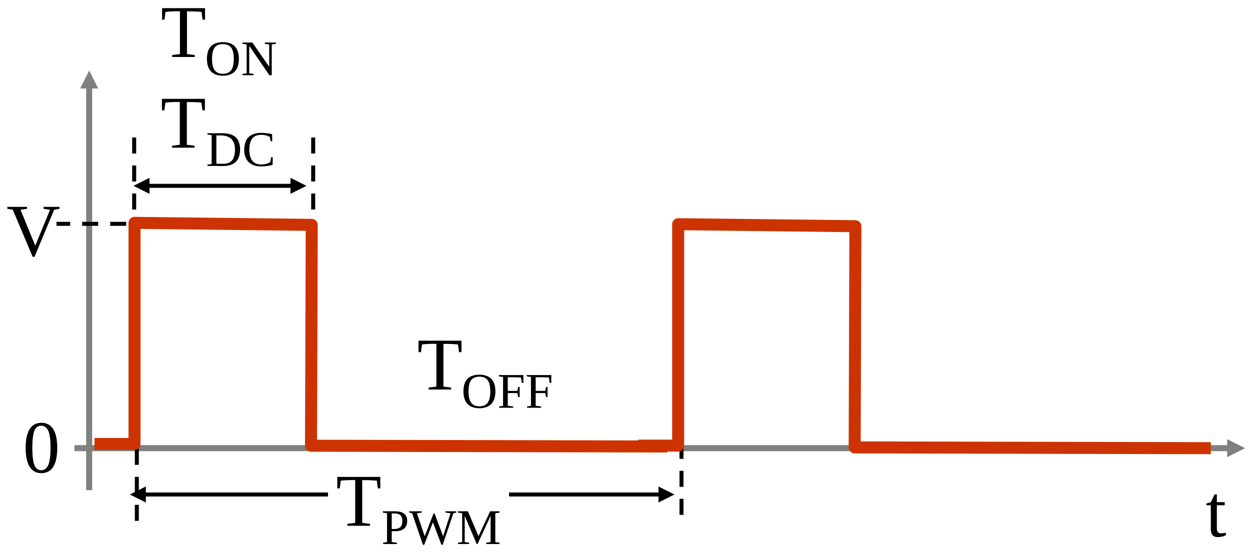
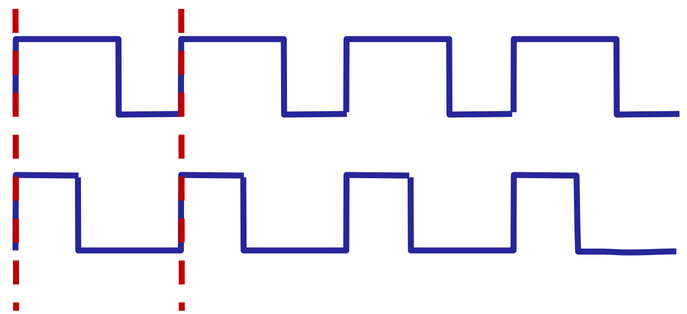
Pulse Width Modulation (PWM)

Square Wave



Duty Cycle T_{DC}/T_{PWM} x 100%

- Special Modules generates continuous square wave
- Pin 17 (CCP1) & 16 (CCP2)
- Must have same T but can have different duty cycle
- Rise of signals always coincide



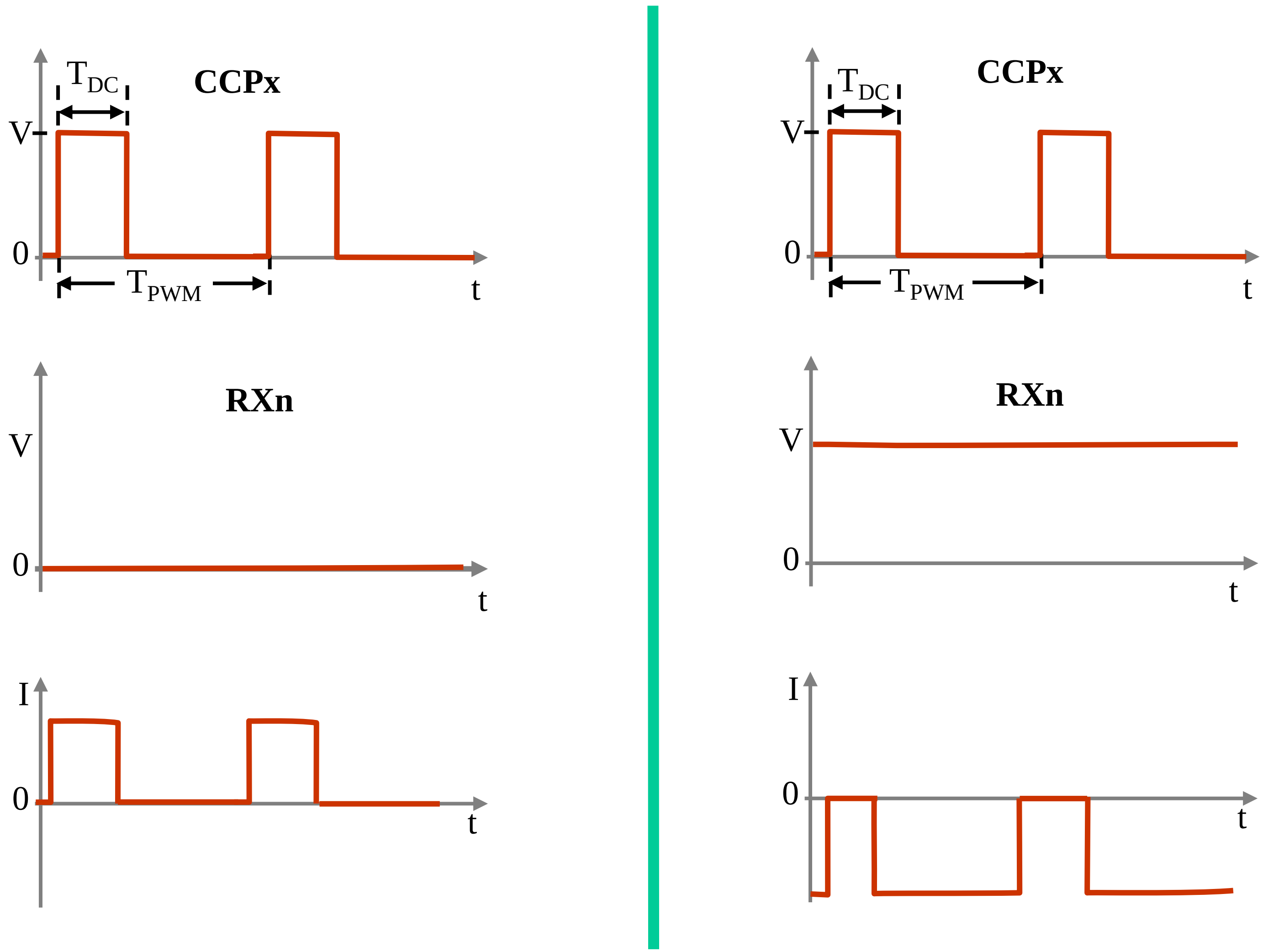
Uses

- Blink an LED
- Play a tone on speaker
- Mainly as a control – usually speed but also colour and brightness

LED Blink & Brightness

- If T long, can see on and off
- If T short, see blend.
- Large dutycycle (mostly on, some off) is bright.
- Small dutycycle (mostly off, some on) is dim.
- Connect CCPx and output RXn can control colour of two colour LED.

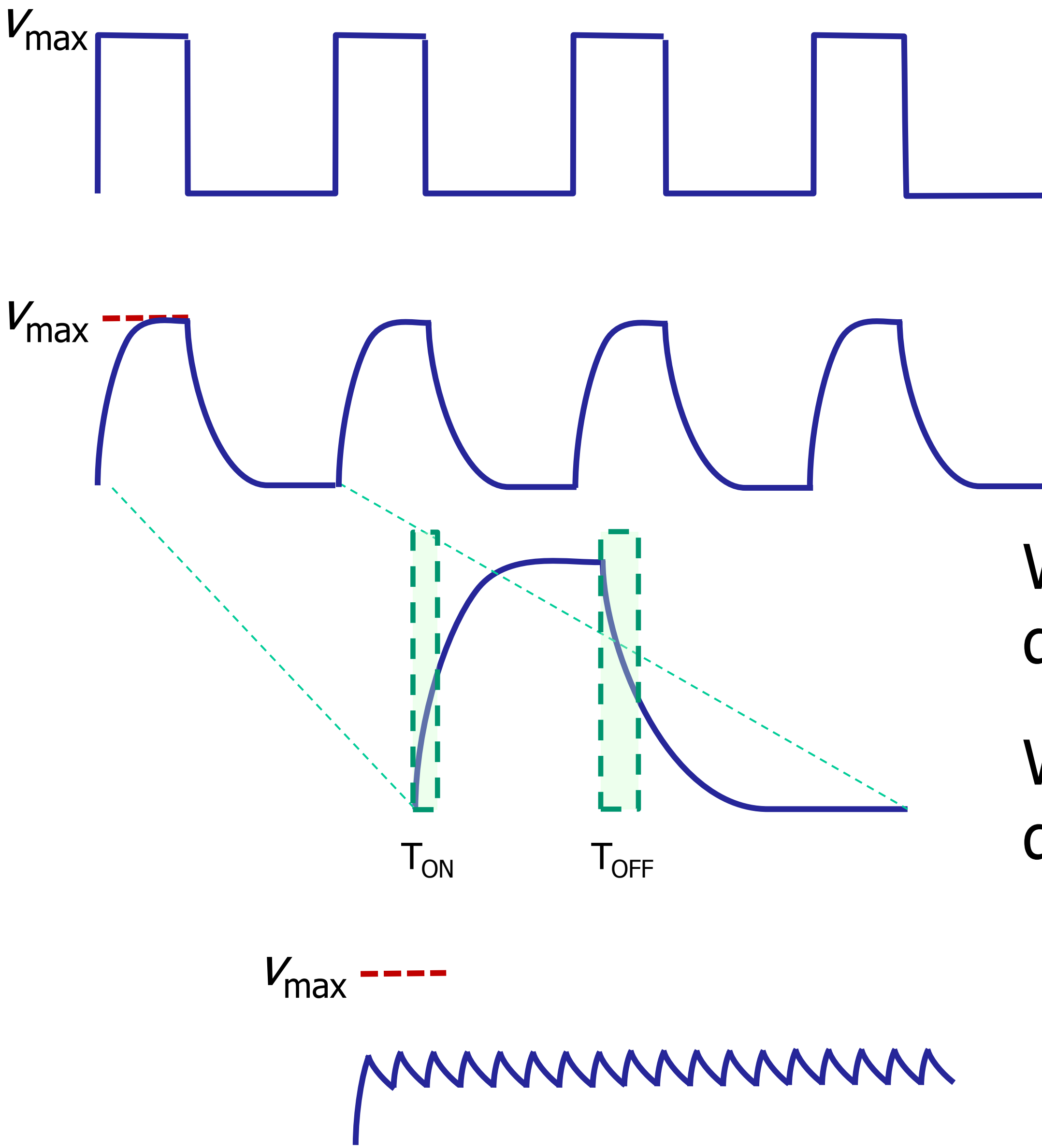
Conventional current flows from high to low.



Always analyze V-t signals using graphs!

Speed Control - Motors

- Long T see start stop
- Small T, inertia keeps motor turning.
- With small dutycycle average speed lower than with large dutycycle
- Can use with a second pin RXn to control direction of rotation



Ideal motor
Instant on and off

Real motor
Has inertia

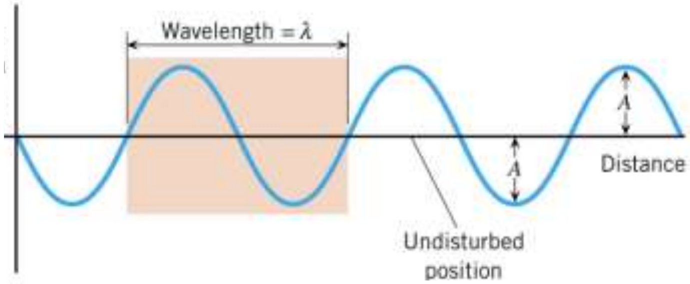
When T_{ON} is short, motor does not reach full speed

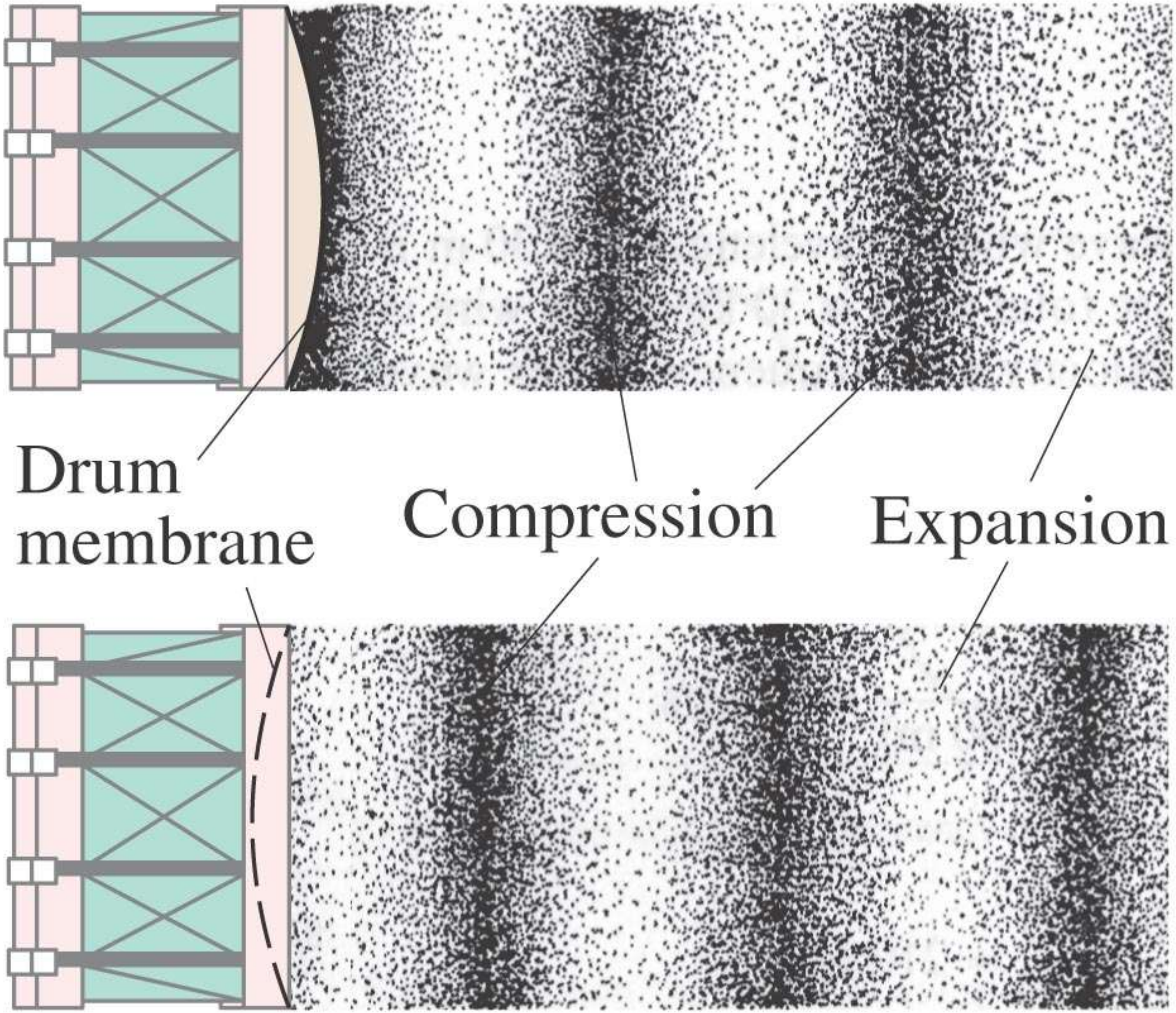
When T_{OFF} is short, motor does not stop completely

Motor speed reduced

Sound Generation

- Pure tone is a sine wave
- Square wave approximates
- f = 1/T
- Need to know frequency of note A, C, G, etc




Drum
membrane
Compression
Expansion

PWM Software

#include <timers.h>

OpenTimer2(unsigned char *config*) **prescaler needed

#include <pwm.h>

OpenPWMx(unsigned char *period*)

SetDCPWMx(unsigned int *dutycycle*)

ClosePWMx(void)

x can be 1 (pin 17) or 2 (pin 16) but period must be the same for both

OpenPWMx(unsigned char *period*)

period :   0 to 255

$T_{PWM} = [(period) + 1] \times T_{CY} \times prescaler$

SetDCPWMx(unsigned int *dutycycle*)

dutycycle: 0 to 1023

$T_{DC} = dutycycle \times \frac{1}{4} T_{CY} \times prescaler$

Remember $T_{DC} = T_{ON}$, $T_{DC} + T_{OFF} = T_{PWM}$

$$\%dutycycle = T_{dutycycle} / T_{PWM}$$

$$= [\tfrac{1}{4} \times \textit{dutycycle}] / [\textit{period} + 1] \times 100\%$$

If exceeds 100% always on!

If period = 2, what %DCs can you have?

OpenTimer2(TIMER_INT_OFF & T2_POST_1_1 & prescaler)

prescaler	Value
T0_PS_1_1	1
T0_PS_1_4	4
T0_PS_1_16	16

```
#include <p18F4525.h>
#include <pwm.h>
#include <timers.h>
#include “../Functions/osc.h”

void main(void)
{
 OpenTimer2(... & T2_PS_1_4 );
 set_osc_4MHz();
 OpenPWM1(200);   // What is TPWM
 SetDCPWM1(158);  // What is %DC
 while(1);
}
```

	Maximum T_{PWM} (sec)		
f_{OSC} (MHz)	T2_PS_1_1	T2_PS_1_4	T2_PS_1_16
32			
16			
8			
4			
2			
1			
0.500			
0.250			
0.125			

	Maximum T_{PWM} (ns)		
T_{CY} (ns)	T2_PS_1_1	T2_PS_1_4	T2_PS_1_16
125	256x125 = 32 000	256x125x4= 128 000	256x125x16= 512 000
250	64 000	256 000	1 024 000
500	128 000	512 000	2 048 000
1 000	256 000	1 024 000	4 096 000
2 000	512 000	2 048 000	8 192 000
4 000	1 024 000	4 096 000	16 384 000
8 000	2 048 000	8 192 000	32 768 000
16 000	4 096 000	16 384 000	65 536 000
32 000	8 192 000	32 768 000	131 072 000

	Minimum T_{PWM} (sec)		
f_{OSC} (MHz)	**T2_PS_1_1**	**T2_PS_1_4**	**T2_PS_1_16**
32			
16			
8			
4			
2			
1			
0.500			
0.250			
0.125			

	Minimum T_{PWM} (ns)		
T_{CY} (ns)	**T0_PS_1_1**	**T0_PS_1_4**	**T0_PS_1_16**
125	1x125 = 125	1x125x4 = 500	1x125x16 = 2 000
250	250	1 000	4 000
500	500	2 000	8 000
1 000	1 000	4 000	16 000
2 000	2 000	8 000	32 000
4 000	4 000	16 000	64 000
8 000	8 000	32 000	128 000
16 000	16 000	64 000	256 000
32 000	32 000	128 000	512 000

What clock speed, prescaler, period, and and dutycycle to generate most precisely a 2550 Hz square wave with a 30% dutycycle? What is the actual frequency and percent dutycycle?

$$1/2550\ \text{Hz} = 392\ 156.86\ \text{ns}$$

$$T_{PWM} = [(\textit{period}) + 1] \times T_{CY} \times \textit{prescaler}$$

$$\textit{period} + 1 = T_{PWM} / [T_{CY} \times \textit{prescaler}]$$

	period + 1 = T_{PWM} / [T_{CY} x prescaler]		
T_{CY} (ns)	**T2_PS_1_1**	**T2_PS_1_4**	**T2_PS_1_16**
125	3 137	784	**196**
250	1 569	392	98
500	784	**196**	49
1 000	392	98	25
2 000	**196**	49	12
4 000	98	25	
8 000	49	12	
16 000	25		
32 000	12		

$$\%dutcycle = T_{dutycycle} / T_{PWM}$$

$$= [1/4 \times \textit{dutycycle}] / [\textit{period} + 1] \times 100\%$$

$$\textit{dutycycle} = 4 \times \%dutcycle \times [\textit{period} + 1]$$

$$= 4 \times 0.30 \times 196$$

$$= 235$$

$$T_{PWM} = [\textit{period} + 1] \times T_{CY} \times \textit{prescaler}$$

$$= 196 \times 2000 \times 1$$

$$= 392\,000 \text{ ns}$$

$$f_{PWM} = 1/\,(0.000\,392\,000 \text{ s}) = 2551.0 \text{ Hz}$$

$$\%dutcycle = [\tfrac{1}{4} \times \textit{dutycycle}\,] \,/\, [\textit{period} + 1] \times 100\%$$

$$= [\tfrac{1}{4} \times 235] \,/\, [196] \times 100\%$$

$$= 29.97\%$$

What clock speed, prescaler, period, and dutycycle to generate most precisely a 452 Hz square wave with a 45% dutycycle? What is the actual frequency and percent dutycycle?