



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



## Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

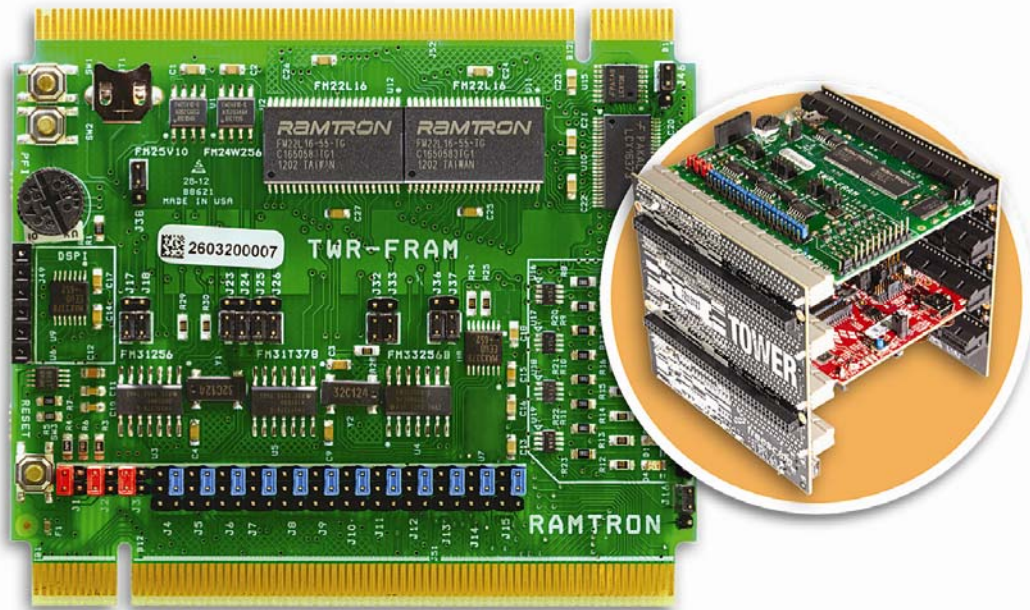
Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



# TWR-FRAM

## F-RAM Memory Module



## Quick Start Guide

## Contents

1 Overview .....	3
2 HW/SW Requirements .....	4
3 Hardware Instructions .....	5
4 Jumper Instructions and Table .....	7
4.1 Jumper Instructions .....	7
4.2 Jumper Table .....	8
5 Software Instructions .....	10
5.1 IAR Embedded Workbench for ARM .....	10
5.2 Keil $\mu$ Vision4 .....	10
5.3 PE Micro OSBDM Virtual Serial Toolkit .....	10
5.4 Connect the Tower System to PC .....	11
6 Demo Instructions for IAR .....	12
6.1 Running the demo .....	13
6.2 Creating a New Project .....	16
7 Demo Instructions for Keil .....	17
7.1 Running the demo .....	17

## 1 Overview

This document gives instructions towards implementing the demo software available for the Tower F-RAM Module (TWR-FRAM). The TWR-FRAM module is designed to work with the Freescale Tower System.

The Tower F-RAM demo software features:

- Control of processor peripherals (SPI and I2C)
- Access of serial memory (SPI and I2C)
- Programming of state savers
- Access of parallel memory (coming soon)

## 2 HW/SW Requirements

### Hardware

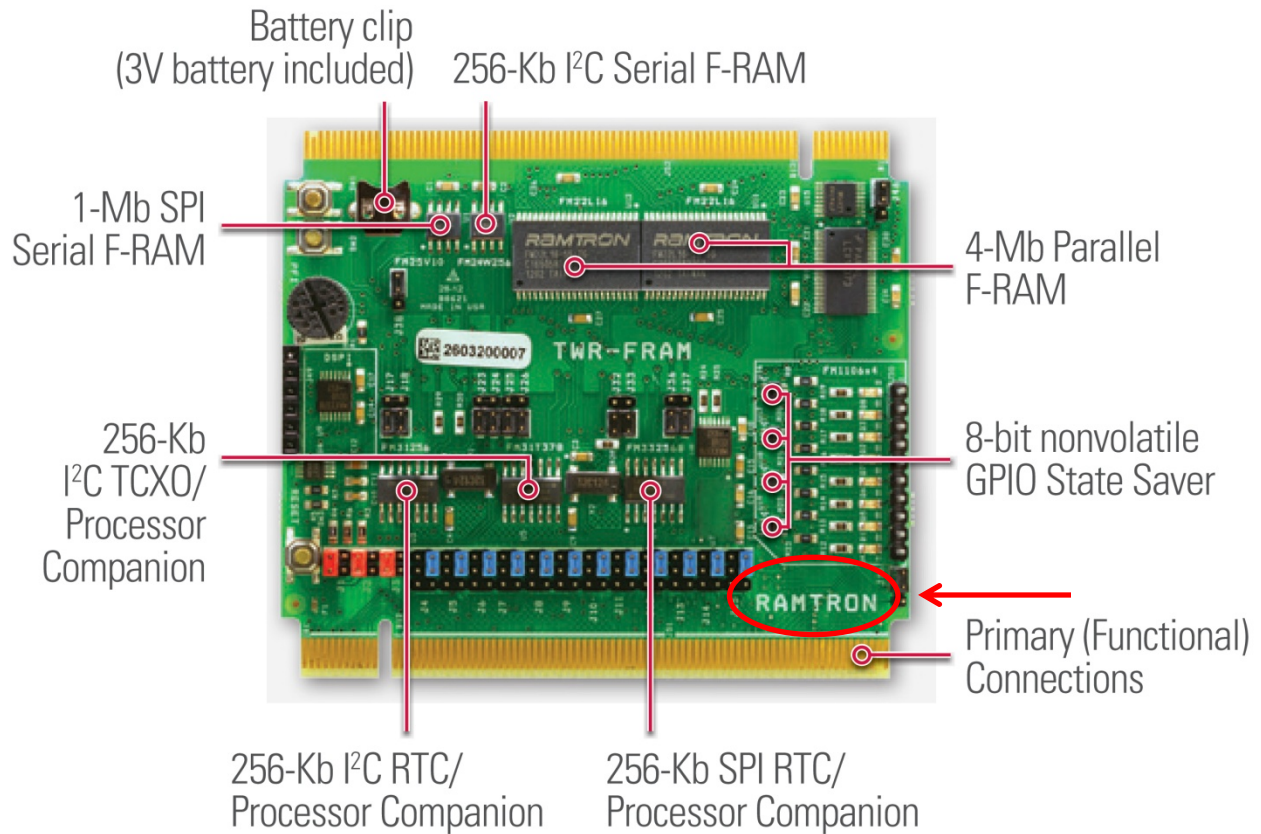
- TWR-FRAM module
- TWR-K53N512 (software developed and tested on this platform)
  - Note: This board is designed to meet the Tower System Module interface so other Freescale Tower MCU modules are compatible for development.
- TWR-ELEV
- miniUSB-to-USB cable (available in TWR-K53N512 kit)

### Software

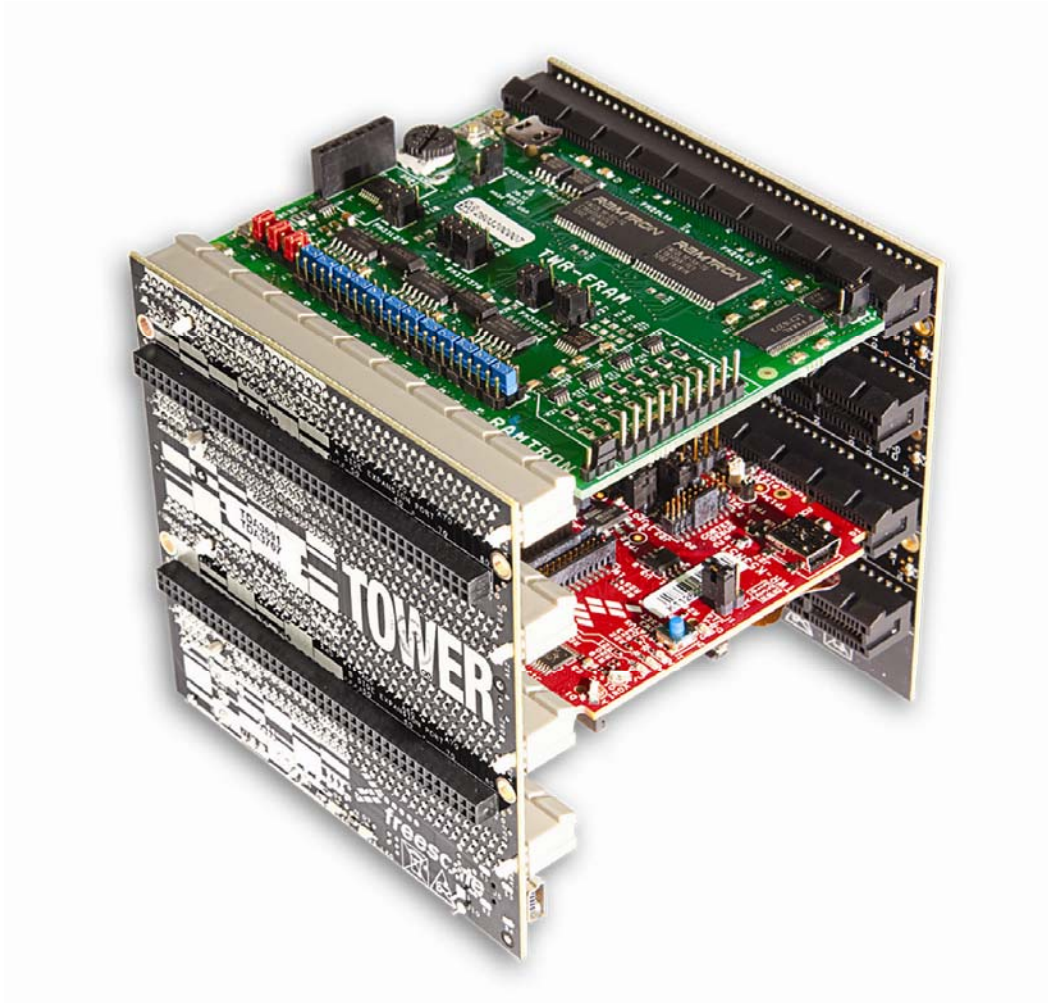
- Windows OS
- IAR Embedded Workbench for ARM
- Keil  $\mu$ Vision4
- PE Micro OSBDM Virtual Serial Toolkit

## 3 Hardware Instructions

After unpacking the TWR-FRAM kit, notice the “RAMTRON” printing on the TWR-FRAM. This is the primary side of the board and should be connected to the Primary Elevator. If the jumpers have not been changed, it is suggested to place the board on the top level of the elevator.



Go ahead and construct the Tower System. Make sure to connect the primary side of all the Tower modules to the primary elevator. The primary side of each board is indicated by the white line along the connector (this can be seen below the RAMTRON label in the picture above). The primary elevator is indicated by white PCI connectors.



Assembled TWR-FRAM Memory Module with Freescale Tower System

## 4 Jumper Instructions and Table

This section provides the jumper table and configuration instructions to interface with demo software.

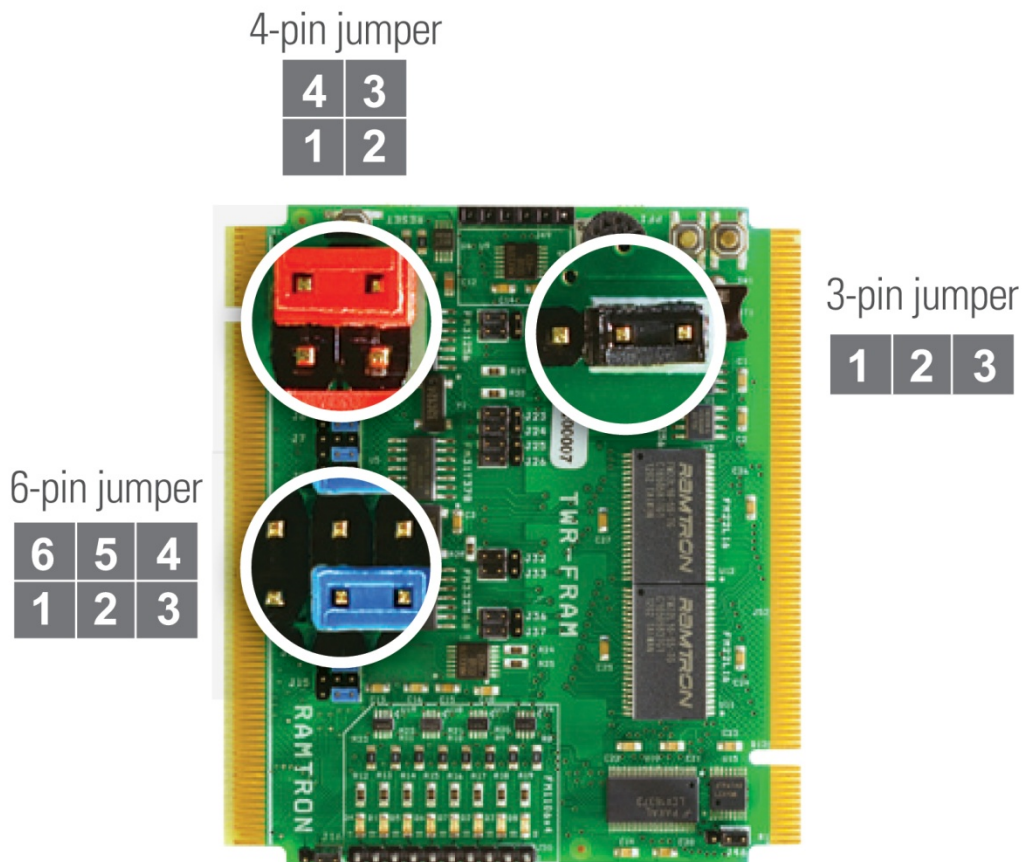
### 4.1 Jumper Instructions

The TWR K53N512 has multiple channels for I2C and SPI communication and there are jumpers on the TWR-FRAM to allow for all possible communication channels. The demo software communicates via I2C0 and SPI0 channels.

For I2C0 (Elevator pins B50 and B51) : Set **J12** and **J13** to **2-3**

For SPI0 (Elevator pins B44, B45, B46 and B48): Set **J4**, **J7**, **J8**, **J10** to **2-3**

To enable FM33256 and disable FM25V10: Set **J32** to **1-2** and **J38** to **2-3**





## 4.2 Jumper Table

Jumper	Option	Type	Setting	Description
J1	DSPI Voltage	4-pin	1-2	DSPI voltage 1.6V
			2-3	DSPI voltage 3.0V
			<b>*3-4*</b>	DSPI voltage 1.8V
J2	VTWR Voltage	4-pin	1-2	VTWR voltage 5V
			<b>*3-4*</b>	VTWR voltage 3.3V
J3	DSPI PWR	4-pin	1-2	/SHDN DSPI via pin B22
			<b>*3-4*</b>	/SHDN DSPI via pin A35
J4	SPI MISO	6-pin	1-2	Connect to pin B11
			<b>*2-3*</b>	Connect to pin B44
			4-5	Connect to pin B56
J5	DSPI D1	6-pin	1-2	Connect to pin B11
			<b>*2-3*</b>	Connect to pin B44
			4-5	Connect to pin B56
J6	DSPI D0	6-pin	1-2	Connect to pin B10
			<b>*2-3*</b>	Connect to pin B45
			4-5	Connect to pin B57
J7	SPI MOSI	6-pin	1-2	Connect to pin B10
			<b>*2-3*</b>	Connect to pin B45
			4-5	Connect to pin B57
J8	SPI SCK	6-pin	1-2	Connect to pin B7
			<b>*2-3*</b>	Connect to pin B48
			4-5	Connect to pin B58
J9	DSPI CLK	6-pin	1-2	Connect to pin B7
			<b>*2-3*</b>	Connect to pin B48
			4-5	Connect to pin B58
J10	SPI CS	6-pin	1-2	Connect to pin B9
			<b>*2-3*</b>	Connect to pin B46
			4-5	Connect to pin B47
			5-6	Connect to pin B55
J11	DSPI CS	6-pin	1-2	Connect to pin B9
			<b>*2-3*</b>	Connect to pin B46
			4-5	Connect to pin B47
			5-6	Connect to pin B55
J12	I2C SCL	6-pin	1-2	Connect to pin A7
			<b>*2-3*</b>	Connect to pin B50

			4-5	Connect to pin B57
J13	I2C SDA	6-pin	1-2	Connect to pin A8
			<b>*2-3*</b>	Connect to pin B51
			4-5	Connect to pin B56
J14	INTERRUPT	6-pin	1-2	Connect to pin B58
			<b>*2-3*</b>	Connect to pin B56
			4-5	Connect to pin B62
J15	RESET	6-pin	1-2	Connect to pin A62
			<b>*2-3*</b>	Connect to pin A47
			4-5	Connect to pin A63
J16	State Saver Enable	3-pin	<b>*1-2*</b>	Connects EN pin to B35
			2-3	Connects EN pin to gnd
J17	FM31L278 RESET	3-pin	1-2	Connects to RESET jumper
			<b>*2-3*</b>	Disconnects pin from RESET jumper
J18	FM31L278 PFO	3-pin	1-2	Connects to INTERRUPT jumper
			<b>*2-3*</b>	Disconnects pin from INTERRUPT jumper
J23	FM31T378 RESET	3-pin	1-2	Connects to RESET jumper
			<b>*2-3*</b>	Disconnects pin from RESET jumper
J24	FM31T378 INT	3-pin	1-2	Connects to INTERRUPT jumper
			<b>*2-3*</b>	Disconnects pin from INTERRUPT jumper
J25	FM31T378 FOUT	3-pin	1-2	Connects to INTERRUPT jumper
			<b>*2-3*</b>	Disconnects pin from INTERRUPT jumper
J26	FM31T378 PFO	3-pin	1-2	Connects to INTERRUPT jumper
			<b>*2-3*</b>	Disconnects pin from INTERRUPT jumper
J32	FM33256B /CS	3-pin	1-2	Disconnects from SPI CS jumper
			<b>*2-3*</b>	Connects pin to SPI CS jumper
J33	FM33256B ACS	3-pin	1-2	Connects to INTERRUPT jumper
			<b>*2-3*</b>	Disconnects pin from INTERRUPT jumper
J36	FM33256B PFO	3-pin	1-2	Connects to INTERRUPT jumper
			<b>*2-3*</b>	Disconnects pin from INTERRUPT jumper
J37	FM33256B RESET	3-pin	<b>*1-2*</b>	Connects to RESET jumper
			2-3	Disconnects pin from RESET jumper
J38	FM25V10 /CS	3-pin	1-2	Connects to SPI CS jumper
			<b>*2-3*</b>	Disconnects pin from SPI CS jumper
J46	FM22L16 SELECT	3-pin	<b>*1-2*</b>	Enables '138 decoder to drive chip enables to FM22L16 devices
			2-3	Disables '138 decoder from driving chip enables to FM22L16 devices

## 5 Software Instructions

### 5.1 IAR Embedded Workbench for ARM

To develop on and access the IAR project demos built for the TWR-K53N512, it is necessary to download IAR Embedded Workbench for ARM.

**Note: Please check that you have the latest version of IAR Embedded Workbench for ARM installed (i.e. these projects were developed on version 6.4 and are confirmed to NOT work on version 6.1).**

Go to IAR home website ([www.iar.com](http://www.iar.com)) under the “Software Download” section and download IAR Embedded Workbench for ARM processors.

Follow the instructions of the install executable.

### 5.2 Keil $\mu$ Vision4

To develop on and access the Keil project demos built for the TWR-K53N512, it is necessary to download Keil  $\mu$ Vision4.

**Note: Please check that you have the latest version Keil  $\mu$ Vision4 for MDK-ARM installed.**

Go to Keil home website ([www.keil.com](http://www.keil.com)) and select “Product Downloads” under the “Software Downloads” section. Download Keil MDK-ARM.

Follow the instructions of the install executable.

You will also need to download and install the Freescale Kinetis OSJTAG drivers for Keil. (<http://www.keil.com/download/docs/408.asp>)

### 5.3 PE Micro OSBDM Virtual Serial Toolkit

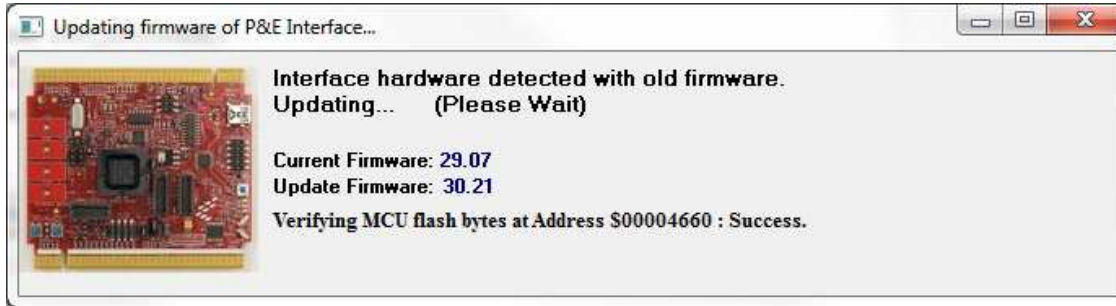
This package is used mainly for the drivers to interface with the OSJTAG and its serial terminal capabilities. A different serial terminal application can be used but it is still necessary to get the appropriate drivers if it is intended to use the OSJTAG.

Go to the PE Micro OSBDM website ([www.pemicro.com/osbdm](http://www.pemicro.com/osbdm)). Download the package by following the “OSBDM Virtual Serial Toolkit” link. Use the pdf installation instructions for the drivers and toolkit by following the link “Installation & Operation of the OSBDM”.

## 5.4 Connect the Tower System to PC

If you have not already, connect the USB cable to the TWR-K53N512 miniUSB port.

If this is the first time you have connected the Tower System with a PC containing updated P&E drivers, the PC may indicate that the firmware needs to be updated (Follow the instructions for the OSBDM Virtual Serial Toolkit if these have not been installed and updated).



Once the system is connected, the onboard demo should be running.

Go to Start -> All Programs -> P&E OSBDM OSJTAG Virtual Serial Toolkit ->Utilities->Terminal Utility

Make sure the settings are:

Port: <appropriate COM> ("USB COM" works on this PC)

Baud: 115200

Parity: None

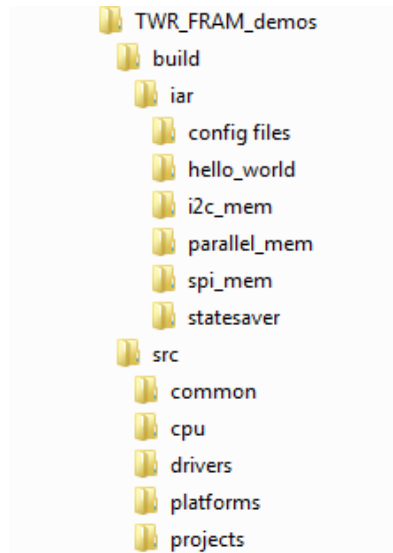
Bits: 8

Click "Open Serial Port" and text should appear below the button saying "Open."

At this point, everything necessary to interface with the demos should be available.

## 6 Demo Instructions for IAR

The demo software can be downloaded at [www.ramtron/go/towerFRAM.com](http://www.ramtron/go/towerFRAM.com) . Unzip “TWR\_FRAM\_demos”. The directory structure is as follows:



Inside the build folder, this contains the IAR workspaces for each project. The src folder contains the source files for the projects.

### TWR-FRAM Project Descriptions:

**SPI\_mem** – provides example code for interfacing with the F-RAM SPI devices FM33256B and FM25V10. For the processor companion part (FM33256B), the code will activate the RTC, set the time and date, read from the event counters, and write data to the memory.

NOTE: The SPI\_mem project can only interact with one SPI device at a time. By default, the program in interface with the processor companion (FM33256B). To interface with the FM25V10, comment the define for the FM33256B and uncomment the define for FM25V10 (like shown below). Then, recompile the project.

```
#include "k53_spi.h"

//available devices
//define FM33256B
#define FM25V10
```

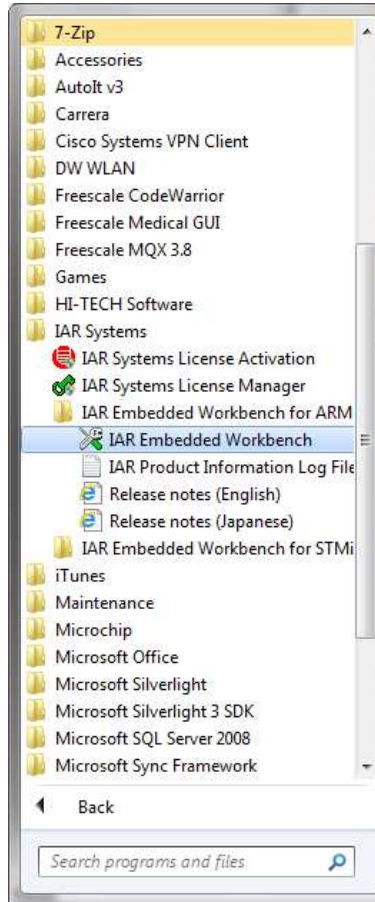
**I2C\_mem** – provides example code for interfacing with the F-RAM I2C devices FM31T378, FM31256, and FM24W256. For the processor companion parts (FM31T378 and FM31256), the code will activate the RTC, set the time and date, read from the event counters, and write data to the memory of all 3 components.

**statesaver** – provides example code of interfacing with the FM1106 devices. The state savers are connected to LEDs and the project will light a pattern by controlling the FM1106.

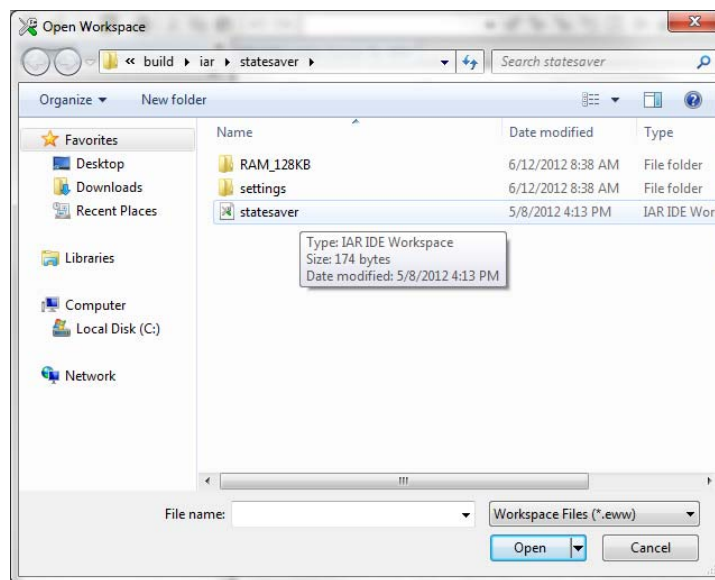
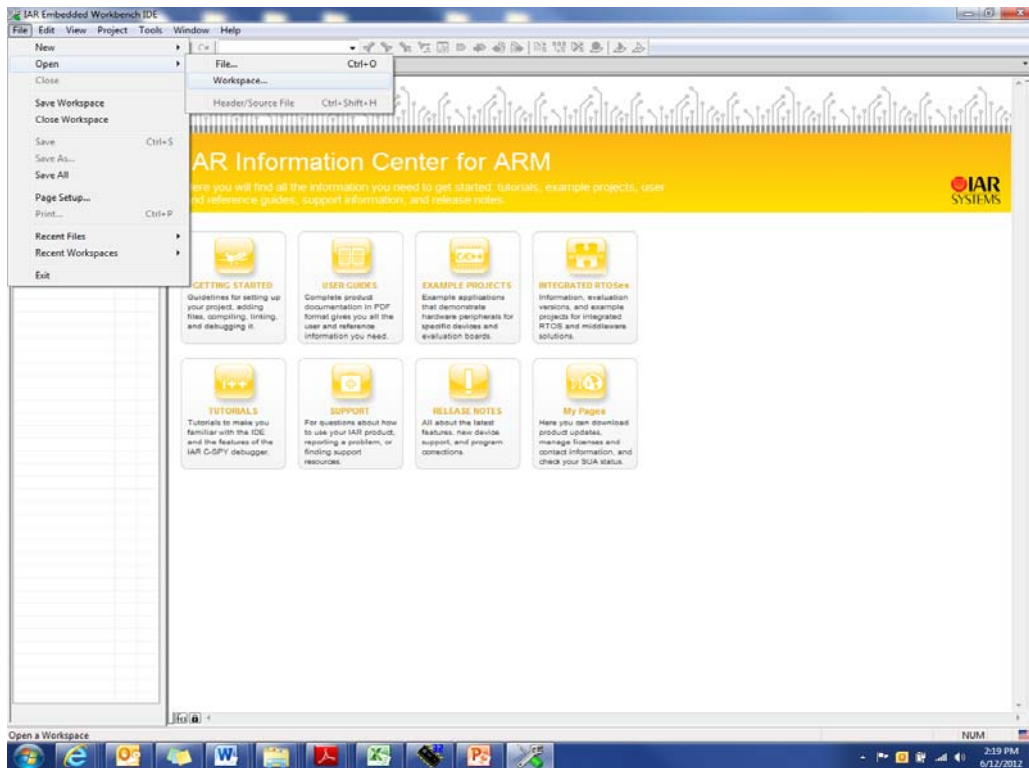
**parallel\_mem** – provides example code of interfacing with the FM22L16 devices.

## 6.1 Running the demo

1. Start IAR
  - a. Start -> All Programs -> IAR Systems -> IAR Embedded Workbench for ARM -> IAR Embedded Workbench

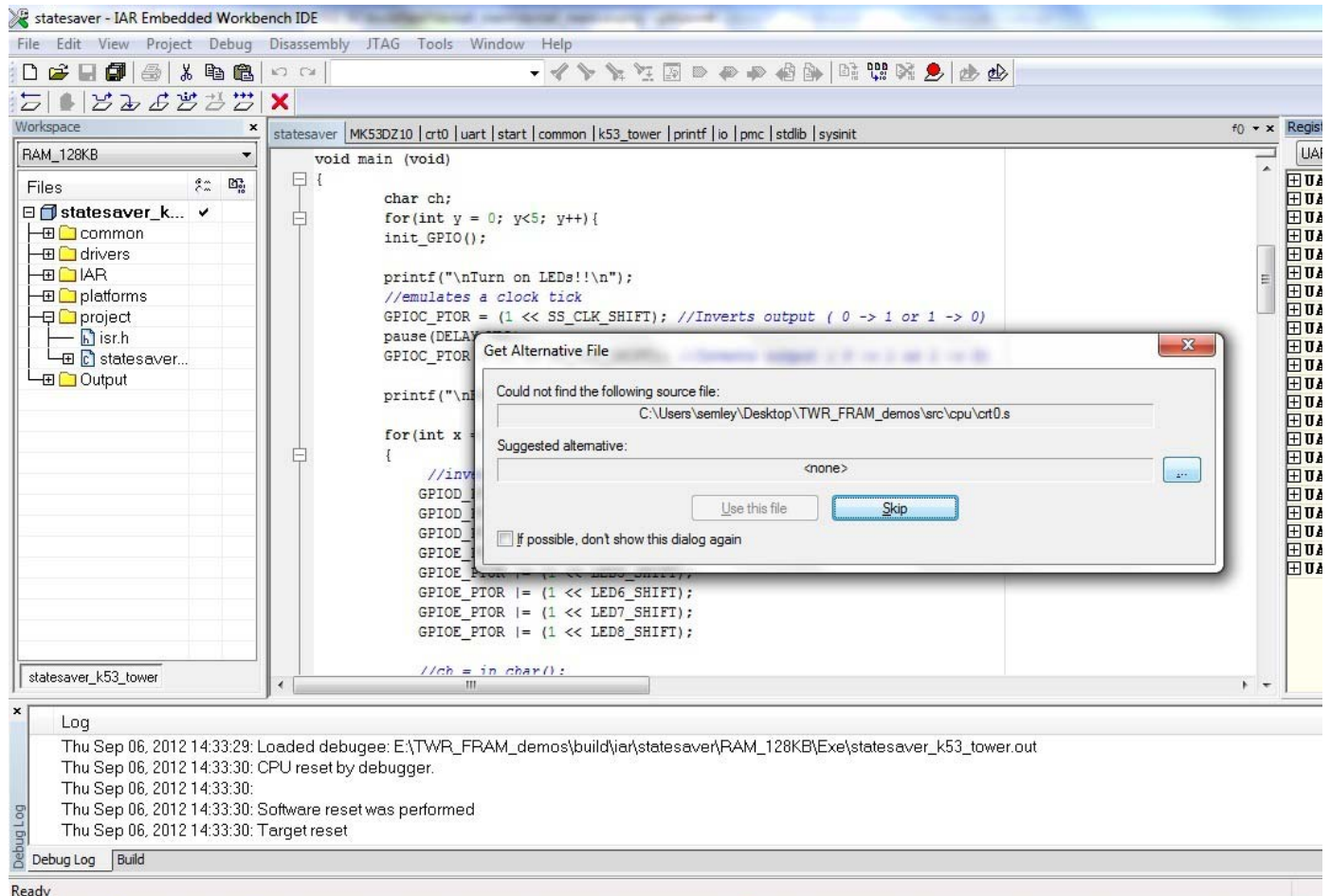


2. Open the demo workspace
  - a. File -> Open -> Workspace...
  - b. Go to the desired project under {unzip directory}/TWR\_FRAM\_demos/build/iar/
  - c. Open the IAR IDE Workspace in the project folder



3. Rebuild and Execute in debugger
  - a. Select: Project-> Rebuild All
  - b. Select: Project -> Download and Debug
    - i. IAR may ask for a changed reference for the "crt0.s" assembly file. Direct it towards the crt0.s file located:
 

```
{unzip directory}/TWR_FRAM_demos/src/cpu/crt0.s
```



- c. Select: Debug -> Go
- d. Select: Debug -> Stop Debugging (when ready to quit Debug mode)



## 6.2 Creating a New Project

There is an executable called “make\_new\_project” located in:

```
{unzip directory}/TWR_FRAM_demos/build/iar/
```

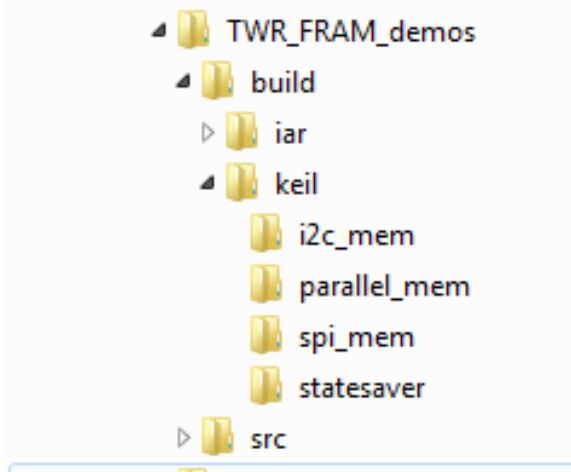
Run the executable and a terminal opens asking for the name of the new project.



Enter the new project name and you will see the directory appear in the build and src folders.

## 7 Demo Instructions for Keil

The demo software can be downloaded at [www.ramtron/go/towerFRAM.com](http://www.ramtron/go/towerFRAM.com) . Unzip “TWR\_FRAM\_demos”. The directory structure is as follows:



Inside the build folder, this contains the Keil workspaces for each project. The src files are contained in the build folder for each project as well.

### TWR-FRAM Project Descriptions:

**SPI\_mem** – provides example code for interfacing with the F-RAM SPI devices FM33256B and FM25V10. For the processor companion part (FM33256B), the code will activate the RTC, set the time and date, read from the event counters, and write data to the memory.

NOTE: The SPI\_mem project can only interact with one SPI device at a time. By default, the program in interface with the processor companion (FM33256B). To interface with the FM25V10, comment the define for the FM33256B and uncomment the define for FM25V10 (like shown below). Then, recompile the project.

```
#include "k53_spi.h"
//available devices
//define FM33256B
#define FM25V10
```

**I2C\_mem** – provides example code for interfacing with the F-RAM I2C devices FM31T378, FM31256, and FM24W256. For the processor companion parts (FM31T378 and FM31256), the code will activate the RTC, set the time and date, read from the event counters, and write data to the memory of all 3 components.

**statesaver** – provides example code of interfacing with the FM1106 devices. The statesavers are connected to LEDs and the project will light a pattern by controlling the FM1106.

**parallel\_mem** – provides example code of interfacing with the FM22L16 devices.

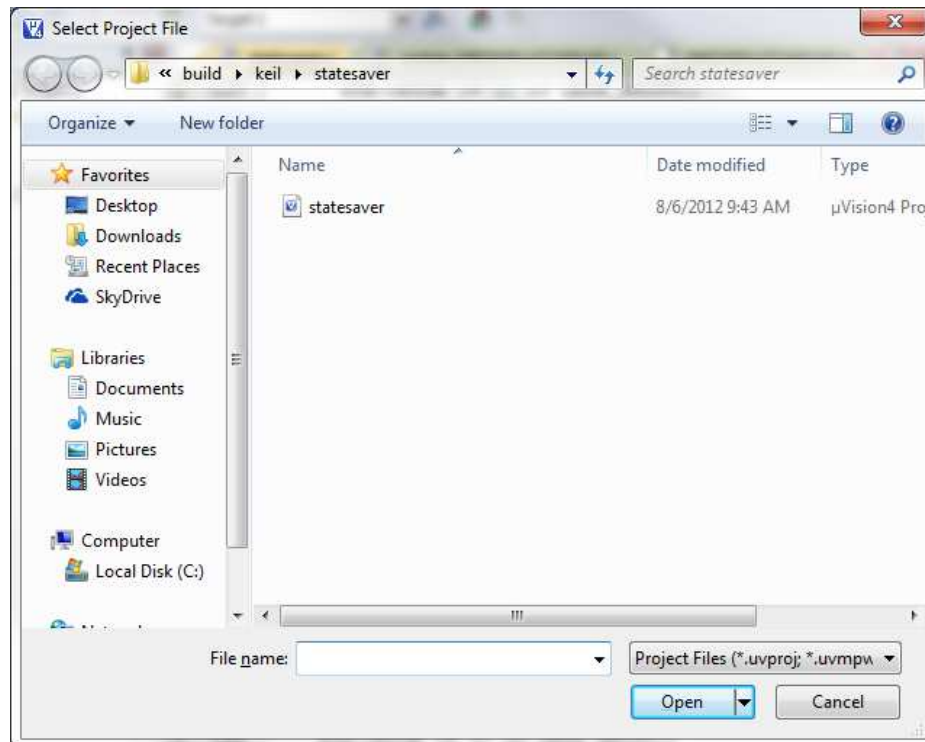
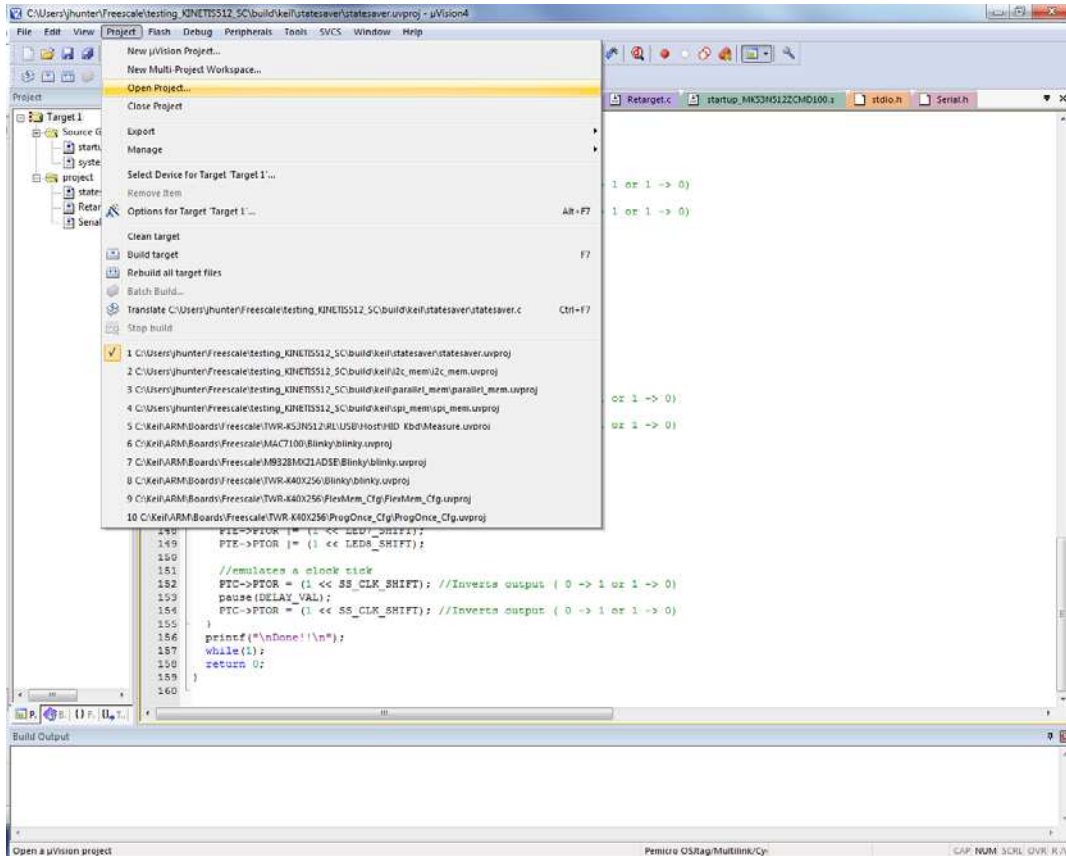
### 7.1 Running the demo

1. Start Keil

- a. Start -> All Programs -> Keil  $\mu$ Vision4

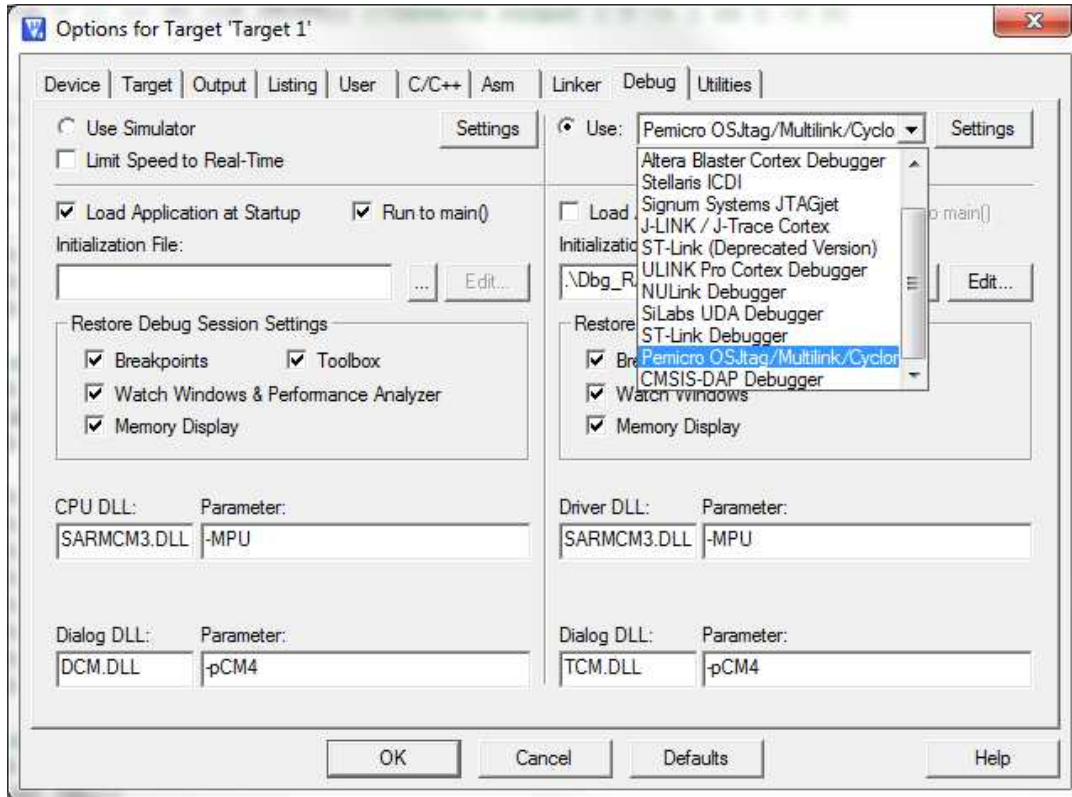


2. Open the demo workspace
  - a. Project -> Open Project
  - b. Go to the desired project under {unzip directory}/TWR\_FRAM\_demos/build/keil/
  - c. Open the  $\mu$ Vision4 Project in the project folder



3. Make that the debugger is configured for Pemicro OSJtag/Multilink/Cyclone

- a. Select: Project -> Options for Target
- b. Click Debug Tab
- c. Select “Pemicro OSJtag/Multilink/Cyclone” on the right and select “Use:” radio button
- d. Click OK



4. Rebuild and Execute in debugger
  - a. Select: Project-> Rebuild all target files
  - b. Select: Debug -> Start/Stop Debug Session
  - c. Select: Debug -> Run
  - d. Select: Debug -> Stop Debugging (when ready to quit Debug mode)