# TI-RSLK MAX

Texas Instruments Robotics System Learning Kit

# Module 3

Activity: ARM Cortex M Architecture

# Activity: ARM Cortex M Architecture

## Question 1
Write an assembly function that returns true (R0=1) if an ASCII character is a letter, and false (R0=0) otherwise. The letters exist from 0x41 to 0x5A and from 0x61 to 0x7A inclusive. The input character is passed as a value in R0, and the return parameter is returned in R0.

## Question 2
Write an assembly function to calculate the average of three numbers. Assume the three numbers are passed by value into your function. The input parameters are passed as a value in R0, R1 and R2. The return parameter is returned in R0.

## Question 3
Write an assembly function to find the maximum of three numbers. Assume the three numbers are passed by value into your function. The input parameters are passed as a value in R0, R1 and R2. The return parameter is returned in R0.

## Question 4
Write an assembly function to calculate the quadratic equation

$$y = 2x^2 - 3x + 1$$

assuming x and y are 32-bit numbers. Some values of x will cause the calculation of y to extend beyond the values allowed by 32-bit signed numbers. Determine the largest possible value for x, such that $y < 2^{31}$. Use this threshold to return $y = 0x7FFFFFF$ ($2^{31}-1$) if the input value would create overflow. Determine the smallest possible value for x, such that $y > -2^{31}$. Use this threshold to return $y = 0x80000000$ ($-2^{31}$) if the input value would create underflow. The input parameter, x, is passed as a value in R0, and the return parameter, y, is returned in R0.

## Question 5
Write an assembly function that calculates the square distance between two points (x1, y1) and (x2, y2)

$$d = (x1-x2)^2 + (y1-y2)^2$$

assuming x1 x2, y1, and y2 are signed 32-bit numbers. You may assume the numbers are small enough that overflow does not occur. The input parameters are passed as a value in x1=R0, y1=R1, x2=R2 and y2=R3. The return parameter is returned in R0.

## Question 6
Write an assembly function that returns true (R0=1) if 10≤x<99, and false (R0=0) otherwise. The input parameter, x, is passed as a value in R0, and the return parameter is returned in R0.

## Question 7
Redo Question 5 using a subfunction. Basically, create a subfunction that takes on input, n, in R0, and returns the square of that input in $R0=n^2$. Then use this subfunction to solve the square distance function. The key issue here is managing AAPCS and the LR register.

## Question 8
Redo Question 4 using floating point arithmetic. Assume the input parameter is passed in register S0 and the output parameter is returned in S0.

2

# ti.com/rslk

# IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to TI's Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2019, Texas Instruments Incorporated