан Митровић, дипл. инж.
e-mail: srdjan.mitrovic@va.mod.gov.rs

зимски семестар, 2016./2017. год.