

Programmation du timer/counter0

I. Trouver les valeurs à mettre dans les bits CLKPS3-0 de registre CLKPR.
(CLKPS3 CLKPS2 CLKPS1 CLKPS0) =

Justification:

II. Donner les valeurs des bits des registres de contrôle TCCR0A et TCCR0B en justifiant.

• CS02 CS01 CS00=(.....)₂

Justification:

• WGM₂=....., WGM₁=....., WGM₀=.....

Justification:

• COM0A1=.....et COM0A0=.....

Justification:

• COM0B1=.....et COM0B0=.....

Justification:

• FOC0A=..... ; FOC0B=.....

Justification:

Bit	7	6	5	4	3	2	1	0	
	COM0A1	COM0A0	COM0B1	COM0B0	-	-	WGM01	WGM00	TCCR0A
									=0x.....

Bit	7	6	5	4	3	2	1	0	
	FOC0A	FOC0B	-	-	WGM02	CS02	CS01	CS00	TCCR0B
									=0x.....

III. Donner la valeur du registre de direction nécessaire pour configurer le pin/OC0.....

	7	6	5	4	3	2	1	0	← Les indices
DDR.....									=0x.....
	.	.	.	.	.	.	.	.	← Mettre e pour entrée et s pour sortie

IV. Donner les valeurs des bits utilisés de registre TIMSK en justifiant.

TOIE0=..... ; OCIE0A=..... ; OCIE0B=.....;

Justification:

.....

.....

.....

.....

.....

Bit	7	6	5	4	3	2	1	0	
	TOIE1	OCIE1A	OCIE1B	-	ICIE1	OCIE0B	TOIE0	OCIE0A	TIMSK
									=0x.....

```
#include <tiny2313.h>
```

```
{
```

```
{
```

```
}
```

```
void main(void)
```

```
{
```

```
#pragma optimize-
```

```
CLKPR=.....;
```

```
CLKPR=.....;
```

```
#ifdef _OPTIMIZE_SIZE_
```

```
#pragma optimize+
```

```
#endif
```

```
PORT...=.....;
```

```
DDR...=.....;
```

```
TCCR0A=.....;
```

```
TCCR0B=.....;
```

```
TCNT0=.....;
```

```
OCR0A=.....;
```

```
OCR0B=.....;
```

```
TIMSK=.....;
```

```
// Global enable interrupts
```

```
#asm("sei")
```

```
while (1)
```

```
{
```

```
};
```

```
}
```