

دانشگاه صنعتی امیر کبیر (پلی تکنیک تهران)

معماری کامپیوتروریزپردازنده: پردازش تصویر

استاد درس:

دكتر سيدين

اعضای گروه:

مازيار دادور | 9723029 على مطلبي |9723086

فهرست مطالب

2	1 الگوريتم پياده شده
5	2 عملكرد كد
7	3 نتایج خروجی

الگوريتم پياده شده:

كرنل گوسى:

- دیتا های تصویر نویزی بعد از اعمال پدینگ را به صورت بایت های پشت سر هم در حافظه ذخیره میکنیم.
 - 2. آدرس شروع دیتا ها را در رجیستر r0 ذخیره میکنیم.
- 3. مقدار رجیستر r0 را با 18 جمع میکنیم و در r0 میریزیم، تا از دیتا های پد سطر اول و همچنین اولین دیتای سطر دوم که آن نیز دیتای پد است بگذریم و به آدرس اولین دیتای اصلی تصویر برسیم.
- آدرس شروع حافظه که میخواهیم پیکسل های فیلتر شده را از آن به بعد در آن ذخیره کنیم را در رجیستر
 r5 میریزیم.
 - 5. مقدار 15 را در رجیستر r1 به عنوان شمارنده سطر ها میریزیم.
 - 6. ليبل حلقه سطر ها را قرار ميدهيم.
 - 7. مقدار 15 را در رجیستر r2 به عنوان شمارنده ستون ها میریزیم.
 - اليبل حلقه ستون ها را قرار ميدهيم.
 - 9. در ابتدای حلقه ستون ها مقدار 0 را در رجیستر r_3 قرار میدهیم.(این رجیستر حاصل عملیات کرنل(پیکسل فیلتر شده) را در خود قرار میدهد.)
- 10. بایت موجود در آدرس موجود در رجیستر r0 (پیکسل مرکزی) با آفست r8- را در رجیستر r4 ذخیره میکنیم.(مقدار این بایت مطابق با در ایه r1 ماتریس کرنل است)
 - 11. از آنجایی که وزن این پیکسل در کرنل 1 است، بنابراین فقط با مقدار r_3 جمع شده و در r_3 ذخیره میشود.
- 12. بایت موجود در آدرس موجود در رجیستر ro (پیکسل مرکزی) با آفست 17- را در رجیستر r4 ذخیره میکنیم.(مقدار این بایت مطابق با درایه 12 ماتریس کرنل است)
- 13. از آنجایی که وزن این پیکسل در کرنل 2 است، بنابراین r4 را یک واحد به چپ شیفت میدهیم سپس با مقدار r3 جمع شده و در r3 ذخیره میشود.
- 14. بایت موجود در آدرس موجود در رجیستر r0 (پیکسل مرکزی) با آفست 16- را در رجیستر r4 ذخیره میکنیم.(مقدار این بایت مطابق با در ایه r4 ماتریس کرنل است)
 - 15. از آنجایی که وزن این پیکسل در کرنل 1 است، بنابراین فقط با مقدار 73 جمع شده و در 73 ذخیره میشود.
- 16. بایت موجود در آدرس موجود در رجیستر r0 (پیکسل مرکزی) با آفست 1- را در رجیستر r4 ذخیره میکنیم.(مقدار این بایت مطابق با درایه 21 ماتریس کرنل است)
- 17. از آنجایی که وزن این پیکسل در کرنل 2 است، بنابراین ۲4 را یک واحد به چپ شیفت میدهیم سپس با مقدار r3 جمع شده و در r3 ذخیره میشود.
- 18. بایت موجود در آدرس موجود در رجیستر r0 (پیکسل مرکزی) را در رجیستر r4 ذخیره میکنیم.(مقدار این بایت مطابق با در ایه 22 ماتریس کرنل است)
- 19. از آنجایی که وزن این پیکسل در کرنل 4 است، بنابر این ۲4 را دو واحد به چپ شیفت میدهیم سپس با مقدار r3 جمع شده و در r3 ذخیره میشود.
- 20. بایت موجود در آدرس موجود در رجیستر r0 (پیکسل مرکزی) با آفست 1+ را در رجیستر r4 ذخیره میکنیم.(مقدار این بایت مطابق با در ایه r4 ماتریس کرنل است)
- 21. از آنجایی که وزن این پیکسل در کرنل 2 است، بنابراین r4 را یک واحد به چپ شیفت میدهیم سپس با مقدار r3 جمع شده و در r3 ذخیره میشود.
- دخیره r4 دخیره ایت موجود در آدرس موجود در رجیستر r0 (پیکسل مرکزی) با آفست r4 را در رجیستر r4 دخیره میکنیم.(مقدار این بایت مطابق با در ایه r4 ماتریس کرنل است)
 - دخیره r3 جمع شده و در r3 ذخیره r3 است، بنابراین فقط با مقدار r3 جمع شده و در r3 ذخیره میشو د.

- 24. بایت موجود در آدرس موجود در رجیستر r0 (پیکسل مرکزی) با آفست r+1 را در رجیستر r+1 ذخیره میکنیم.(مقدار این بایت مطابق با در ایه 32 ماتریس کرنل است)
- 25. از آنجایی که وزن این پیکسل در کرنل 2 است، بنابراین r4 را یک واحد به چپ شیفت میدهیم سپس با مقدار r3 جمع شده و در r3 ذخیره میشود.
- 26. بایت موجود در آدرس موجود در رجیستر r0 (پیکسل مرکزی) با آفست r4 را در رجیستر r4 ذخیره میکنیم.(مقدار این بایت مطابق با در ایه r4 ماتریس کرنل است)
 - دخیره r3 از آنجایی که وزن این پیکسل در کرنل 1 است، بنابراین فقط با مقدار r3 جمع شده و در r3 ذخیره میشود.
 - 28. مقدار رجیستر r3 (رجیستر حاصل اعمال کرنل) را بر مقدار 16 تقسیم میکنیم.
 - 29. حاصل رجیستر r3 را در آدرس موجود در رجیستر r5 (آدرس ذخیره پیکسل فیلتر شده در حافظه)
 - 30. رجیستر ۲5 (آدرس ذخیره پیکسل فیلتر شده در حافظه) را یکی اضافه میکنیم.
 - 31. رجیستر r0 را یکی اضافه میکنیم. (برای رفتن به پیکسل نویزی بعدی)
 - 32. رجیستر r2 (شمارنده ستون) را یکی کم میکنیم.
 - 33. رجیستر r2 (شمارنده ستون) را با مقدار 0 مقایسه میکنیم.
 - 34. اگر برابر نبود به 9 بازمیگردیم.
 - 35. به آخر سطر رسیده ایم بنابر این رجیستر ro را دوتا اضافه میکنیم تا به آدرس پیکسل اصلی در سطر بعد، پس از طی کردن پیکسل های پد شویم.
 - 36. رجیستر r1 (شمارنده سطر) را یکی کم میکنیم.
 - 37. رجیستر r1 (شمارنده سطر) را با مقدار 0 مقایسه میکنیم.
 - 38. اگر برابر نبود به 7 بازمیگردیم.
 - 39. ڀايان

كرنل تشخيص لبه:

- 1. دیتا های تصویر اصلی بعد از اعمال پدینگ را به صورت بایت های پشت سر هم در حافظه ذخیره میکنیم.
 - 2. آدرس شروع دیتا ها را در رجیستر r0 ذخیره میکنیم.
- 3. مقدار رجیستر r0 را با 18 جمع میکنیم و در r0 میریزیم، تا از دیتا های پد سطر اول و همچنین اولین دیتای سطر دوم که آن نیز دیتای پد است بگذریم و به آدرس اولین دیتای اصلی تصویر برسیم.
 - 4. آدرس شروع حافظه که پیکسل های فیلتر شده را از آن به بعد در آن ذخیره کنیم را در رجیستر r5 میریزیم.
 - 5. مقدار 15 را در رجیستر r1 به عنوان شمارنده سطر ها میریزیم.
 - 6. ليبل حلقه سطر ها را قرار ميدهيم.
 - 7. مقدار 15 را در رجیستر r2 به عنوان شمارنده ستون ها میریزیم.
 - 8. ليبل حلقه ستون ها را قرار ميدهيم.
 - 9. در ابتدای حلقه ستون ها مقدار 0 را در رجیستر r3 قرار میدهیم.(این رجیستر حاصل عملیات کرنل(پیکسل فیلتر شده) را در خود قرار میدهد.)
- 10. بایت مُوجود در آدرس موجود در رجیستر r0 (پیکسل مرکزی) با آفست r1- را در رجیستر r4 ذخیره میکنیم.(مقدار این بایت مطابق با در ایه r1 ماتریس کرنل است)
 - 11. از آنجایی که وزن این پیکسل در کرنل 1 است، بنابراین فقط با مقدار r_3 جمع شده و در r_3 ذخیره میشود.
- 12. بایت موجود در آدرس موجود در رجیستر r0 (پیکسل مرکزی) با آفست 1- را در رجیستر r4 ذخیره میکنیم.(مقدار این بایت مطابق با در ایه r4 ماتریس کرنل است)
 - دخیره r3 از آنجایی که وزن این پیکسل در کرنل 1 است، بنابراین فقط با مقدار r3 جمع شده و در r3 ذخیره میشود.

- 14. بایت موجود در آدرس موجود در رجیستر r0 (پیکسل مرکزی) را در رجیستر r4 ذخیره میکنیم.(مقدار این بایت مطابق با در ایه 22 ماتریس کرنل است)
 - 15. از آنجایی که وزن این پیکسل در کرنل 4- است، بنابراین ابتدا ۲4 را دو واحد به چپ شیفت میدهیم.
 - 16. رجیستر r4 را قرینه میکنیم سپس با مقدار 1 جمع میکنیم تا متمم 2 آن بدست آید (قرینه شود)
 - 17. رجیستر r4 با مقدار رجیستر r3 جمع شده و در r3 ذخیره میشود.
- 18. بایت موجود در آدرس موجود در رجیستر r0 (پیکسل مرکزی) با آفست 1+ را در رجیستر r4 ذخیره میکنیم.(مقدار این بایت مطابق با در ایه r4 ماتریس کرنل است)
 - 19. از آنجایی که وزن این پیکسل در کرنل 1 است، بنابراین فقط با مقدار ۲3 جمع شده و در ۲3 ذخیره میشود.
- 20. بایت موجود در آدرس موجود در رجیستر r0 (پیکسل مرکزی) با آفست r+1 را در رجیستر r+1 ذخیره میکنیم.(مقدار این بایت مطابق با در ایه 32 ماتریس کرنل است)
 - 21. از آنجایی که وزن این پیکسل در کرنل 1 است، بنابراین فقط با مقدار ۲3 جمع شده و در ۲3 ذخیره میشود.
 - 22. مقدار رجیستر r3 (رجیستر حاصل اعمال کرنل) را بر مقدار 4 تقسیم میکنیم.
 - 23. مقدار رجيستر r3 (رجيستر حاصل اعمال كرنل) را با مقدار 64 جمع ميكنيم.
- 24. مقدار رجیستر r3 را با عدد 100 مقایسه می کنیم، اگر کوچکتر یا مساوی بود مقدار آن را 0 میکنیم در غیر اینصورت مقدار آن را حداکثر مقدار یک بایت یعنی 255 میکنیم.
 - 25. حاصل رجیستر r3 را در آدرس موجود در رجیستر r5 (آدرس ذخیره پیکسل فیلتر شده در حافظه)
 - 26. رجیستر r5 (آدرس ذخیره بیکسل فیلتر شده در حافظه) را یکی اضافه میکنیم.
 - 27. رجیستر r0 را یکی اضافه میکنیم. (برای رفتن به پیکسل بعدی)
 - 28. رجیستر r2 (شمارنده ستون) را یکی کم میکنیم.
 - 29. رجیستر r2 (شمارنده ستون) را با مقدار 0 مقایسه میکنیم.
 - 30. اگر برابر نبود به 9 بازمیگردیم.
 - 31. به آخر سطر رسیده ایم بنابراین رجیستر ro را دوتا اضافه میکنیم تا به آدرس پیکسل اصلی در سطر بعد، پس از طی کردن پیکسل های پد شویم.
 - 32. رجیستر r1 (شمارنده سطر) را یکی کم میکنیم.
 - 33. رجیستر r1 (شمارنده سطر) را با مقدار 0 مقایسه میکنیم.
 - 34. اگر برابر نبود به 7 بازمیگردیم.
 - 35. پایان

عملکرد کد:

در کد اسمبلی نوشته شده دو بخش موجود است:

بخش مربوط به تنظیم رجیستر های پردازنده stm32f407VGT برای تنظیم برخی از پورت های GPIO و همچنین بخش مربوط به عملیات محاسباتی فیلترینگ و اعمال کرنل ها.

بخش مربوط به محاسبات فیلترینگ به تفصیل در قسمت قبلی توضیح داده شد.

حال به بررسی تنظیمات رجیستری میپردازیم.

همانطور که در صورت پروژه خواسته شده است، با فعال شدن پین 0 پورت A عملیات فیلترینگ شروع میشود و با پایان یافتن عملیات پین 1 پورت B را فعال میکنیم.

بنابراین باید تنظیمات رجیستری برای پورت ها و پین های مربوطه انجام دهیم. در تمام خطوط کد سعی شده که با کامنت گذاری عملکر د کد روشن باشد.

ابتدا در قطعه كد زير ، كلاك باس مربوط به پورت هاى A,B را فعال ميكنيم.

سپس رجیستر های MODER, OSPEEDR, PUPDR در PORTA را برای تنظیم کردن PINO آن تنظیم میکنیم.

```
;****** Configuring GPIOA.PINO as input 2MHz Without Pullup/Pulldown ********
       rO, #GPIOA MODER ; rO <- GPIOA MODER Register Address
MOV32
       r1, [r0]
                          ; r1 <- GPIOA MODER Register Value
T.DR
       r2, #0x00000003
MOV32
                          ; r2 <- 0b0000000000000000000000000000000011
       r2, r2;
                          ; r2 <- ~r2
AND
       r1, r1, r2
                          ; r1 <- r1 AND r2 (GPIOA MODER & ~0b11)
STR
       r1, [r0]
                          ; GPIOA MODER Register <- r1
       r0, #GPIOA OSPEEDR
MOV32
                          ; r0 <- GPIOA OSPEEDR Register Address
                         ; r1 <- GPIOA OSPEEDR Register Value
T<sub>1</sub>DR
       r1, [r0]
       r2, #0x00000003
                          MOV32
MVN
       r2, r2;
                          ; r2 <- ~r2
       r1, r1, r2
                          ; r1 <- r1 AND r2 (GPIOA OSPEEDR & ~0b11)
AND
       r1, [r0]
                          ; GPIOA OSPEEDR Register <- r1
STR
MOV32
       rO, #GPIOA PUPDR ; rO <- GPIOA PUPDR Register Address
                         ; r1 <- GPIOA PUPDR Register Value
       r1, [r0]
       r2, #0x00000003
MOV32
                         ; r2 <- 0b0000000000000000000000000000000011
MVN
       r2, r2;
                          ; r2 <- ~r2
AND
       r1, r1, r2
                          ; r1 <- r1 AND r2 (GPIOA PUPDR & ~0b11)
STR
                          ; GPIOA PUPDR Register <- rl
       r1, [r0]
• * * * * * *
```

به همین ترتیب سپس رجیستر های MODER, OTYPER, OSPEEDR, PUPDR در PORTB را برای تنظیم کردن PIN1 آن تنظیم میکنیم.

```
;****** Configuring GPIOB.PORT1 as output 2MHz Push Pull Without Pullup/Pulldown *******
MOV32 r0, #GPIOB MODER ; r0 <- GPIOB MODER Register Address
T-DR
      r1, [r0]
                       ; r1 <- GPIOB MODER Register Value
      r2, #0x00000004
                       MOV32
      r1, r1, r2
                       ; r1 <- r1 OR r2 (GPIOB MODER | 0b100)
      r1, [r0]
                       ; GPIOB MODER Register <- r1
MOV32
      r0, #GPIOB OTYPER ; r0 <- GPIOB OTYPER Register Address
                     ; r1 <- GPIOB_OTYPER Register Value
LDR
      r1, [r0]
      r2, #0x00000002
MOV32
                       r2, r2;
MVN
                       ; r2 <- ~r2
                       ; r1 <- r1 AND r2 (GPIOB OTYPER & ~0b10)
AND
      r1, r1, r2
      r1, [r0]
STR
                       ; GPIOB OTYPER Register <- r1
MOV32
      r0, #GPIOB_OSPEEDR ; r0 <- GPIOB_OSPEEDR Register Address
                     ; r1 <- GPIOB OSPEEDR Register Value
LDR
      r1, [r0]
                       r2, #0x0000000C
MOV32
MVN
      r2, r2;
                       ; r2 <- ~r2
                       ; r1 <- r1 AND r2 (GPIOB_OSPEEDR & ~0b1100)
      r1, r1, r2
AND
STR
      r1, [r0]
                       ; GPIOB OSPEEDR Register <- r1
MOV32
      rO, #GPIOB_PUPDR ; rO <- GPIOB_PUPDR Register Address
                     ; r1 <- GPIOB PUPDR Register Value
      r1, [r0]
LDR
MOV32
      r2, #0x0000000C
                       MVN
      r2, r2;
                       ; r2 <- ~r2
AND
      r1, r1, r2
                       ; r1 <- r1 AND r2 (GPIOB PUPDR & ~0b1100)
      r1, [r0]
                       ; GPIOB PUPDR Register <- r1
STR
;******
```

A پورت A استفاده از قطعه کد زیر بوسلیه یک حلقه منتظر فعال شدن پایه A پورت A میمانیم.

```
;***** Waiting for PORTA.0 to Activate **************
PA0Loop
        MOV32
             r0, #GPIOA IDR
                            ; r0 <- GPIOA IDR Register Address
              r1, [r0]
                             ; r1 <- GPIOA IDR Register Value
              r2, #0x00000001
        MOV32
                             AND
              r1, r1, r2
                             ; r1 <- r1 AND r2 (GPIOA IDR & 0b10)
        CMP
              r1, #1
              PA0Loop
```

(نکته: از آنجایی که برای بررسی کد نوشته شده از شبیه سازی (simulator) استفاده می شود، چون پین 0 پورت A ورودی تعیین شده است و باید بصورت خارجی (با استفاده از کلید یا روشی دیگر) آن را فعال کنیم (رجیستر ReadOnly IDR) است و تنها با اعمال خارجی بر پین مربوطه مقدار میگیرد). در شبیه سازی راهی برای تست آن وجود ندارد. بنابراین در شبیه سازی راهی برای خروج از حلقه بالاوجود ندارد بنابراین باید کامنت شود.)

پس از آن به بدنه اصلی کد یعنی محاسبات کرنل ها میرسیم که توضیحات کامل آنها در قسمت قبل آمده است.

و پس از اتمام عملیات مربوط به کرنل ها پین 1 پورت B را فعال میکنیم.

نتایج خروجی:

نتایج خروجی شامل تغییرات رجیستر های GPIOA, GPIOB و AHB1ENR مربوط به بخش SRAM منبیرات رجیستر های system viewer و در بخش system viewer و همچنین پیکسل های خروجی عملیات فیلتیرینگ پس از ذخیره در حافظه 3x20000000 از آدرس شروع 3x20000000 به بعد یعنی آدرس 3x20000000000000 حافظه است.

با اجرای کد و رفتن به قسمت system viewer میتوان رجیستر ها را مشاهده کرد و برای مشاهده حافظه با رفتن به قسمت memory window و تایپ آدر س حافظه 0x200000000 اطلاعات ذخیره شده را نیز مشاهده کرد.

در ادامه تصویر بلوک حافظه که در آن پیکسل های فیلتر شده را قرار داده ایم و همچنین پیکسل های مربوطه در شبیه سازی متلب که اسکریپت های آن موجود است را قرار داده و علاوه بر آن مقایسه ای در نتایج خواهیم داشت.

نتیجه کرنل فیلتر گاوسی در حافظه پردازنده:

```
Memory 1
 Address: 0x20000000
0x20000000: 77 7B 89 85 7E 87 71 4C 62 8A 8D 80 87 8D 71
0x2000000F: 69 74 81 7D 71 6F 5C 3F 54 6F 6E 70 80 8F 87
0x2000001E: 60 6B 76 77 6A 5F 52 40 51 64 5D 5E 6D 7D 80
0x2000002D: 62 67 6A 69 63 5E 57 46 4E 5F 5D 5D 68 71 6F
0x2000003C: 72 63 5A 60 65 66 59 41 45 53 5B 69 73 6C 63
0x2000004B: 8D 6E 61 6E 79 77 59 3C 4C 5C 62 71 71 62 5E
0x2000005A: 78 68 5E 63 6B 65 4A 38 4A 56 55 56 51 51 59
0x20000069: 51 4F 4B 4D 4C 44 37 30 3E 43 37 31 31 40 55
0x20000078: 5C 56 5B 62 5D 59 43 31 47 50 3D 37 3D 55 6F
0x20000087: 6D 65 6B 6A 6E 74 53 35 4F 5D 52 54 5B 6B 7D
0x20000096: 61 65 6F 65 6D 79 5B 41 5B 6D 62 5E 63 68 6A
0x200000A5: 58 5B 6B 6A 6D 76 62 4C 65 78 66 55 58 60 60
0x200000B4: 6F 62 65 6B 6D 70 5F 49 5A 71 69 57 51 5B 67
0x200000C3: 87 80 7A 77 7C 7A 5F 44 56 75 73 60 52 57 6E
0x200000D2: 88 8F 91 8D 8F 87 61 42 59 71 69 5F 50 53 6D
```

نتیجه شبیه سازی در متلب:

```
Command Window
  >> format hex
  >> denoised Img
  denoised_Img =
    15×15 uint8 matrix
    77
         7c
              8a
                  86
                       7 f
                            87
                                 71
                                                    8d
                                                         81
                                                              87
                                                                  8d
                                                                       71
                                      4c
                                           62
                                               8a
     69
         74
              81
                   7d
                       71
                            70
                                 5c
                                      40
                                           55
                                               6f
                                                    6e
                                                         70
                                                              80
                                                                  90
                                                                       87
     60
         6c
              76
                   77
                        ба
                            5f
                                 53
                                      40
                                           52
                                                    5e
                                                         5f
                                                              6d
                                                                  7e
                                                                       81
     62
         67
              6a
                   6a
                        64
                            5f
                                 57
                                      46
                                           4e
                                               5f
                                                    5d
                                                         5d
                                                              68
                                                                  71
                                                                       6f
    73
         63
              5b
                   61
                        65
                            66
                                 59
                                      41
                                           45
                                               54
                                                    5b
                                                              73
                                                                  6d
                                                                       64
     8d
              61
                        79
                            77
                                      3d
                                           4d
                                                    62
                                                         72
                                                              72
                                                                       5f
                                 5a
    78
         68
                   63
                        6b
                                      38
                                           4b
                                                    55
                                                              52
              5e
                                 4a
    51
         4f
              4c
                   4d
                       4c
                            45
                                 37
                                      30
                                           3f
                                               43
                                                    38
                                                         32
                                                              32
                                                                  41
                                                                       56
         57
              5c
                   62
                            59
                                 43
                                      31
                                           48
                                               50
                                                    3e
                                                         37
                                                              3e
                                                                  56
                                                                       70
    5c
                       5e
     6d
         66
              6b
                   6b
                       6f
                            75
                                53
                                     36
                                           4f
                                               5d
                                                    53
                                                        55
                                                              5b
                                                                  6с
                                                                       7d
                                                        5f
     62
         66
              6f
                  66
                       6e
                            7a
                                5b
                                     41
                                          5b
                                               6d
                                                    62
                                                              63
                                                                  68
                                                                       6a
                           77
                                 62
                                              78
                                                       56
    59
                                                    67
                                                             58
                                                                  61
                                                                       61
         5c
              6c 6b
                      6d
                                     4d
                                          65
                                                    6a 58
     6f
        63
            66 6b 6d 71
                               5f 49
                                              71
                                                             52 5b
                                                                       68
                                          5a
                 77
                      7c 7b 5f 44
                                         57
                                              75 73 61 52 58
     88
        81
             7a
                                                                       6e
     88
        90 91
                 8d
                      90 88 62 43 59
                                              71
                                                    6a 5f 51
                                                                 53
                                                                       6d
```

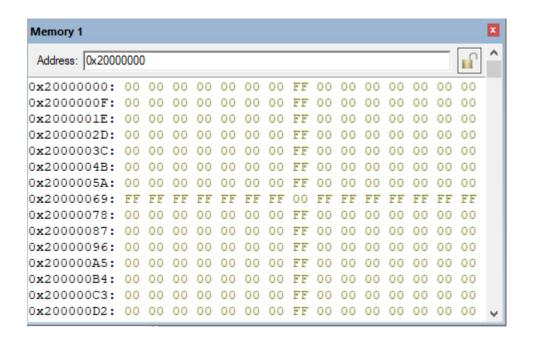
نتیجه کرنل تشخیص لبه در حافظه پردازنده بیش از اعمال حد آستانه:

```
Memory 1
                                                       Address: 0x20000000
0x20000000: 3B 3D 41 42 38 41 22 7D 1F 3E 3C 3E 42 40 48
0x2000000F: 40 54 23 36 5C 3B 25 66 28 56 3E 41 41 27 3B
0x2000001E: 40 47 4D 44 1B 58 1C 7C 20 39 48 3D 37 50 3E
0x2000002D: 3F 3C 2A 4A 5A 3F 19 7C 1E 38 3F 59 2D 4D 40
0x2000003C: 45 36 55 3A 41 45 1E 7F 15 59 44 41 4F 29 3D
0x2000004B: 32 60 5C 37 40 38 2A 7B 16 47 57 2E 41 61 47
0x2000005A: 21 1A 15 23 1C 1E 08 81 0B 1E 20 27 1C 19 1E
0x20000069: 7A 6D 7B 72 81 7D 80 22 80 77 70 69 6F 74 72
0x20000078: 24 20 1A 1F 12 23 0A 7F 03 17 34 32 23 3A 1E
0x20000087: 37 56 4F 39 61 45 17 7E 1D 4B 4F 47 38 37 40
0x20000096: 3C 34 42 47 34 58 1A 77 26 41 37 3A 4A 39 45
0x200000A5: 38 53 24 62 28 33 2A 69 2C 3B 3D 37 4F 3A 43
0x200000B4: 3B 44 5E 26 5A 4C 20 65 2C 3B 3D 4E 42 48 37
0x200000C3: 3E 4B 3D 3D 46 36 27 76 2B 54 2D 40 34 58 37
0x200000D2: 3E 3B 3F 33 3F 41 21 7D 1C 42 4B 33 50 39 3B
```

نتیجه شبیه سازی کرنل تشخیص لبه پیش از اعمال حد آستانه در متلب:

```
Command Window
  >> format hex
  >> edge_Img1
  edge_Img1 =
    15×15 <u>uint8</u> matrix
                                     21
                                          7d
                                               1f
                                                               3e
                                                                     42
                                                                          40
                                                                                48
          3d
               41
                   43
                          38
                               42
                                                     3e
                                                          3c
     3f
          55
               23
                     35
                                     25
                                          67
                                                28
                                                     56
                                                                41
                                                                     41
                                                                          27
                          5c
                               3a
                                                          3e
     40
          47
                    44
                               59
                                          7d
                                                1f
               4 e
                          1b
                                     1c
                                                     39
                                                          49
                                                                3d
                                                                     36
                                                                          50
                               3f
                                          7d
     3f
          3c
               2 a
                     4b
                          5b
                                     19
                                                1e
                                                     37
                                                          3f
                                                               59
                                                                     2d
                                                                          4e
     46
          36
               55
                     3a
                          42
                                45
                                     1e
                                          80
                                                15
                                                     59
                                                          45
                                                                42
                                                                     4f
                                                                          29
                                                                                3c
     31
          60
                5c
                     37
                          41
                                38
                                     2a
                                          7b
                                                16
                                                     48
                                                          58
                                                                2e
                                                                     41
                                                                          62
     21
          1a
               15
                     23
                          1c
                               1e
                                     08
                                          81
                                                0b
                                                     1e
                                                          20
                                                               26
                                                                     1b
                                                                          19
     7b
                7c
                     72
                          81
                                7e
                                     80
                                          22
                                                81
                                                     77
                                                          71
                                                                69
                                                                     6f
                                                                          74
                                                                                72
          20
                          12
                                23
                                     0a
                                          7f
                                                03
                                                     17
                                                          33
                                                                32
                                                                     23
                                                                          3a
     37
          57
                50
                     39
                          61
                                46
                                     16
                                                1d
                                                     4b
                                                          50
                                                                47
                                                                     38
                                                                          37
                                          7e
                42
                     47
                          33
                                58
                                     19
                                          78
                                                26
                                                     42
                                                          37
                                                                     4b
                                                                          39
                                                                3a
               24
                          27
                                32
                                                                37
                                                                     50
                                          6a
                                                     3b
                                                          3c
               5e
                               4d
                                          66
                                                2c
                                                     3b
                                                          3d
                                                               4f
                                                                     42
          4b
               3d
                     3d
                          47
                                36
                                          76
                                                2b
                                                     54
                                                          2d
                                                                40
                                                                     34
                               41
                                     21
                                          7e
                                               1c
                                                     42
                                                          4c
                                                               32
                                                                     50
                                                                          38
```

نتیجه کرنل تشخیص لبه در حافظه پردازنده پس از اعمال حد آستانه:



نتیجه شبیه سازی کرنل تشخیص لبه پس از اعمال حد آستانه در متلب:

```
>> format hex
   >> edge Img
   edge Img =
     15×15 <u>uint8</u> matrix
      00
            00
                  0.0
                       00
                              0.0
                                    00
                                          00
                                               ff
                                                      00
                                                           00
                                                                  00
                                                                       00
                                                                              00
                                                                                   0.0
                                                                                         00
      0.0
            0.0
                        0.0
                              0.0
                                          0.0
                                               ff
                                                      0.0
                                                           0.0
                                                                       0.0
                                                                              0.0
                                                                                   0.0
                                                                                         0.0
                  0.0
                                    00
                                                                  0.0
      0.0
            0.0
                  0.0
                        0.0
                              00
                                    0.0
                                          0.0
                                                ff
                                                      0.0
                                                            0.0
                                                                  0.0
                                                                       0.0
                                                                              0.0
                                                                                   0.0
                                                                                         00
      00
            00
                  00
                        00
                              00
                                    00
                                          00
                                                      00
                                                            00
                                                                  00
                                                                       00
                                                                              00
                                                                                         00
                                                ff
            00
                              00
                                                                  00
      00
                        00
                                    00
                                          00
                                                            00
                                                                       00
      00
            00
                  00
                        00
                              00
                                    00
                                          00
                                                ff
                                                      00
                                                           00
                                                                  00
                                                                       00
                                                                              00
                                                                                   00
                                                                                         00
      00
            00
                  00
                        0.0
                              00
                                    00
                                          0.0
                                                ff
                                                      00
                                                           00
                                                                  00
                                                                       00
                                                                              00
                                                                                   0.0
                                                                                         00
            ff
                                    ff
      ff
                  ff
                        ff
                              ff
                                          ff
                                                0.0
                                                      ff
                                                            ff
                                                                  ff
                                                                       ff
                                                                              ff
                                                                                   ff
                                                                                          ff
      00
            00
                  00
                        00
                              00
                                    00
                                          00
                                                ff
                                                      00
                                                            00
                                                                  00
                                                                       00
                                                                              00
                                                                                   00
                                                                                         00
      00
            00
                  00
                        00
                              00
                                    00
                                          00
                                                ff
                                                      00
                                                            00
                                                                  00
                                                                        00
                                                                              00
      00
            00
                  00
                        00
                              00
                                    00
                                          00
                                                ff
                                                            00
                                                                  00
                                                                        00
                                                                              00
                                                                                         00
                                                      00
      00
            00
                  00
                        00
                              00
                                    00
                                          00
                                                ff
                                                      00
                                                            00
                                                                  00
                                                                       00
                                                                              00
                                                                                   00
                                                                                         00
      0.0
            0.0
                  00
                        00
                              00
                                    00
                                          00
                                                ff
                                                      00
                                                           00
                                                                  00
                                                                       00
                                                                              00
                                                                                   00
                                                                                         0.0
      0.0
            0.0
                       0.0
                              0.0
                                    0.0
                                         0.0
                                               ff
                                                     0.0
                                                           0.0
                                                                  0.0
                                                                       0.0
                                                                             0.0
                                                                                   0.0
                                                                                         0.0
                  0.0
      00
                              00
                                    00
                                                           00
                                                                  00
fx >>
```

تصاویر مربوط به شبیه سازی در متلب با استفاده از نتایج اجرای اسکرییت های آماده بدست آمده اند.

همچنین برای مقایسه بهتر با مقادیر ذخیره شده در حافظه حاصل کد اسمبلی format در متلب را بصورت hex تبدیل کرده ایم.

نکته: با مقایسه مقادیر شبیه سازی متلب و مقادیر بدست آمده در حافظه پردازنده متوجه میشویم که برخی مقادیر به اندازه 1 واحد با یکدیگر اختلاف دارند. دلیل این موضوع آن است که در دستور العمل تقسیم در پردازنده بخش اعشاری عدد به کلی حذف میشود. اما در متلب پس از تقسیم عدد بسته به مقدار اعشارش گرد میشود. (اگر اعشار بزرگتر از 0.5 باشد به یک واحد بزرگتر گرد شده در غیر اینصورت جزء صحیح عدد نتیجه میشود.)