Application Note

Jacinto™ 7 DDRSS Register Configuration Tool

Kevin Scholz

ABSTRACT

At the center of every application is the need for memory. With limited on-chip processor memory, external memory serves as a solution for large software systems and data storage, and an unstable external memory interface can result in system failures or hinder software development. To prevent potential system level anomalies and ensure robust systems, hardware must be configured correctly and tested thoroughly.

The Jacinto 7 DDRSS Register Configuration Tool focuses on post layout activities, and provides a simplified solution to configure the Texas Instruments (TI) Jacinto 7 processors for accessing the specific double data rate (DDR) memory part number that is selected for a system. This document provides a detailed description on how to use the associated application files to generate appropriate register settings for a unique system and memory component, updating the source code of supported software development kits (SDKs), and address common questions or issues that may arise. The document introduction provides a complete list of processors and memory types supported by the Jacinto 7 DDRSS Register Configuration Tool.

The spreadsheet discussed in this document can be downloaded from the following URL: https://www.ti.com/lit/zip/spracu8.

.

Table of Contents

1 Introduction 2
1.1 Features 2
1.2 Spreadsheet Overview 3
1.3 Default SDK Configurations 4
2 Customizing DDR Configuration 5
2.1 Config Worksheet 5
2.2 DRAMTiming Worksheet 8
2.3 IO Control Worksheet 9
3 Software Considerations 12
3.1 Updating U-Boot 12
3.2 Updating RTOS PDK 14
4 Troubleshoot Guide 14
4.1 Topics/Issues 14
5 References 18
Revision History 18

List of Figures

Figure 1-1. Jacinto 7 DDRSS Register Configuration Tool Worksheets 3
Figure 1-2. "Save" Left-Click Push Button 4
Figure 1-3. Steps to Load Default SDK DDR Configurations 4
Figure 4-1. Type Mismatch Error 15
Figure 4-2. Type Mismatch Source 15
Figure 4-3. Workbook Ouput Not Updating Based on User Input 16
Figure 4-4. Enablement of Formula Calculations 16
Figure 4-5. "Cannot run the macro" Error 17
Figure 4-6. Enabling Workbook Macros 17

List of Tables

Table 2-1. Jacinto 7 DDR IO Drive Strength to IBIS Model Mapping....10
Table 2-2. Jacinto 7 DDR IO Termination to IBIS Model Mapping....11

Trademarks

Jacinto™ and E2E™ are trademarks of Texas Instruments.
All trademarks are the property of their respective owners.

1 Introduction

The Jacinto™ 7 DDRSS Register Configuration Tool provides a simplified solution to configuring the Texas Instruments Jacinto 7 processors for accessing DDR memories. The tool consists of this document and a corresponding spreadsheet that generates register settings based on user input. The output of the spreadsheet is intended to be easily integrated with the supporting DDR drivers provided with the software development kits available for each supported processor.

This document provides details pertaining to the features of the tool, as well as steps outlining the procedure to utilize the spreadsheet and update software accordingly.

Note

While some content in this document may apply to other spreadsheet versions of the Jacinto 7 DDRSS Register Configuration Tool, this version of the document is intended specifically for spreadsheet version 0.10.0.

1.1 Features

The key supported and unsupported features of the Jacinto 7 DDRSS Register Configuration Tool are provided in the lists shown in Section 1.1.1 and Section 1.1.2.

Note

The supported/unsupported features listed in this section are not exhaustive. If a feature in question is not listed in either section, submit a ticket on the TI E2E™ design support forums.

1.1.1 Supported Features (version 0.10.0)

- **DDR interface of following TI processor part numbers**: DRA821x, DRA829x, TDA4AH, TDA4AL, TDA4AP, TDA4VE, TDA4VH, TDA4VL, TDA4VM, TDA4VP
- **DDR Memory Types**: LPDDR4
- **DDR Bus Width**: 32 bits or 16 bits
- Single or dual rank LPDDR4 memories
- Customization of IO drive strength / termination
- Configurable DDR timing parameters
- Enable the DDRSS hardware training algorithms during DRAM initialization, including:
  - IO calibration of the controller/PHY
  - Command bus training, including the reference voltage programmed in MR12 of the DRAM
  - Write leveling
  - Read DQS gate training
  - Read data eye (DQ) training, including the reference voltage of the controller / PHY
  - Write data eye (DQ) training, including the reference voltage programmed in MR14 of the DRAM
- Enablement of the following DDRSS hardware periodic training algorithms during normal operation:
  - Write data eye (DQ) training, excluding the reference voltage
  - ZQ Calibration
- Non-power-of-2 LPDDR4 densities (example: 3Gb, 6Gb, 12Gb)
- Data bus inversion

1.1.2 Unsupported Features (version 0.10.0)

- Configuring ECC
- Bit or byte lane swapping
- LPDDR4 frequency set points
- LPDDR4 densities less than 2Gb or greater than 16Gb
- 32-bit READ / WRITE burst length
- Training flexibility: trainings are enabled by default and customization is not supported by the tool

1.2 Spreadsheet Overview

The corresponding spreadsheet consists of three user input worksheets, four register output worksheets, a title page, and a revision history as shown in Figure 1-1.

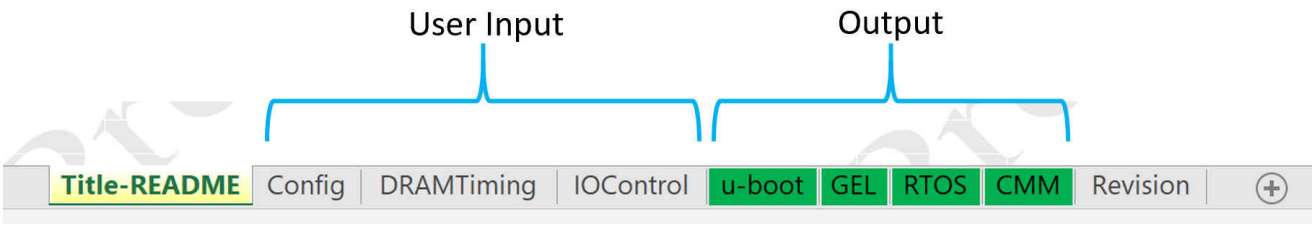

Figure 1-1. Jacinto 7 DDRSS Register Configuration Tool Worksheets

1.2.1 Input Worksheets

The three user input worksheets include:

- **Config**
- **DRAMTiming**
- **IOControl**

The **Config** input worksheet is intended to capture high level details such as the TI processor used in the system, number of DDR subsystems active and the interleaving scheme between them (if applicable), desired DDR frequency, and high level details of the DDR. High level knowledge of the system is required to complete this worksheet.

The **DRAMTiming** input worksheet captures the timing requirements of the DDR, and could vary for different DDR part numbers. The information required to complete this worksheet is extracted from the device-specific data sheet for the DDR component/part number used in the system, which can typically be found on the DDR vendor's website. Inputs are required for each DDRSS in use.

The **IOControl** input worksheet captures desired input/output (IO) settings such as drive strength and termination for both the processor and DDR. The IO settings can be unique for each system, independent of the TI processor and DDR selected. Inputs are required for each DDRSS in use. For more information regarding selecting IO settings, see Section 2.3.1.

1.2.2 Output Worksheets

The four user output worksheets include:

- **u-boot**
- **GEL**
- **RTOS**
- **CMM**

Each output worksheet contains the calculated register settings in the corresponding software repository format. As an example, the **RTOS** worksheet provides the register settings in an array format utilized by the secondary bootloader (SBL) DDR driver, while the **u-boot** worksheet provides the register settings as a #define preprocessor directive. In addition, each worksheet contains a "Save" left-click push button to allow the contents of the worksheet to be saved to your hard drive. Figure 1-2 shows an example of a "Save" left-click push button.

Figure 1-2. "Save" Left-Click Push Button

Note

The workbook formula calculation options must be set to "Automatic", or the "Calculate Now" button must be utilized, to generate the proper register settings based on the user input. You should ensure that the calculations have been performed before integrating the register settings into software. For more information, see Section 4.1.2 of Section 4.

1.2.3 Other Worksheets

The title page has generic information and allows you to pre-populate the workbook with inputs used to generate the SDK evaluation board (EVM) default values. For more information, see Section 1.3.

The “Revision” worksheet provides the history of changes for each revision of the workbook, including feature changes as well as documenting registers impacted by formula updates.

1.3 Default SDK Configurations

The DDR configurations for the TI EVMs included in the software development kits are generated by the Jacinto 7 DDRSS Register Configuration Tool. As the SDK only includes the output register values, the Jacinto 7 DDRSS Register Configuration Tool includes macros to populate the user inputs to match the corresponding SDK configuration.

To use the macros and populate the tool to match the user inputs used to generate the TI EVM configuration, follow the steps listed below:

1. Navigate to the "Title-README" worksheet of the tool.
2. Select the appropriate TI processor part number from the drop down menu as shown in Figure 1-3.
3. Left click the "Load User Config" push button.

Note

Macros of the Excel workbook must be enabled for this feature to work. For more information, see Section 4.1.3 of Section 4.

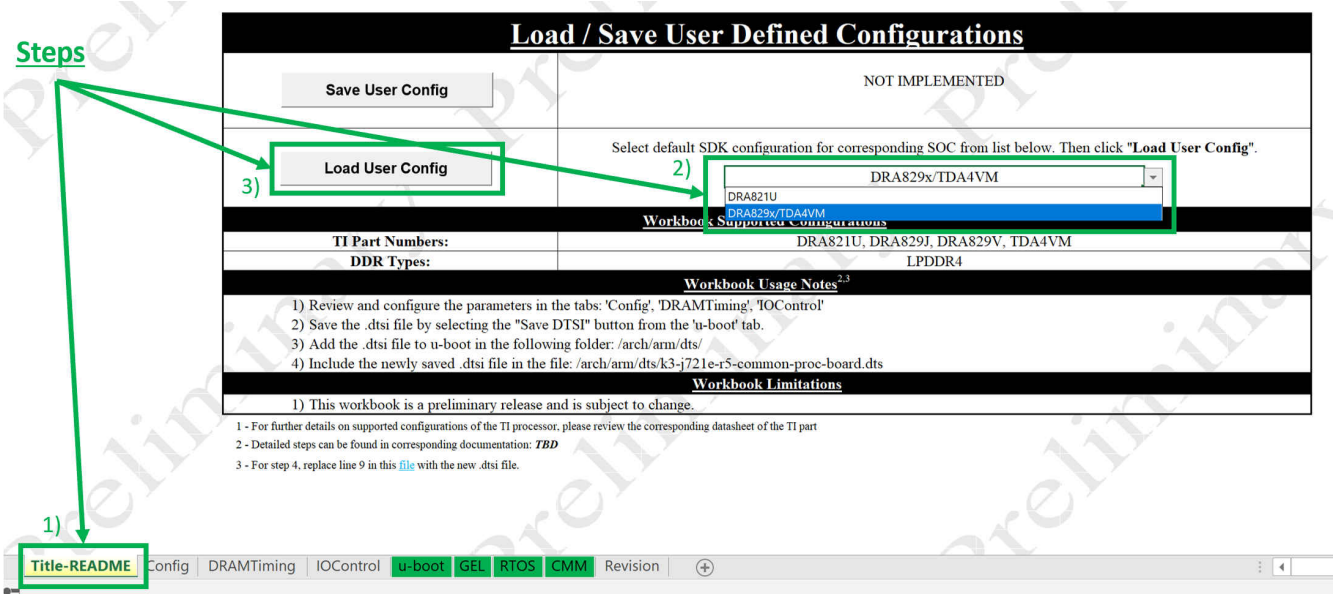

Figure 1-3. Steps to Load Default SDK DDR Configurations

2 Customizing DDR Configuration

The Jacinto 7 DDRSS Register Configuration Tool allows you to easily customize the DDRSS register values to ensure proper operation for a given system. As described in Section 1.2, the corresponding workbook consists of three user input worksheets that control the register configuration output. This section provides detailed information pertaining to each input worksheet of the tool.

2.1 Config Worksheet

The first worksheet to be configured is the "Config" worksheet, which is divided into two sections. The first section of the worksheet, *System Configuration*, is where high-level details regarding the custom design would be entered, and more than likely will need customization compared to the default inputs. The second section of the worksheet, *Memory Burst Configuration*, allows some customization of the read/write burst of the DDR interface. If unsure on how to set the parameters of the *Memory Burst Configuration* section in the workbook, it is recommended to match the TI default value.

2.1.1 System Configuration

Additional details of each parameter of this section can be found in the list below:

1. **Board Project Name**:
   a. Description: This parameter is not used in version 0.10.0 or earlier of the Jacinto 7 DDRSS Register Configuration Tool.
   b. Valid Values: NA
   c. Recommended Value: NA
2. **TI System-on-Chip (SoC) Part Number**:
   a. Description: Defines the TI processor part number used in the system.
   b. Valid Values: Defined by a drop-down menu list, matching the supported TI processors by the tool.
   c. Recommended Value: This parameter must be selected to match the TI processor part number being used in the system. The output of the tool could vary depending on the selected SoC, so it is imperative that the correct part number is used!
3. **DDR Controllers Utilized in System**:
   a. Description: Defines which DDRSS instances of the TI SoC are connected to external memory and should be configured by software.
   b. Valid Values: Defined by a drop-down menu list, matching the supported combination of DDR sub-systems that can be enabled for the selected TI SoC.
   c. Recommended Value: N/A

   Note

   The drop-down list for this parameter is dependent on the user input of the 'TI SoC Part Number.
4. **DDR Memory Type**:
   a. Description: Defines the type of DDR memory used in the system.
   b. Valid Values: Defined by a drop-down menu list, matching the supported DDR memory types by the tool.
   c. Recommended Value: Only one option, "LPDDR4", is supported by the tool.
5. **DDR Memory Boot Frequency (Frequency Set 0)**:
   a. Description: Defines the DDR memory clock frequency during the DRAM initialization, or ***tCKb*** as defined in the specific DDR component data sheet.
   b. Valid Values: For supported ***tCKb*** frequencies, see the specific DDR component data sheet.
   c. Recommended Value: It is recommended to match the TI default value.

Note

The usage of this parameter has different impact depending on the versions of SDK and Jacinto 7 DDRSS Register Configuration Tool being used. Prior to SDK8.0 and Jacinto 7 DDRSS Register Configuration Tool version 0.6.0, this parameter impacted the *_F0 timing parameters but did NOT control the actual LPDDR4 boot frequency. Rather, the LPDDR4 boot frequency is 2x the PLL12 bypass clock, or 2x the oscillator frequency, in SDKs released prior to SDK8.0. Starting with SDK8.0 and Jacinto 7 DDRSS Register Configuration Tool version 0.6.0, this parameter impacts both the *_F0 timing parameters and the LPDDR4 boot frequency.

6. **DDR Memory Frequency (Frequency Set 1)**:
   a. Description: Defines the target DDR memory clock frequency during normal operation when the LPDDR4 is configured for frequency set point 0.
   b. Valid Values: Not configurable. As the tool does not support different frequency set points, this value must match input parameter DDR Memory Frequency (Frequency Set 2), and automatically updates in the tool to reflect this requirement.
   c. Recommended Value: NA
7. **DDR Memory Frequency (Frequency Set 2)**:
   a. Description: Defines the target DDR memory clock frequency during normal operation when the LPDDR4 is configured for frequency set point 1.
   b. Valid Values: For supported frequencies, see the TI device-specific data sheet, as well as the specific DDR component data sheet. Note that this parameter should be set to the clock rate, and not the data rate.
   c. Recommended Value: Value must be within the min/max supported limits of both the TI processor DDRSS and DDR.
8. **DDR Data Bus Width**:
   a. Description: Defines the bus width utilized by the DDR interface.
   b. Valid Values: Defined by a drop-down menu list, matching the supported bus widths by the tool.
   c. Recommended Value: Value should be set to match the number of physical data (DQ) IO pins which are connected between the processor's DDRSS and the LPDDR4 memory on the printed circuit board (PCB).
9. **DDR Density**:
   a. Description: Defines the density of a single channel from a single rank of the LPDDR4 memory. As an example, if the LPDDR4 memory has total density of 32Gb across 2 channels and 2 ranks, then this parameter should be set to 8Gb. Or if the LPDDR4 memory has total density of 8Gb across 2 channels and 1 rank, then this parameter should be set to 4Gb.
   b. Valid Values: Defined by a drop-down menu list, matching the supported densities by the tool.
   c. Recommended Value: Value should be set to match the density, as described by this parameter's description, of the LPDDR4 used in the system.
10. **Chip Selects / Ranks**:
    a. Description: Defines the number of ranks, or chip selects, utilized for the DDR interface. Although the LPDDR4 interface has a unique chip select for each channel, usage of CS0_A and CS0_B should be considered 1 rank. Usage of CS0_A and CS1_A should be considered 2 ranks. In other words, this parameter defines the number of chip selects per channel.
    b. Valid Values: Defined by a drop-down menu list, matching the supported ranks by the tool.
    c. Recommended Value: Value should be set to match the number of ranks, as described by this parameter's description, utilized by the DDR interface.
11. **Enable DRAM Temperature Polling**:
    a. Description: This parameter enables or disables LPDDR4 temperature polling during normal operation. When enabled, the controller will periodically send a Mode Register Read request to the LPDDR4 to read MR4. The purpose of enabling DRAM temperature polling would be to dynamically change the refresh rate depending on the temperature of the LPDDR4.
    b. Valid Values: Defined by a drop-down menu list.
    c. Recommended Value: Usage of this feature will be system dependent, and could be impacted by the temperatures the LPDDR4 may be subject to in the end application, as well as by throughput requirements of the system.

Note

Enabling this parameter ONLY allows the controller to periodically read MR4 of the LPDDR4. It does NOT change the refresh rate. A software interrupt service routine is required to service the changes in temperature.

If the refresh rate is not changed dynamically, you should ensure that the fastest refresh rate required by the LPDDR4 is programmed in the corresponding timing parameters in the "DRAMTiming" worksheet.

Temperature de-rating must be accounted for in the "DRAMTiming" worksheet, regardless of the configuration of this parameter.

12. **System Temperature Gradient**:
    a. Description: Defines the maximum temperature gradient the system, specifically the DDR, will be subject to in the target end application. In other words, this parameter defines how quickly the temperature of the DDR will change.
    b. Valid Values: Any decimal value greater than zero.
    c. Recommended Value: NA, this parameter is system dependent and must be defined by the end user.

Note

This parameter is not used when DRAM temperature polling is disabled.

13. Multi DDRSS Interleave Hybrid Config
    a. Description: Determines how the DDR subsystems are interleaved when multiple DDR subsystems are in use.
    b. Valid Values: Defined by a drop-down menu list, matching the supported configurations for the selected TI SOC and number of DDR subsystems in use.
14. Multi DDRSS Interleave Memory Size
    a. Description: Determines the size of the interleaved memory region.
    b. Valid Values: Defined by a drop-down menu list, matching the supported configurations based on the number of DDR subsystems in use and the corresponding LP4 memories connected.
15. Multi DDRSS Interleave Granularity
    a. Description: Determines the granularity at which the DDR subsystems are interleaved within the interleaved memory region.
    b. Valid Values: Defined by a drop-down menu list, matching the supported configurations based on the number of DDR subsystems in use and the interleaved memory region size.

2.1.2 Memory Burst Configuration

Additional details of each parameter of this section can be found in the list below:

1. **Burst Length**:
    a. Description: This parameter corresponds to MR1[1:0] of the LPDDR4 memory and defines the number of data bits transferred on each data pin during a single READ or WRITE command.
    b. Valid Values: Defined by a drop-down menu list, matching the supported burst length by the tool. Although additional configurations may be supported by the LPDDR4 memory, only 16-bit sequential burst length is supported by the tool.
2. **Read Preamble**:
    a. Description: This parameter corresponds to MR1[3] of the LPDDR4 memory and defines whether the READ preamble toggles.
    b. Valid Values: Defined by a drop-down menu list.
3. **Read Postamble**:
    a. Description: This parameter corresponds to MR1[7] of the LPDDR4 memory and defines the length of the READ postamble in clock cycles.
    b. Valid Values: Defined by a drop-down menu list.

4. **Write Postamble**:
   a. Description: This parameter corresponds to MR3[1] of the LPDDR4 memory and defines the length of the WRITE postamble in clock cycles.
   b. Valid Values: Defined by a drop-down menu list.
5. **Data Bus Inversion (Read)**:
   a. Description: This parameter corresponds to MR3[6] of the LPDDR4 memory and defines whether the data bus inversion (DBI) function is enabled during READs.
   b. Valid Values: Defined by a drop-down menu list.
6. **Data Bus Inversion (Write)**:
   a. Description: This parameter corresponds to MR3[7] of the LPDDR4 memory and defines whether the data bus inversion (DBI) function is enabled during WRITEs.
   b. Valid Values: Defined by a drop-down menu list.

2.2 DRAMTiming Worksheet

The second worksheet to be configured is the "DRAMTiming" worksheet. The "DRAMTiming" worksheet is divided into two sections: latency parameters that vary with frequency and non-latency parameters that generally will not change with frequency. The timing parameters were divided into these two sections to simplify the procedure of updating inputs for different frequencies. The values to be entered into the "DRAMTiming" worksheet parameters should be based on the specific DDR component data sheet.

Note

You are responsible to ensure that the timing values input into the spreadsheet adhere to the device-specific DDR component data sheet used in the corresponding system.

2.2.1 Latency Parameters

As the latency parameters are frequency dependent, each parameter has a unique input for frequency 0 (F0), frequency 1 (F1), and frequency 2 (F2). F0, F1, and F2 are defined by you in the "Config" worksheet. Thus, you should set all three inputs for each latency parameter to correspond with the appropriate value as defined in the specific DDR component data sheet for the given frequency. For example values based on the clock frequency, see **Table 100** and **Table 167** of ***JESD209-4D*** at https://www.jedec.org/system/files/docs/JESD209-4D.pdf. Additional details of each latency parameter can be found in the list below:

1. **Read Latency**: This parameter should be set to match the read latency of the DDR at the defined frequency, as well as based on whether read DBI is enabled. Valid input is defined by a drop-down list.

   Note

   The drop-down list for read latency is dependent on the user input of the "Data Bus Inversion (Read)" parameter defined in the "Config" worksheet.

2. **Write Latency Set**: This parameter defines the write latency set. Valid input is defined by a drop-down list. It is recommended to keep this setting as the default of the tool.
3. **Write Latency**: This parameter should be set to match the write latency of the DDR at the defined frequency, as well as based on the write latency set selected. Valid input is defined by a drop-down list.

   Note

   The drop-down list for write latency is dependent on the user input of the "Write Latency Set" parameter defined in the "DRAMTiming" worksheet.

4. **Write Recovery**: This parameter should be set to match the write recovery time of the DDR at the defined frequency. Valid input is defined by a drop-down list.
5. **ODTLon**: This parameter should be set to match the ODTLon latency of the DDR at the defined frequency, as well as based on the write latency set selected. Valid input is defined by a drop-down list.

   Note

   The drop-down list for ODTLon is dependent on the user input of the "Write Latency Set" parameter defined in the "DRAMTiming" worksheet.

6. **ODTLoff**: This parameter should be set to match the ODTLoff latency of the DDR at the defined frequency, as well as based on the write latency set selected. Valid input is defined by a drop-down list.

Note

The drop-down list for ODTLoff is dependent on the user input of the "Write Latency Set" parameter defined in the "DRAMTiming" worksheet.

2.2.2 Non-Latency Parameters

In addition to the latency parameters, there are several additional DDR timings that need to be input into the Jacinto 7 DDRSS Register Configuration Tool. All of these parameters should be set based on the values obtained from the specific DDR component data sheet.

Note

Some parameters may require de-rating at high temperature. You are responsible for entering the de-rated timing into the spreadsheet; this is not accounted for in software.

2.3 IO Control Worksheet

The third and final worksheet for you to configure is the "IOControl" worksheet, which separates the IO parameters into two sections based on the corresponding integrated circuit (IC) component. Both sections allow customization of similar types of parameters:

- Reference Voltage (VREF)
- Driver Strength
- On-DIE Termination

2.3.1 Determining IO Settings

The optimal IO settings may vary from system to system due to board layout and routing differences, even if the systems utilize the same processor and DDR. Thus, it is important to perform simulations as described in the *Jacinto7 LPDDR4 Board Design and Layout Guidelines* and analyze the waveforms to identify the settings which give the best margins. This procedure may require several iterations of simulations before discovering the best settings. As a general starting point, you can select drive strength and termination settings to match the trace impedance on the PCB. For more information on performing the simulations, see the *Jacinto7 LPDDR4 Board Design and Layout Guidelines*.

2.3.2 Processor/DDR Controller IO

Additional details of each parameter of this section can be found in the list below:

1. **VREF Control**: The VREF control parameters impact the reference voltage used for inputs on the processor data (DQ) and strobe (DQS) input/output pins during READ cycles.
   a. **Range**: This parameter defines the range of reference voltage values available in the succeeding parameter, "*% of VDDQ*". The recommended configuration for this parameter is "Range 0".
   b. **% of VDDQ**: This parameter defines the target reference voltage level, as a percentage of the IO voltage, *vdds_ddr*. The recommended configuration of this parameter is to set to half of the DDR pull-up calibration configuration. As an example, if the DDR pull-up calibration is set to "VDDQ / 3", then this parameter should be set to [(1/3) / 2] = 16.67% of VDDQ.

Note

The actual reference voltage used during normal operation is determined by the outcome of the VREF training algorithm performed during the initialization of the DDR interface.

2. **Drive Strength**: The drive strength parameters impact the voltage swing and signal integrity of the processor DDR pins during WRITE cycles. As discussed in Section 2.3.1, the appropriate value should be selected based on the IO model settings used to achieve the best simulation results. Table 2-1 illustrates the mapping between the IBIS IO model name and the appropriate drive strength parameter value.

Table 2-1. Jacinto 7 DDR IO Drive Strength to IBIS Model Mapping

Tool Parameter	IO Model (1) (2)	Corresponding Parameter Value
Driver Pull-Up	lpddr4_ocd_240p_240n	240 Ω
	lpddr4_ocd_120p_120n	120 Ω
	lpddr4_ocd_80p_80n	80 Ω
	lpddr4_ocd_60p_60n	60 Ω
	lpddr4_ocd_48p_48n	48 Ω
	lpddr4_ocd_40p_40n	40 Ω
	lpddr4_ocd_120pd_60p_40n(3)	Not Supported
	lpddr4_ocd_120pd_48p_40n(3)	Not Supported
Driver Pull-Down	lpddr4_ocd_240p_240n	240 Ω
	lpddr4_ocd_120p_120n	120 Ω
	lpddr4_ocd_80p_80n	80 Ω
	lpddr4_ocd_60p_60n	60 Ω
	lpddr4_ocd_48p_48n	48 Ω
	lpddr4_ocd_40p_40n	40 Ω
	lpddr4_ocd_120pd_60p_40n(3)	Not Supported
	lpddr4_ocd_120pd_48p_40n(3)	Not Supported

(1) Model names based on IBIS file *j7es_v0p2.ibs*, *DRA829 and TDA4VM IBIS File*. While the IO model name should be the same across Jacinto 7 processors, the correct processor IBIS model must be used for simulations and can be obtained from the corresponding product home page.
(2) Model names also used to represent the differential version of the model (models ending in ***_diff***)
(3) This IO model is not currently supported by the Jacinto 7 DDRSS Register Configuration Tool.

3. **Termination**: The termination parameters impact the voltage swing and signal integrity of the processor DDR pins during READ cycles. As discussed in Section 2.3.1, the appropriate value should be selected based on the IO model settings used to achieve the best simulation results. Table 2-2 illustrates the mapping between the IBIS IO model name and the appropriate termination parameter value.

Table 2-2. Jacinto 7 DDR IO Termination to IBIS Model Mapping

Tool Parameter	IO Model (1) (2)	Corresponding Parameter Value
ODT Pull-Up	lpddr4_odt_240	Hi-Z
	lpddr4_odt_120	Hi-Z
	lpddr4_odt_80	Hi-Z
	lpddr4_odt_60	Hi-Z
	lpddr4_odt_48	Hi-Z
	lpddr4_odt_40	Hi-Z
	lpddr4_odt_off	Hi-Z
ODT Pull-Down	lpddr4_odt_240	240 Ω
	lpddr4_odt_120	120 Ω
	lpddr4_odt_80	80 Ω
	lpddr4_odt_60	60 Ω
	lpddr4_odt_48	48 Ω
	lpddr4_odt_40	40 Ω
	lpddr4_odt_off	Hi-Z

(1) Model names based on IBIS file *j7es_v0p2.ibs*, *DRA829 and TDA4VM IBIS File*. While the IO model name should be the same across Jacinto 7 processors, the correct processor IBIS model must be used for simulations and can be obtained from the corresponding product home page.
(2) Model names also used to represent the differential version of the model (models ending in ***_diff***)

2.3.3 DRAM I/O

Additional details of each parameter of this section can be found in the list below:

1. **VREF Control**:
   a. **VREF Range (DQ or CA)**: This parameter corresponds to MR14[6] for DQ signals and MR12[6] for command / address signals, and defines which VREF range is used for the respective signals.
   b. **VREF (DQ or CA)**: This parameter corresponds to MR14[5:0] for DQ signals and MR12[5:0] for command / address signals, and defines the target reference voltage level, as a percentage of the I/O voltage.
2. **Drive Strength**:
   a. **Pull-Down (PDDS)**: This parameter corresponds to MR3[5:3] of the LPDDR4 memory and defines the drive strength of the DDR data (DQ) and strobe (DQS) pins during READ cycles. As discussed in Section 2.3.1, the appropriate value should be selected based on the I/O model settings used to achieve the best simulation results.
   b. **Pull Up Calibration**: This parameter corresponds to MR3[0] of the LPDDR4 memory and defines the target VOH during READ cycles. It is recommended to leave this parameter set to the default, "VDDQ / 3".

3. **Termination**:
   a. **CA ODT Disable**: This parameter corresponds to MR22[5] of the LPDDR4 memory. When this parameter is set to "Disable", the termination of the command / address pins are disabled regardless of how the termination is configured in MR11 or the state of the ODT_CA pin. When this parameter is set to "ODT_CA Bond Pad", the termination of the command / address pins are configured based on the MR11 configuration along with the ODT_CA pin. It is recommended to leave this parameter set to the default, "ODT_CA Bond Pad".
   b. **CK ODT Override**: This parameter corresponds to MR22[3] of the LPDDR4 memory. When set to "Enable", the clock termination is determined by the MR11 configuration regardless of the ODT_CA pin. This parameter is used to enable termination on the clock when the CA bus is shared between two ranks, but the clock is not. Because the Jacinto 7 processors share both the CA bus and clock between ranks, it is recommended to leave this parameter set to the default, "Disable".
   c. **CS ODT Override**: This parameter corresponds to MR22[4] of the LPDDR4 memory. When set to "Enable", the chip select termination is determined by the MR11 configuration regardless of the ODT_CA pin. This parameter is used to enable termination on the chip select pin when the CA bus is shared between two ranks, but the chip select is not. Because the Jacinto 7 processors share the CA bus between ranks but have unique chip select signals, it is recommended to leave this parameter set to the default, "Enable".
   d. **CA ODT**: This parameter corresponds to MR11[6:4] of the LPDDR4 memory and defines the termination of the command / address pins of the LPDDR4 memory. As discussed in Section 2.3.1, the appropriate value should be selected based on the I/O model settings used to achieve the best simulation results.
   e. **DQ ODT**: This parameter corresponds to MR11[2:0] of the LPDDR4 memory and defines the termination of the data (DQ), data mask (DM), and strobe (DQS) pins of the LPDDR4 memory during WRITE cycles. As discussed in Section 2.3.1, the appropriate value should be selected based on the I/O model settings used to achieve the best simulation results.
   f. **SOC ODT**: This parameter corresponds to MR22[2:0] of the LPDDR4 memory and defines the termination of the processor / DDR controller. This parameter must be configured to match the termination as defined in 3.

3 Software Considerations

This section provides details on how to incorporate the Jacinto 7 DDRSS Register Configuration Tool output with existing software repositories.

3.1 Updating U-Boot

The Jacinto 7 DDRSS Register Configuration Tool outputs the unique register settings in a device tree source include (DTSI) file that can be used with the associated drivers in u-boot to initialize the LPDDR4 interface.

This section describes in detail how to update u-boot source code using the Jacinto 7 DDRSS Register Configuration Tool output. This section also describes u-boot source code updates that are required to address the available memory size.

Note

This document does not cover building the source files. Please refer to the appropriate SDK documentation for steps to re-build the binary files.

3.1.1 Updating DDR Register Settings

To update the DDRSS register settings, the following steps should be followed:

1. Navigate to the "u-boot" worksheet (once all user input worksheets have been completed) and save the DTSI file by selecting the "Save DTSI" button at the top of the worksheet. When prompted, save and add the file to the u-boot source code, placing the file inside the ***<UBOOT_BASE>/arch/arm/dts/*** folder.
2. Update the corresponding R5 common processor device tree source file to include the new DDR DTSI file generated by the Jacinto 7 DDRSS Register Configuration Tool. As an example, the "k3-j721e-r5-common-proc-board" device tree source file includes the Jacinto 7 DDRSS Register Configuration Tool output DTSI file ("k3-j721e-ddr-evm-lp4-4266.dtsi") when building the R5 boot loader for the TDA4VM EVM.

   Source: arch/arm/dts/k3-j721e-r5-common-proc-board.dts

```
// SPDX-License-Identifier: GPL-2.0
/*
 * Copyright (C) 2019 Texas Instruments Incorporated - http://www.ti.com/
 */

/dts-v1/;

#include "k3-j721e-som-p0.dtsi"
#include "k3-j721e-ddr-evm-lp4-4266.dtsi"
#include "k3-j721e-ddr.dtsi"
```

3. Rebuild the source code by following the instructions in the SDK documentation.

3.1.2 Updating Source to Set Available Memory Size

In addition to updating the DDRSS registers, other source files in u-boot may need to be updated for the system to operate as expected.

As an example, the total available memory size on a custom board may differ compared to the TI EVM. The default software in the SDK makes use of the u-boot global data and board info structures, specifically the *ram_size* variable, which should be configured to match the total DDR memory size, and the *bi_dram* structure's *start* and *size* parameters, which map the processor's corresponding address space to the DDR memory region. However, these parameters are not configured based on register settings or output from the Jacinto 7 DDRSS Register Configuration Tool. Thus, these variables must be updated to ensure that the system does not try to access unavailable physical memory when the custom board has less memory compared to the EVM, as well as allow the system to utilize the full DDR memory space when the custom board has more memory compared to the EVM.

In the default SDK source code, the *ram_size* and *bi_dram* structure variables are configured in functions *dram_init* and *dram_init_banksize*, located in the corresponding processor board file. An example is provided below:

Source: board/ti/j721e/evm.c

As shown below, the function *dram_init* configures the global data variable *ram_size*. This function should be modified in custom code and *ram_size* should be configured to match the total available DDR memory.

```
int dram_init(void)
{
#ifdef CONFIG_PHYS_64BIT
    gd->ram_size = 0x100000000;
#else
    gd->ram_size = 0x80000000;
#endif

    return 0;
}
```

As shown below, the function *dram_init_banksize* configures the global data variables *ram_size* and board info *bi_dram*. This function should be modified in custom code. The variable *ram_size* should be configured to match the total available DDR memory. The *start* and *size* parameters of the *bi_dram* structure should be configured to match the available memory for each DDR section. For Jacinto 7 processors, the DDR memory is split into a low and high region. The low region is 32-bit addressable, but limited to 2GB. In the example below, 4GB is split across the low and high region such that each region is mapped to 2GB.

```
int dram_init_banksize(void)
{
    /* Bank 0 declares the memory available in the DDR low region */
    gd->bd->bi_dram[0].start = CONFIG_SYS_SDRAM_BASE;
    gd->bd->bi_dram[0].size = 0x80000000;
    gd->ram_size = 0x80000000;

#ifdef CONFIG_PHYS_64BIT
    /* Bank 1 declares the memory available in the DDR high region */
    gd->bd->bi_dram[1].start = CONFIG_SYS_SDRAM_BASE1;
    gd->bd->bi_dram[1].size = 0x80000000;
    gd->ram_size = 0x100000000;
#endif

    return 0;
}
```

3.2 Updating RTOS PDK

The Jacinto 7 DDRSS Register Configuration Tool outputs the unique register settings in a header file that can be used with the associated DDR drivers in the Platform Development Kit (PDK) included in the RTOS SDK. This section describes in detail how to update the PDK source code using the Jacinto 7 DDRSS Register Configuration Tool output.

Note

This document does not cover building the source files. For steps to re-build the binary files, see the appropriate SDK documentation.

3.2.1 Updating DDR Register Settings

To update the DDRSS register settings, the following steps should be followed:

1. Once all user input worksheets have been completed, navigate to the "RTOS" worksheet and save the header file by selecting the "Save RTOS Header File" button at the top of the worksheet. When prompted, save the file to the RTOS PDK source code, replacing the *board_ddrRegInit.h* file inside the corresponding board include folder. Example directory paths are included below:
   a. **TDA4VM EVM**: <RTOS_INSTALL_DIR>/pdk_jacinto_xx_xx_xx/packages/ti/board/src/j721e_evm/include/
   b. **DRA821x EVM**: <RTOS_INSTALL_DIR>/pdk_jacinto_xx_xx_xx/packages/ti/board/src/j7200_evm/include/
2. Re-build the binary files per the steps outlined in the corresponding SDK documentation.

4 Troubleshoot Guide

The following section documents initial steps to try for corresponding issues that could occur. If the steps here do not resolve the issue, submit a ticket on the TI E2E design support forums.

4.1 Topics/Issues

1. When I try to save the register output, I receive an error that states "Run-time error '13': Type mismatch". I also observe "#N/A" for some of the register values (as shown in Figure 4-1).
2. When I change inputs to the worksheets and save the register settings, I notice that the values are the same as they were before (as shown in Figure 4-3).
3. When I select any of the push buttons within the workbook, I receive an error message that states "Cannot run the macro ..." (as shown in Figure 4-5).

4.1.1 Topic 1

Issue: When I try to save the register output, I receive an error that states "Run-time error '13': Type mismatch". I also observe "#N/A" for some of the register values (as shown in Figure 4-1).

```
#define DDRSS_CTL_170_DATA 0x021600D6
#define DDRSS_CTL_171_DATA 0x02161010
#define DDRSS_CTL_172_DATA 0x00000000
#define DDRSS_CTL_173_DATA 0x00000000
#define DDRSS_CTL_174_DATA 0x00000000
        #N/A
        #N/A
#define DDRSS_CTL_177_DATA 0x00003333
#define DDRSS_CTL_178_DATA 0x56000000
#define DDRSS_CTL_179_DATA 0x27270056
#define DDRSS_CTL_180_DATA 0x0F0F0000
#define DDRSS_CTL_181_DATA 0x16000000
#define DDRSS_CTL_182_DATA 0x00841616
        #N/A
#define DDRSS_CTL_184_DATA 0x33333300
#define DDRSS_CTL_185_DATA 0x00000000
```

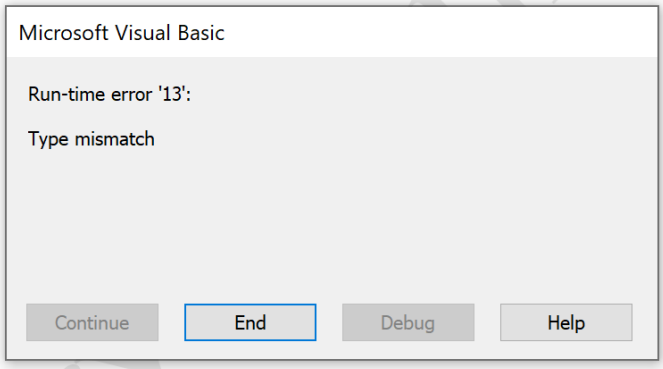

Figure 4-1. Type Mismatch Error

Solution: Double check that all parameters which have dependencies on other parameters are set to a valid value. Dependencies are documented with "Notes" for each parameter in Section 2.

Explanation: A type mismatch error is likely due to an error in the user input. Some inputs are dependent on other inputs of the spreadsheet. As an example, the drop-down options for "Write Latency" on the "DRAMTiming" worksheet are dependent on the input of "Write Latency Set". Thus, the value selected for "Write Latency" may not be valid if the value for "Write Latency" is configured, and then the value of "Write Latency Set" is changed. As shown in Figure 4-2, the previous "Write Latency" value of "34" is no longer valid after changing the "Write Latency Set" value from "WL set B" to "WL set A".

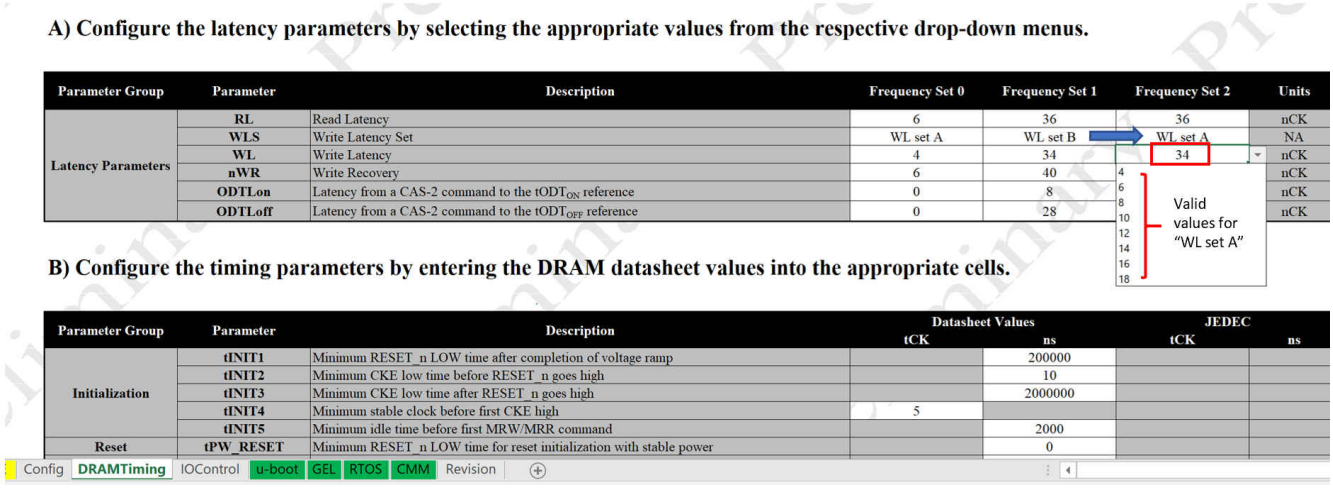

Figure 4-2. Type Mismatch Source

4.1.2 Topic 2

Issue: When I change inputs to the worksheets and save the register settings, I notice that the values are the same as they were before (as shown in Figure 4-3).

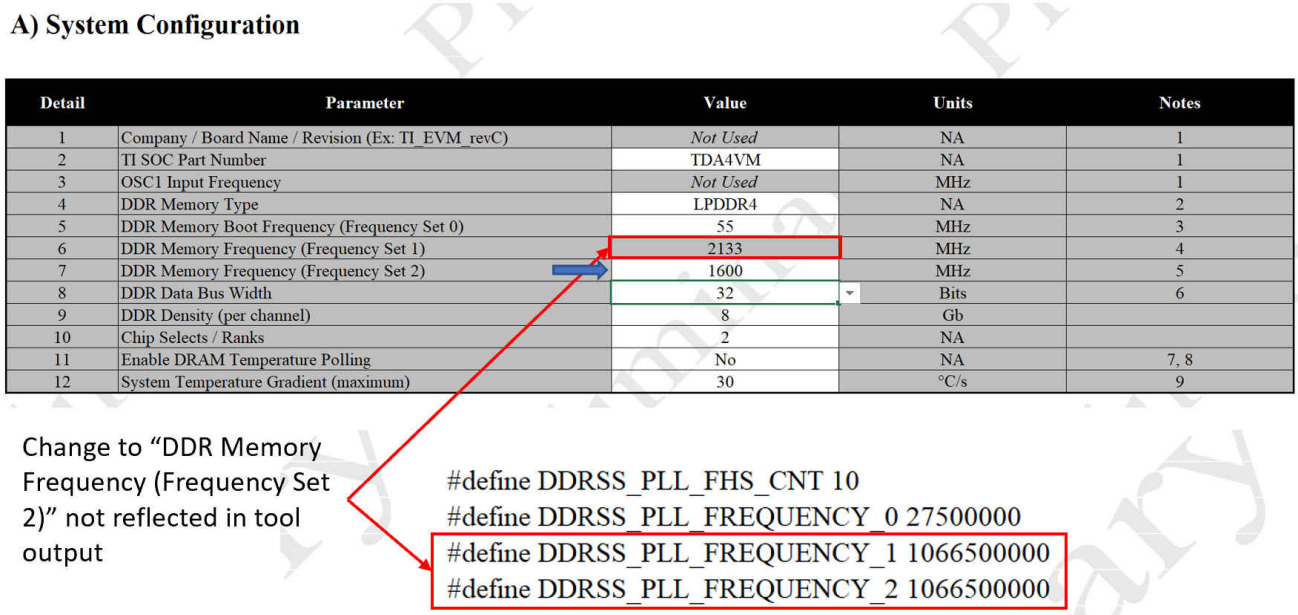

Figure 4-3. Workbook Ouput Not Updating Based on User Input

Solution: Try one of the following two options:

1. Turn on automatic formula calculations from the workbook Formulas Ribbon (as shown in Figure 4-4).
2. Select the "Calculate Now" button from the workbook Formulas Ribbon (as shown in Figure 4-4).

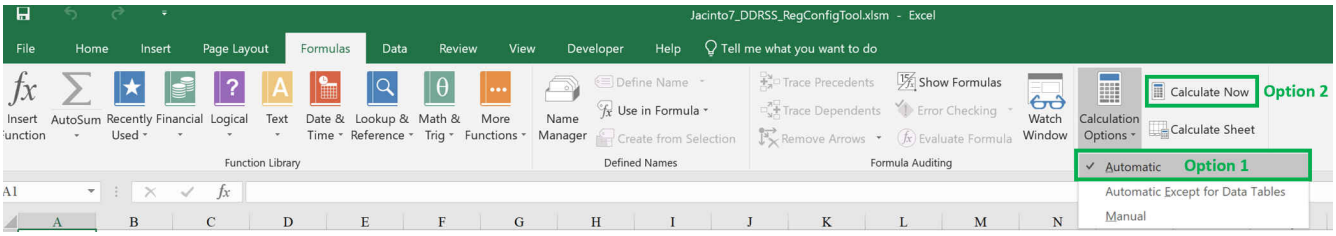

Figure 4-4. Enablement of Formula Calculations

Explanation: The register settings are generated with formulas and not macros. The formulas must either be calculated automatically or by selecting the "Calculate Now" button from the workbook Formulas Ribbon.

4.1.3 Topic 3

Issue: When I select any of the push buttons within the workbook, I receive an error message that states "Cannot run the macro ..." (as shown in Figure 4-5).

Figure 4-5. "Cannot run the macro" Error

Solution: Enable macros for the workbook using the steps listed below and as shown in Figure 4-6:

1. From the workbook "File" menu, select "Options"
2. Select "Trust Center"
3. Select "Trust Center Settings..."
4. Select "Macro Settings"
5. Select "Disable all macros with notification"
6. Save the workbook to ensure any content modified is saved. Close the workbook and then re-open.
7. When prompted upon re-opening the file, select "Enable Content"

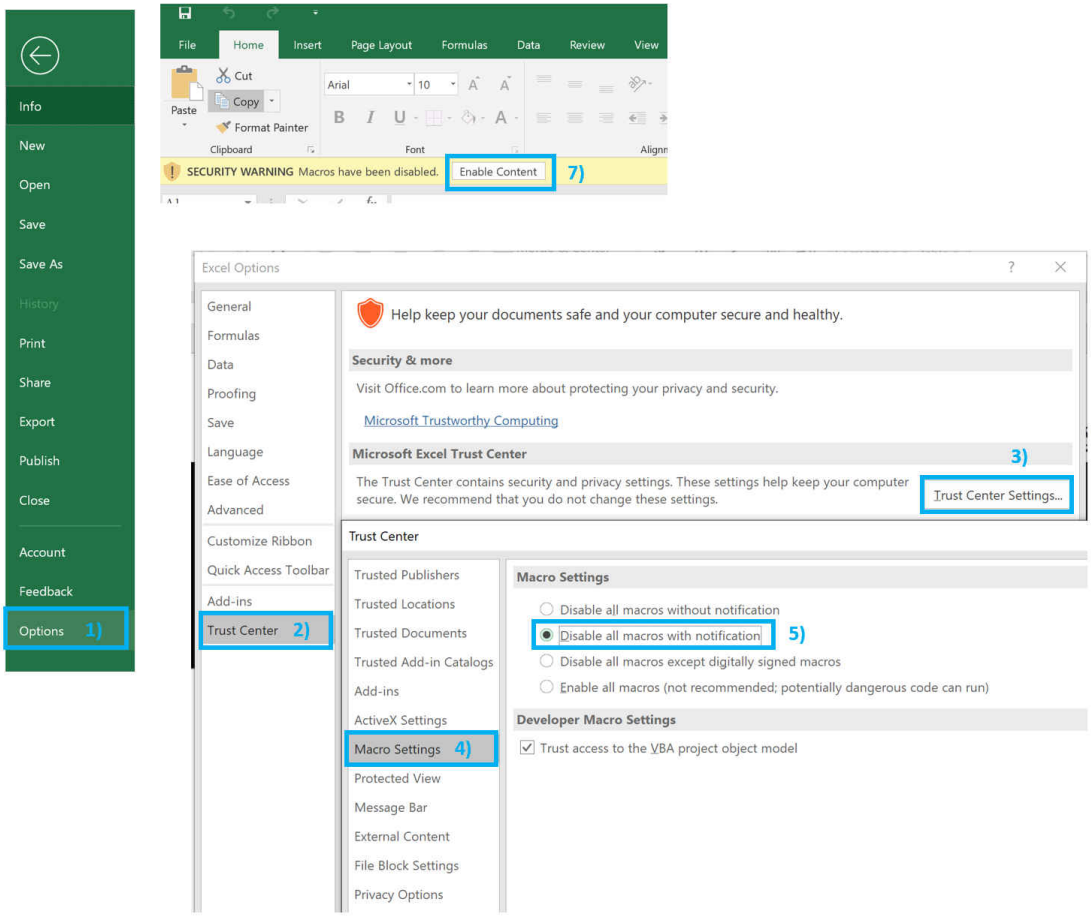

Figure 4-6. Enabling Workbook Macros

Explanation: Macros may be disabled by default on some computers, and may not provide any option to enable when the file is opened. However, the push buttons rely on macros included with the workbook and thus must be enabled to work properly.

5 References

- *Jacinto7 LPDDR4 Board Design and Layout Guidelines*
- *DRA829 and TDA4VM IBIS File*

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Revision A (July 2022) to Revision B (January 2023)	Page
• Updated the numbering format for tables, figures and cross-references throughout the document	2
• Updated Section 1	2
• Updated Section 1.1.1	2
• Updated Section 1.1.2	3
• Updated Section 2.1.1	5

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), 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, regulatory 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 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.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265