**به نام خدا**

**گزارش سوال چهارم عملی درس سیستم های نهفته**

**پارسا آقاعلی 400521072**

**هدف تمرین**

هدف این برنامه طراحی یک سیستم است که مقادیر یک حسگر حافظه‌نگاشت شده (Memory-Mapped Sensor) را در طی ۱۰ چرخه با فاصله زمانی ۱۰ میلی‌ثانیه بخواند، این مقادیر را ذخیره کند، میانگین آن‌ها را با دقت دو رقم اعشار محاسبه کند، تمام مقادیر و میانگین را چاپ کند و در نهایت اجرای برنامه متوقف شود.

**توضیحات فنی برنامه**

**حسگر حافظه‌نگاشت شده**

* حسگرهای حافظه‌نگاشت شده به صورت مستقیم با استفاده از یک آدرس خاص در حافظه سیستم قابل دسترسی هستند.
* در این برنامه از آدرس فرضی 0x3FF44000 برای شبیه‌سازی حسگر استفاده شده است.
* از کلمه کلیدی volatile استفاده شده تا اطمینان حاصل شود که کامپایلر مقدار این آدرس را بهینه‌سازی نکند، زیرا مقدار آن ممکن است در زمان اجرا تغییر کند.

**چرخه نمونه‌برداری**

* برنامه مقدار حسگر را در بازه‌های ۱۰ میلی‌ثانیه به مدت ۱۰ چرخه می‌خواند و سپس با توجه به زمان دلخواهی که در حلقه while آخر ایجاد کرده ایم می توانیم دوباره این مراحل را تکرار کنیم.
* پس از اتمام ۱۰ چرخه:
  + تمام مقادیر ثبت‌شده در هر چرخه چاپ می‌شوند.
  + میانگین مقادیر محاسبه و با دقت دو رقم اعشار چاپ می‌شود

**شرح بخش‌های اصلی کد**

**۱. راه‌اندازی (Setup)**

* ارتباط سریال با سرعت 115200 آغاز می‌شود تا نتایج از طریق Serial Monitor قابل مشاهده باشد

  Serial.begin(115200); // Start serial communication

مقدار اولیه حسگر برای شبیه‌سازی روی 100 تنظیم می‌شود:

  \*sensorRegister = 100; // Initial value for simulation

**۲. ذخیره و پردازش مقادیر**

* از آرایه‌ای به نام values برای ذخیره مقادیر حسگر در هر چرخه استفاده شده است:

  uint32\_t values[numCycles]; // Array to store the 10 values

در هر چرخه:

* مقدار حسگر از آدرس حافظه خوانده می‌شود و در آرایه ذخیره می‌گردد:

   uint32\_t value = \*sensorRegister; // Read the sensor value

  values[i] = value; // Store the value in the array

مقدار حسگر به متغیر sum اضافه می‌شود تا برای محاسبه میانگین استفاده شود:

    sum += value; // Add to the sum

برای شبیه‌سازی تغییرات مقدار حسگر، از تابع random(90, 110) استفاده شده است:

    \*sensorRegister = random(90, 110);

**۳. محاسبه میانگین**

* میانگین مقادیر با استفاده از تقسیم مجموع (sum) بر تعداد چرخه‌ها محاسبه می‌شود:

  float average = sum / (float)numCycles;

* این محاسبه از نوع float استفاده می‌کند تا دقت اعشاری حفظ شود.

**۴. نمایش نتایج**

* تمام مقادیر ذخیره‌شده چاپ می‌شوند:

  for (int i = 0; i < numCycles; i++) {

    Serial.println("Value " + String(i + 1) + ": " + String(values[i]));

  }

میانگین مقادیر با دقت دو رقم اعشار چاپ می‌شود:

  Serial.println(average, 2); // Print with 2 decimal points

**۵. توقف اجرای برنامه**

* پس از چاپ نتایج، برنامه وارد یک حلقه while می‌شود که با توجه به مقدار بولین x و زمان تاخیر دلخواه ما می توان آن را برای هر مدت که میخواهیم اجرای آن را متوقف کنیم:

  while (x) {

    delay(6000);

    x=false;

  }

**خروجی مورد انتظار**

برای هر چرخه مقدار حسگر چاپ می‌شود:

All Sensor Values:

Value 1: 97

Value 2: 93

Value 3: 95

Value 4: 92

Value 5: 100

Value 6: 98

Value 7: 107

Value 8: 92

Value 9: 100

Value 10: 99

Average Sensor Value: 97.3

برنامه به صورت صحیح مقادیر را ذخیره، میانگین را محاسبه و نمایش می‌دهد و سپس متوقف می‌شود.

**منابع:**

<https://forum.arduino.cc/t/obtaining-the-average-of-10-readings/537976>

<https://stackoverflow.com/questions/8453239/getting-an-average-from-values-obtained-by-a-sensor-using-c>

<https://community.st.com/t5/stm32-mcus-products/memory-mapped-qspi-reads-nonsensical-data/td-p/274097>