

Entregable Ejercicio Final

Diego Giordano

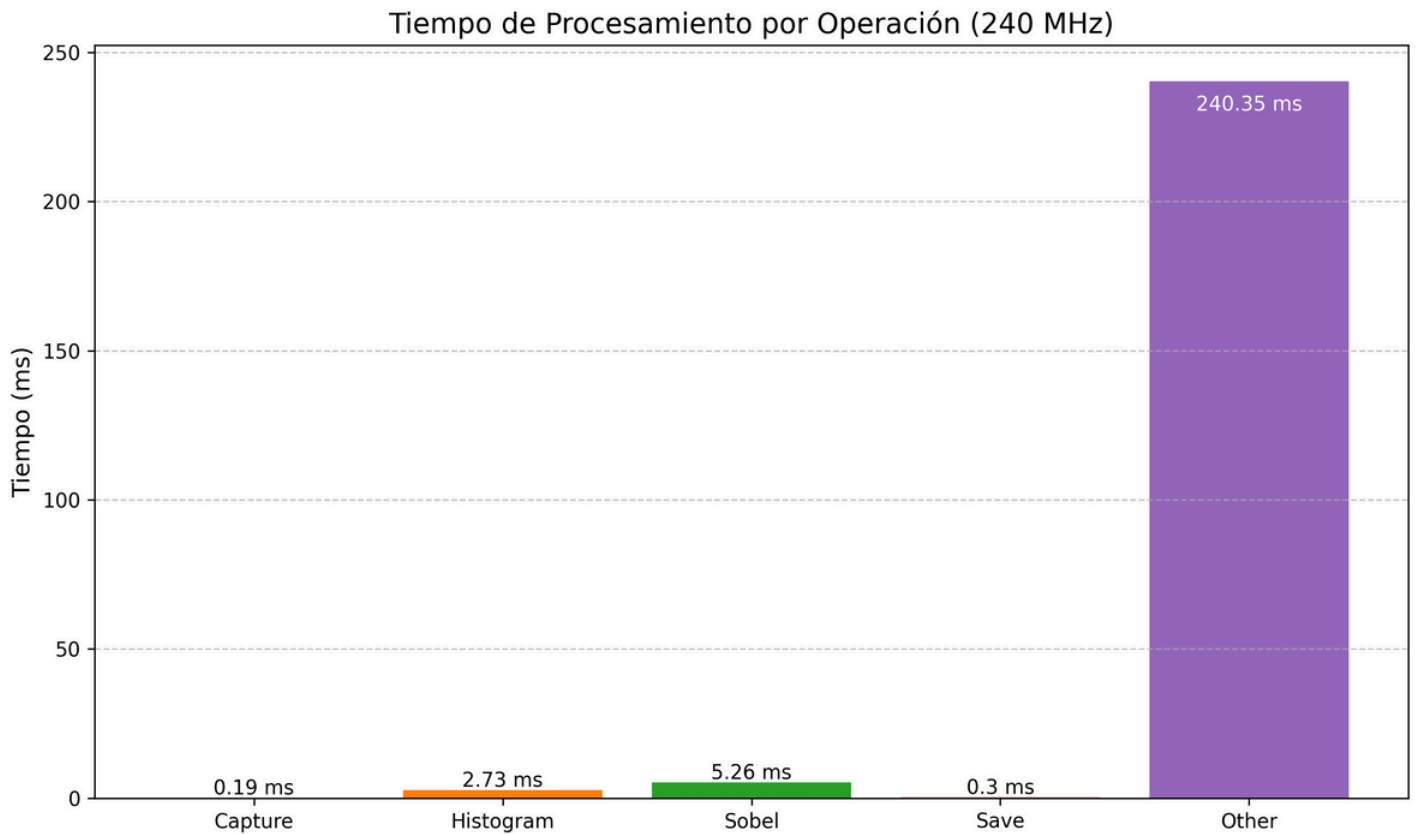
Benjamín Rogers

1-

2.- Frames por segundo

Frecuencia CPU	FPS
240 MHz	2.98
160 MHz	2.79
80 MHz	2.51

3.- Grafico de Barras, donde muestra el tiempo de carga de cada operación



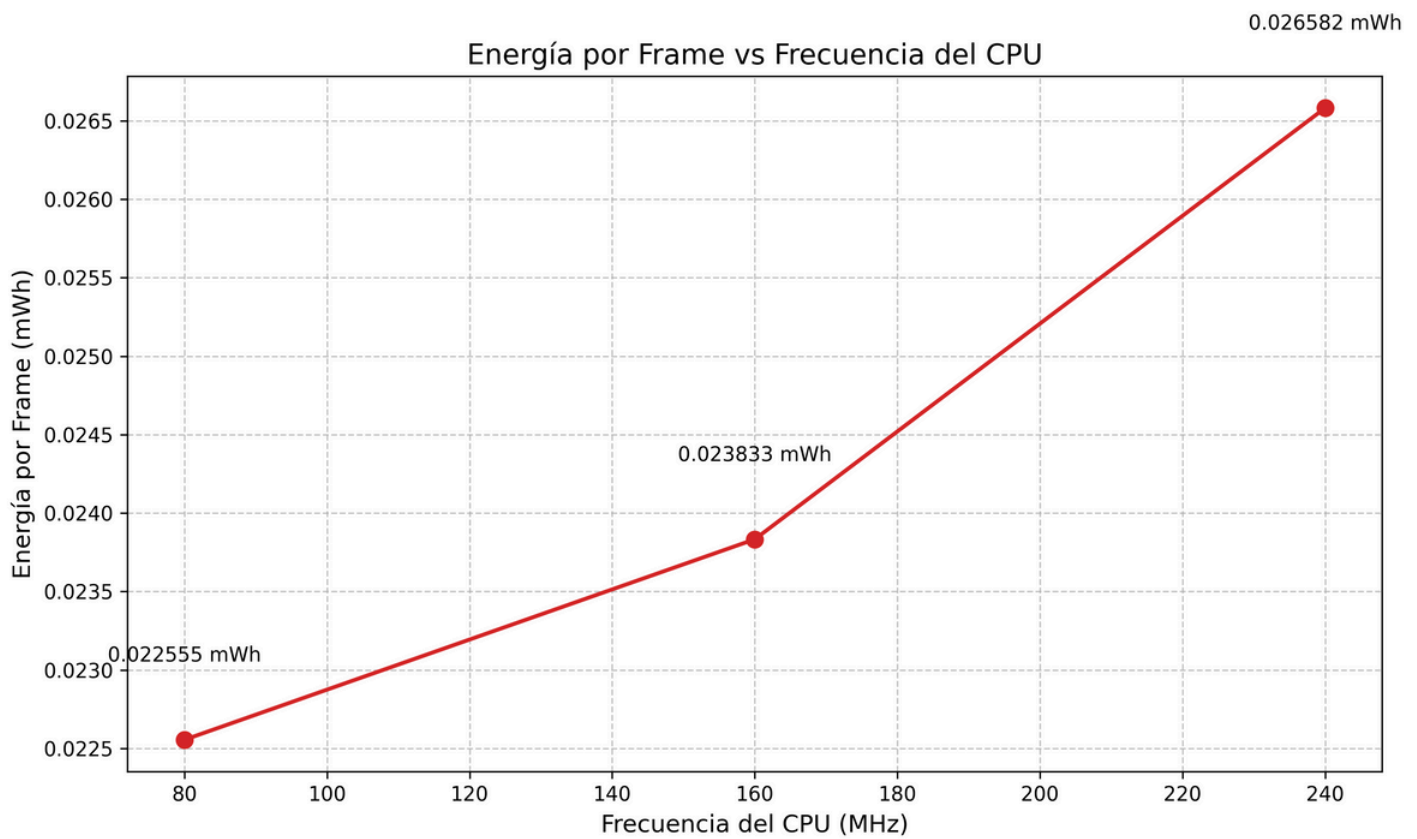
Estos tiempos "Other" incluyen:

1. Transferencia de datos entre la cámara y el ESP32-CAM
2. Operaciones de escalado de la imagen (de 640x480 a 320x240)
3. Operaciones de memoria (asignación, liberación, y copia)
4. Tiempo de planificación del sistema operativo FreeRTOS
5. Comunicación con la memoria PSRAM

Lo que toma más tiempo es la transferencia de datos y manejo general de la memoria, siendo varias veces más costoso que las operaciones definidas en el código. No hay un cuello de botella muy marcado pero podríamos decir que es la operación Sobel dentro de lo que si definimos nosotros. No es posible optimizar el tiempo de "Other" porque es un overhead de nuestro ESP32, son las funciones básicas para operar.

El histograma del ecualizador tiene un tiempo de proceso de 2,73 ms

El guardado de la imagen tiene un tiempo del proceso de 0,3 ms



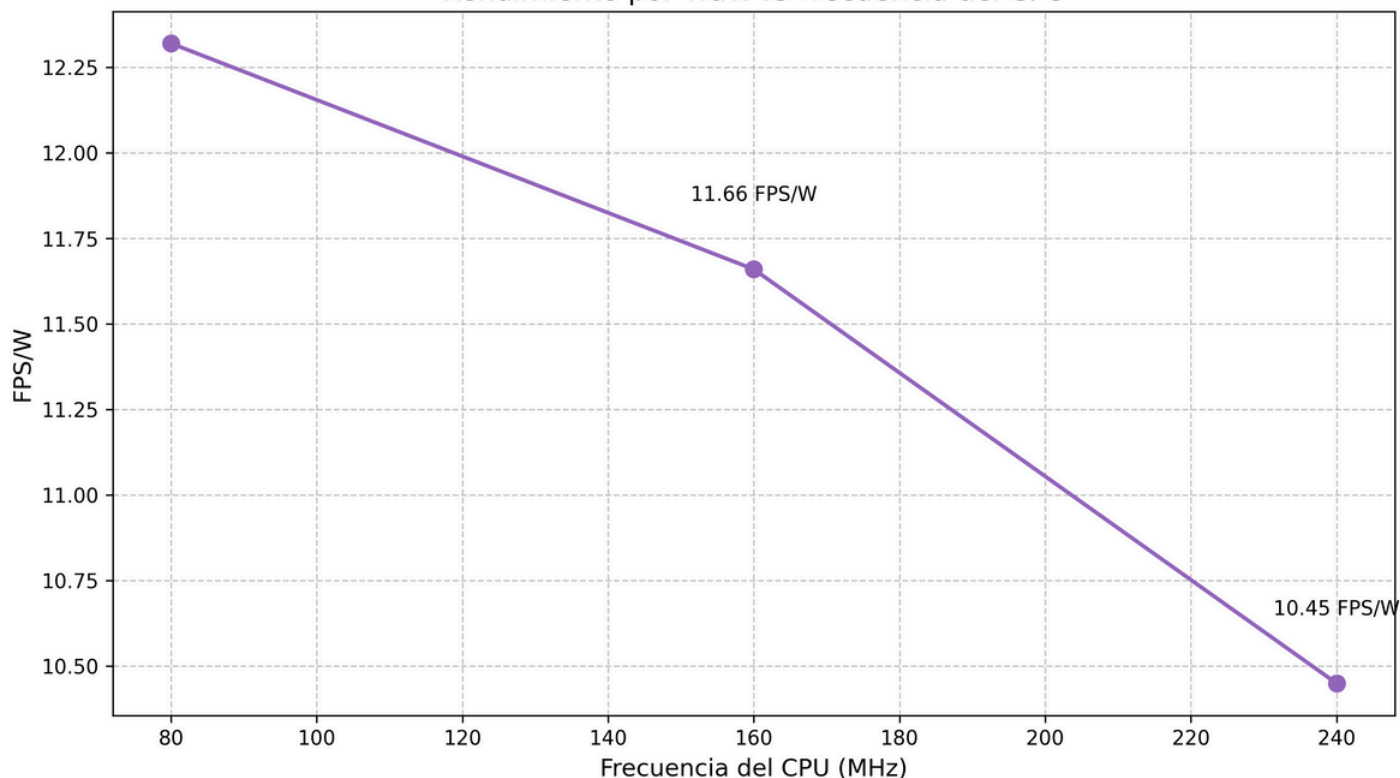
4.- Cantidad de energía que consume al procesar un frame vs Cantidad de energía que consume al guardar un frame

Frecuencia CPU	Potencia al procesar un frame	Energía por frame
240 MHz	370.00 mW	0.026582 mWh
160 MHz	330.00 mW	0.023833 mWh
80 MHz	290.00 mW	0.032060 mWh

Frecuencia CPU

12.32 FPS/W (ÓPTIMO)

Rendimiento por Watt vs Frecuencia del CPU



5.- Nueva tabla al cambiar la frecuencia en el microcontrolador

Tabla nueva Frecuencia

Frecuencia CPU	FPS	Energía por Frame (mWh)
80 MHz	2.51	0.03206
160 MHz	3.85	0.029569
240 MHz	3.87	0.031033

Prácticamente no varia al reducir la frecuencia de 240 MHz a 160 MHz, indicando que varia por factores como la velocidad de cámara o la transferencias de la datos, no por velocidad de CPU.

6.- Requisitos energía mínima para soportar aplicación durante 10 días a máximo FPS

Requisitos de batería (10 días)		
Frecuencia CPU	Energia total requerida	Capacidad de Bateria
80 MHz	69599.99 Wh	18810808.00 mAh a 3.7V
160 MHz	88800.00 Wh	21405404.00 mAh a 3.7V
240 MHz	79200.00 Wh	24000000.00 mAh a 3.7V

7.- Medida de Desempeño (FPS/W)

FPS/Potencia(W)

Frecuencia CPU	FPS/W
80 MHz	8.66
160 MHz	8.46
240 MHz	8.04

La configuración optima seria 80 MHz con 8.66 FPS/W, entre todas las combinaciones el de la frecuencia de 80MHz es la mas eficiente, ya que tiene es el que mayor tiene entre rendimiento y consumo de energía. aunque tenga menos FPS que otras configuraciones, tiene un consumo significativamente menor.

Código de muestra

Problems 17 Output Debug Console Terminal Ports

zsh + - - -

I (39018) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240
I (39408) ESP32-CAM: Memory: Free DRAM: 2750163 bytes, Free PSRAM: 2511348 bytes
I (39418) ESP32-CAM: === Results for 80 MHz ===
I (39418) ESP32-CAM: Avg. FPS: 2.51
I (39418) ESP32-CAM: Avg. frame time: 388.20 ms
I (39418) ESP32-CAM: Avg. capture time: 0.55 ms (0.1%)
I (39418) ESP32-CAM: Avg. histogram time: 3.02 ms (0.8%)
I (39428) ESP32-CAM: Avg. Sobel time: 5.53 ms (1.4%)
I (39428) ESP32-CAM: Avg. save time: 4.39 ms (1.1%)
I (39428) ESP32-CAM: Estimated power consumption: 290.00 mW, 0.032060 mWh per frame
I (39438) ESP32-CAM: Performance per Watt: 8.66 FPS/W
I (39448) ESP32-CAM: Total energy for 10 days: 69599.99 Wh (18810808.00 mAh at 3.7V)
I (40448) ESP32-CAM: ===== SUMMARY =====
I (40448) ESP32-CAM: CPU Freq (MHz) | FPS | Power (mW) | FPS/Watt
I (40448) ESP32-CAM: 240 | 2.98 | 370.00 | 8.04
I (40448) ESP32-CAM: 160 | 2.79 | 330.00 | 8.46
I (40458) ESP32-CAM: 80 | 2.51 | 290.00 | 8.66 (OPTIMAL)
I (40458) ESP32-CAM: Setting CPU frequency to 240 MHz
I (40468) pm: Frequency switching config: CPU_MAX: 240, APB_MAX: 240, APB_MIN: 240, Light sleep: DISABLED
I (40478) ESP32-CAM: CPU frequency set successfully
I (40478) ESP32-CAM: All tests completed. Entering continuous capture mode...
I (40488) ESP32-CAM: Freed oldest image set (index 0)
I (40488) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240
I (40788) ESP32-CAM: Freed oldest image set (index 1)
I (40788) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240
I (41108) ESP32-CAM: Freed oldest image set (index 2)
I (41108) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240
I (41428) ESP32-CAM: Freed oldest image set (index 3)
I (41428) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240
I (41748) ESP32-CAM: Freed oldest image set (index 4)
I (41748) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240

⌘K to generate a command

0 0 Build - Ln 41, Col 45 Spaces: 4 UTF-8 LF Plain Text Go Live Cursor-Tab -

Problems 17 Output Debug Console Terminal Ports

zsh + - - -

I (23648) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240
I (24008) ESP32-CAM: Freed oldest image set (index 6)
I (24008) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240
I (24368) ESP32-CAM: Freed oldest image set (index 7)
I (24368) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240
I (24728) ESP32-CAM: Freed oldest image set (index 8)
I (24728) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240
I (25088) ESP32-CAM: Freed oldest image set (index 9)
I (25088) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240
I (25438) ESP32-CAM: Memory: Free DRAM: 2750163 bytes, Free PSRAM: 2511348 bytes
I (25448) ESP32-CAM: === Results for 160 MHz ===
I (25448) ESP32-CAM: Avg. FPS: 2.79
I (25448) ESP32-CAM: Avg. frame time: 348.18 ms
I (25448) ESP32-CAM: Avg. capture time: 0.33 ms (0.1%)
I (25448) ESP32-CAM: Avg. histogram time: 2.53 ms (0.7%)
I (25458) ESP32-CAM: Avg. Sobel time: 5.24 ms (1.5%)
I (25458) ESP32-CAM: Avg. save time: 3.85 ms (1.1%)
I (25458) ESP32-CAM: Estimated power consumption: 330.00 mW, 0.032816 mWh per frame
I (25468) ESP32-CAM: Performance per Watt: 8.46 FPS/W
I (25478) ESP32-CAM: Total energy for 10 days: 79200.01 Wh (21405408.00 mAh at 3.7V)
I (26478) ESP32-CAM: Setting CPU frequency to 80 MHz
I (26478) pm: Frequency switching config: CPU_MAX: 80, APB_MAX: 80, APB_MIN: 80, Light sleep: DISABLED
I (26478) ESP32-CAM: CPU frequency set successfully
I (27478) ESP32-CAM: Testing with CPU frequency: 80 MHz
I (27478) ESP32-CAM: Freed oldest image set (index 0)
I (27478) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240
I (27838) ESP32-CAM: Freed oldest image set (index 1)
I (27838) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240
I (28228) ESP32-CAM: Freed oldest image set (index 2)
I (28228) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240
I (28628) ESP32-CAM: Freed oldest image set (index 3)
I (28628) ESP32-CAM: Scaling down image for processing: 640x480 -> 320x240

⌘K to generate a command

0 0 Build - Ln 41, Col 45 Spaces: 4 UTF-8 LF Plain Text Go Live Cursor-Tab -