**Pendientes y avances del 22 de mayo al 5 de Junio**

1. Presión

* Organizar todos los elementos de presión en un archivo (columna video, Graficas, etc)
* Regular válvula mecánica a 30cmH2O ( Teflón y 2 vueltas menos)
* Se instaló una válvula PEEP impresa con un adaptador para el AMBU
* Si paciente espira durante inspiración (aumento de presión mayor al esperado) motor suspende medio ciclo para sincronizarse con paciente. Igual en lo inverso

1. Volumen

* Aprobada donación válvula intermedia sensores de Robert Reed (USA)

1. Mecánica

* Mejorar retornos a origen iniciales y sobre la marcha
* Programar interrupt para suspender alarmas
* Medidas de presión (2 sensores) cada 100 milisegundos
* Construir maquinas 3 y 4

1. Pruebas

* prueba con pulmón artificial BuhlerFarma (Lunes 8 de Junio)

1. Financiamiento

* Carta a Rafael Henríquez

1. Publicar github. Actualizar tabla valores médicos y testing

**Ideas interrupt o millis()**

Usar millis siempre que quieras, interrupts solo cuando indispensable

Demonstration code for several things at the same time

<https://forum.arduino.cc/index.php?topic=223286.0>

<https://forum.arduino.cc/index.php?topic=503368.0>

Opcion multiple <https://www.instructables.com/id/Non-blocking-Virtual-Delay-Library-for-the-Arduino/>

Ultima instancia <https://code.google.com/archive/p/rogue-code/wikis/ToneLibraryDocumentation.wiki>

**Medidas presion cada 100 milisegundos**

USAR BMP en Normal mode, cada 100ms

Ver tabla 15, elevator

single burst read

**Alarma cada 500 milisegundos**, numero pitidos segun alarma, luego pausa de 1 segundo

**Interrupts**

<https://www.instructables.com/id/Arduino-Timer-Interrupts/>

Hay dos interrupt pins para input, pero los interrupt pueden trabajar sin pin.

Configurar tiempos de muestreo

compare match register = [ 16,000,000Hz/ (prescaler \* desired interrupt frequency) ] - 1

remember that when you use timers 0 and 2 this number must be less than 256, and less than 65536 for timer1

One last thing to note- certain timer setups will actually disable some of the Arduino library functions. Timer0 is used by the functions millis() and delay(), if you manually set up timer0, these functions will not work correctly.

Additionally, all three timers underwrite the function analogWrite(). Manually setting up a timer will stop analogWrite() from working.

you can't use millis() to create a delay IN AN ISR. And, you can't use delay() IN AN ISR.

<https://circuitdigest.com/microcontroller-projects/arduino-interrupt-tutorial-with-examples>

<https://aprendiendoarduino.wordpress.com/tag/isr/>

Placa int.0 int.1 int.2 int.3 int.4 int.5

Nano, Uno, Ethernet 2 3

<https://www.arduino.cc/reference/en/language/functions/external-interrupts/attachinterrupt/>

**arduino I2C interrupt**

<https://arduino.stackexchange.com/questions/9388/reading-i2c-sensors-with-a-timer-interrupt>

<http://www.gammon.com.au/i2c>

[Where is my I2C address](http://blascarr.com/i2c-address/)

**SPI interrupts**

<https://www.arduino.cc/en/Reference/SPIusingInterrupt>

<https://www.arduino.cc/en/Tutorial/DigitalPotControl>

<https://www.allaboutcircuits.com/technical-articles/using-interrupts-on-arduino/>

| method | resolution | max. time | typ. overhead |

|----------------|---------------|-----------------|----------------|

| `millis()` | 1 – 2 ms | 49.7 days | 0.69 – 1.3 µs |

| `micros()` | 4 µs | 71.6 min | 2.8 – 2.9 µs |

| Timer 1 | 0.0625 µs | 4.096 ms | 0.25 – 0.5 µs |

| pin toggling | scope-limited | ∞ | 0.125 µs |

| cycle counting | 0.0625 µs | boredom-limited | 0 |

| looping | N.A. | N.A. | 0.25 µs |