

## ETAS INCA V7.5

Serial (X)ETK Calibration Concepts  
Limited Emulation RAM



User Guide

## Copyright

The data in this document may not be altered or amended without special notification from ETAS GmbH. ETAS GmbH undertakes no further obligation in relation to this document. The software described in it can only be used if the customer is in possession of a general license agreement or single license. Using and copying is only allowed in concurrence with the specifications stipulated in the contract.

Under no circumstances may any part of this document be copied, reproduced, transmitted, stored in a retrieval system or translated into another language without the express written permission of ETAS GmbH.

**© Copyright 2024** ETAS GmbH, Stuttgart

The names and designations used in this document are trademarks or brands belonging to the respective owners.

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See [mathworks.com/trademarks](https://mathworks.com/trademarks) for a list of additional trademarks.

INCA V7.5 | User Guide R01 EN | 03.2024

## Contents

1	Introduction .....	5
1.1	Intended Use .....	5
1.2	Target Group .....	5
1.3	Classification of Safety Messages .....	6
1.4	Safety Information .....	6
1.5	Data Protection .....	7
1.6	Data and Information Security .....	7
2	About Serial (X)ETK Calibration .....	8
2.1	System Requirements .....	8
2.2	Definitions and Abbreviations .....	9
3	Basic Concepts .....	10
3.1	What is Calibration? .....	10
3.2	What is Emulation RAM? .....	10
3.3	Emulation with Manual Preparation (Old Concept) .....	11
3.3.1	Auto-configuration of the Emulation RAM through INCA .....	12
3.3.2	Experiment Preparation by Interacting with the Operator .....	13
3.3.3	Programming Back Calibrated Data upon Switching of Emulated Area .....	14
3.4	Dynamic Emulation with Fixed Block Sizes in Emulation RAM .....	15
3.5	Dynamic Emulation with Reconfigurable Block Sizes in Emulation RAM .....	17
3.6	Summary: Comparison of the Concepts .....	18
4	Using Limited Emulation with Experiment Preparation for Serial (X)ETKs .....	20
4.1	Setting the User Option for Emulation with Manual Preparation of the Experiment ...	20
4.2	Preparing the Workspace .....	20
4.3	Working on the Experiment .....	21
4.3.1	Initialization .....	21
4.3.2	Limitations on the Operation of the Memory Pages .....	21
4.3.3	Settings for the Emulation RAM Area .....	22
4.3.4	Displaying the Emulation RAM Configuration .....	26
5	Using Dynamic Emulation for Serial (X)ETKs .....	28
5.1	Setting the User Options for Dynamic Emulation .....	28
5.2	Preparing the Workspace .....	29
5.3	Working with the Experiment .....	30
5.3.1	Initialization .....	30

5.3.2	Resolving Differences by Means of Memory Page Operations .....	30
5.3.3	Calibrating Variables in the Experiment Environment using Dynamic Emulation .....	32
5.3.4	Displaying the Emulation RAM Configuration .....	34
6	Restrictions .....	36
6.1	General Restrictions .....	36
6.2	Restrictions using Limited Emulation with Experiment Preparation .....	36
6.3	Restrictions using Dynamic Emulation .....	36
7	A2L Examples .....	38
7.1	A2L Example using FixedSizeMoveableEmuRAM .....	38
7.2	A2L Example using FixedSizeNonMoveableEmuRAM .....	39
7.3	A2L Example using FlashRangeSpecificEmuRAM .....	40
7.4	A2L Example using ReconfigurableSizeMoveableEmuRAM .....	41
8	Tips and Tricks .....	43
9	Contact Information .....	45

# 1 Introduction

## 1.1 Intended Use

The Limited Emulation RAM functionality is supported by the INCA measurement and calibration tool. It manages the emulation RAM of the microcontroller in order to enable calibration of ROM variables with Working and Reference Page concept.

This document, which supplements the "INCA Getting Started" manual, gives an overview of the concepts employed for using emulation RAM, as well as how to set them up and use them with INCA.

INCA and INCA add-ons are developed and approved for automotive applications and procedures as described in the user documentation for INCA and INCA add-ons.

INCA and the INCA add-ons are intended to be used in industrial labs and in test vehicles.

ETAS GmbH cannot be made liable for damage that is caused by incorrect use and not adhering to the safety information.

## 1.2 Target Group

This software product and this user guide address qualified personnel working in the fields of automotive ECU development and calibration, as well as system administrators and users with administrator privileges who install, maintain, or uninstall software. Specialized knowledge in the areas of measurement and ECU technology is required.

Users should be familiar with general operations of INCA.

## 1.3 Classification of Safety Messages

Safety messages warn of dangers that can lead to personal injury or damage to property:



### **DANGER**

**DANGER** indicates a hazardous situation that, if not avoided, will result in death or serious injury.



### **WARNING**

**WARNING** indicates a hazardous situation that, if not avoided, could result in death or serious injury.



### **CAUTION**

**CAUTION** indicates a hazardous situation that, if not avoided, could result in minor or moderate injury.

### **NOTICE**

**NOTICE** indicates a situation that, if not avoided, could result in damage to property.

## 1.4 Safety Information



### **WARNING**

#### **Risk of unexpected vehicle behavior**

Calibration activities influence the behavior of the ECU and the systems that are connected to the ECU.

This can lead to unexpected vehicle behavior, such as engine shutdown as well as breaking, accelerating, or swerving of the vehicle.

Only perform calibration activities if you are trained in using the product and can assess the possible reactions of the connected systems.



## WARNING

### **Risk of unexpected vehicle behavior**

Sending messages via bus systems, such as CAN, LIN, FlexRay, or Ethernet, influences the behavior of the systems connected to it.

This can lead to unexpected vehicle behavior, such as engine shutdown as well as breaking, accelerating, or swerving of the vehicle.

Only perform the sending of messages via a bus system if you have sufficient knowledge in using the respective bus system and can assess the possible reactions of the connected systems.

Adhere to the instructions in the ETAS Safety Advice and the safety information given in the online help and user guides. Open the ETAS Safety Advice in the INCA help menu ? > **Safety Advice**.

## 1.5 Data Protection

If the product contains functions that process personal data, legal requirements of data protection and data privacy laws shall be complied with by the customer. As the data controller, the customer usually designs subsequent processing. Therefore, he must check, if the protective measures are sufficient.

## 1.6 Data and Information Security

To securely handle data in the context of this product, see the INCA Help section "Data and Information Security".

## 2 About Serial (X)ETK Calibration

Nowadays many microcontrollers are provided with internal memories, sometimes with a so-called emulation RAM function. These controllers allow to overlay areas of the ROM with parts of the RAM. The overlay areas of the RAM and the ROM are defined in the control registers of the ECU. Serial (X)ETKs use this mechanism to realize the Working Page and Reference Page concept on this type of controllers.

Those ECUs only have a limited number of blocks with fixed sizes, which makes it necessary for the user to select the variables to be calibrated. Until INCA V5.3, the handling of emulation RAM was offered only in a static way. This means that a preparation step was necessary before calibration could be started.

In order to make the handling a lot easier, INCA V5.4 has introduced the dynamic emulation of memory blocks. As long as memory blocks are still available, the user can add as many calibration variables to the experiment as he wants. Variables that remain unchanged are not emulated. Whenever a variable is modified, the emulation is modified automatically. If all sections of the emulation RAM are used up, the not emulated variables get write-protected by INCA. The user can free the occupied memory by flashing the INCA Working Page to the ECU-ROM.

With INCA V7.1, dynamic emulation has been further improved by using reconfigurable blocks of different sizes in the emulation RAM. This method optimizes the usage of the emulation RAM so that the operator may calibrate more variables in an equally-sized project.

The new approaches of dynamic emulation are not only easy-to-use, they are especially useful at testbenches, where it is not possible to control additional configuration windows required in the old approach for selecting emulation segments for variables.

The current INCA version supports both the old and the new concept. Whether the original method with manual experiment preparation or dynamic emulation is to be used can be set in the INCA user options.

If the user option is set to dynamic emulation, it depends on the definition of the moveable emulation RAM calibration method in the A2L file and the driver version of the (X)ETK device whether fixed-size memory segments or reconfigurable-size memory segments are used.

### 2.1 System Requirements

The Limited Emulation RAM functionality gets automatically installed with INCA. For information on the INCA system requirements and installation procedure refer to the INCA Installation Guide.

Using the Limited Emulation RAM functionality requires the latest HSP version (Hardware Service Package) for the interface hardware; the latest HSP versions can be downloaded on the ETAS web pages at [http://www.etas.com/en/products/download\\_center.php](http://www.etas.com/en/products/download_center.php).



Which of the moveable emulation RAM calibration methods that are supported by INCA is actually supported by the (X)ETK device depends on its type and driver version. Moreover, whether this function is supported depends on type and driver version of the (X)ETK device. If you need a driver update, please contact your local ETAS support.

## 2.2 Definitions and Abbreviations

### *Definitions*

#### **Emulation**

The fixed values of the parameters are replaced by variable parameter values

#### **Emulation RAM**

RAM that is used to replace (overlay) the FLASH memory in order to make the parameter values changeable

#### **Dynamic Emulation**

Emulation mode where the available emulation RAM is used on demand

#### **Emulation with Manual Preparation / Limited Emulation with Experiment Preparation**

Emulation mode where the available emulation RAM is assigned manually (before the usage) to emulate a dedicated set of parameter values

#### **Project Full Emulation**

The available emulation RAM is sufficient to cover all parameters from the DATA and OFFLINE\_DATA memory segments that are described in the A2L file.

#### **Experiment Full Emulation**

The available emulation RAM is sufficient to cover all parameter values used in the given experiment (but not necessarily the complete project). INCA tries to establish this state when opening an experiment. The emulation RAM is not updated when calibration variables are added or removed.

### *Abbreviations*

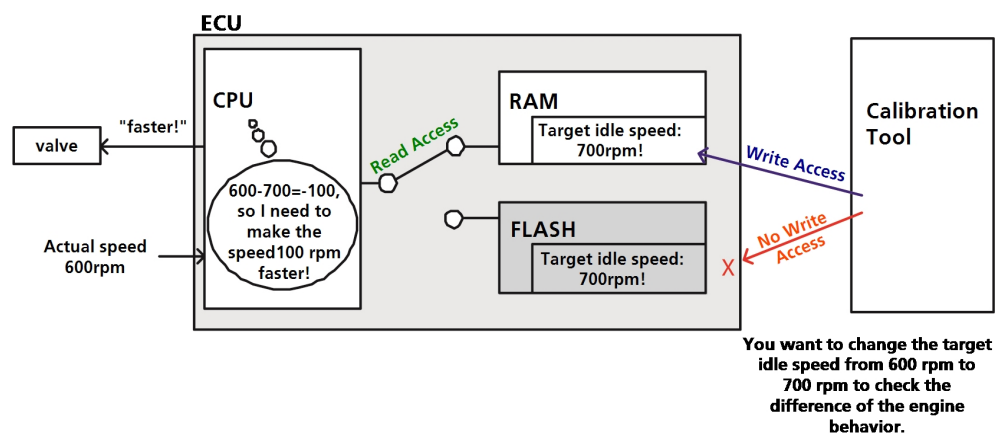
<b>ECU</b>	Electronic Control Unit
<b>HDC</b>	Hardware Definition Code ((X)ETK Firmware)
<b>HSP</b>	Hardware Service Pack (Firmware for ETAS hardware)
<b>RAM</b>	Random Access Memory (Flash Memory)
<b>ROM</b>	Read Only Memory (Flash Memory)
<b>PR</b>	Reference Page
<b>WP</b>	Working Page

## 3 Basic Concepts

INCA's **Limited Emulation RAM** functionality manages the **Emulation RAM** of the ECU in order to enable calibration of ROM variables using the Working and Reference Page concept. This chapter explains the concepts of calibration, RAM emulation and the management of Limited Emulation RAM in INCA.

### 3.1 What is Calibration?

Calibration means changing the values of the control parameters (also called "calibration variables") of the ECU online while the control program is running on it, to make the behavior of the ECU closer to the ideal status. To do this, the data area which includes the control parameters needs to be located on a RAM.



There are two types of memories on which calibration variables can be located:

- RAM (**R**andom **A**ccess **M**emory)
  - Both reading and writing are possible.
- Flash EPROM (Flash memory, ROM)

In Flash EPROMs it is possible to erase selected memory blocks or the content of the complete memory electrically by means of so-called flashes, and to re-program the memory afterwards.

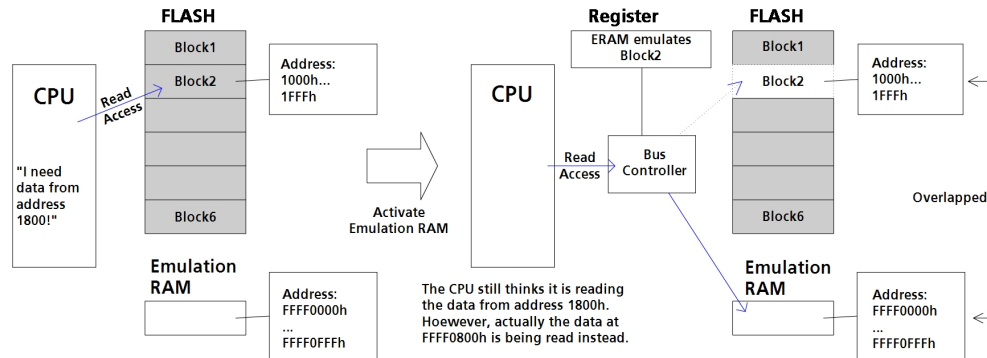
Re-programming of the Flash EPROM is carried out with a programming tool. The advantage of the Flash technology lies within the fact that it is possible to re-program the Flash memory with a programming tool even in ECUs that are mounted in the vehicle. Therefore Flash technology is applied in such cases where comparatively large data volumes have to be stored permanently but need to be modified during the product life cycle, e.g. in the program and data memory of an ECU.

### 3.2 What is Emulation RAM?

Emulation RAM is RAM that is used to replace (emulate) Flash memory with the intention to change the data stored in the Flash.

During the last years, many microcontrollers have been equipped with internal memories. Many of those microcontrollers have **Emulation RAM**, which is used by serial (X)ETKs to emulate the values in the Flash memory.

This section gives an overview of the Emulation RAM function.



In the normal state, the control program on the ECU runs using the parameter values in its flash memory. When the Emulation RAM function is activated, a specific area within the internal RAM is used as the emulation RAM, and this area emulates a part of the flash memory which is called a block. Activation and deactivation of the emulation RAM is controlled by the control register(s) in the ECU.

For typical microcontrollers, the emulation RAM is also divided into several blocks so that several blocks within the flash memory can be emulated at a time.

The number of blocks and their sizes, as well as the address ranges of the flash memory that can be emulated, depend on the type of the microcontroller. The microcontrollers supported by each (X)ETK are listed in the system requirements which can be found on the ETAS web pages at <http://www.etas.com>.

When the calibration is performed using INCA, the internal flash memory is used for the Reference Page (RP) and the internal emulation RAM is used for the Working Page (WP). To set this configuration, INCA (with (X)ETK) sets the control register(s) for the emulation RAM via the debug port of the microcontroller. The number and usage of the control register(s) depend on the microcontroller.

### 3.3 Emulation with Manual Preparation (Old Concept)

Whenever an experiment is opened or hardware is explicitly initialized, INCA has to make sure that the calibration variables can be emulated in the emulation RAM. The underlying process consists of the following basic steps:

1. INCA checks whether the relevant variables can all be emulated at once and tries to configure the emulation RAM. Everything that can be done automatically gets done without the interaction of the user.
2. If there are conflicts that cannot be resolved automatically, the INCA operator is prompted to decide which variables should be selected for calibration. This step is repeated until all selected variables can be emulated.
3. INCA makes the settings valid at the ECU.

After the experiment preparation, the user can work in the Experiment Environment as usual.



### Note

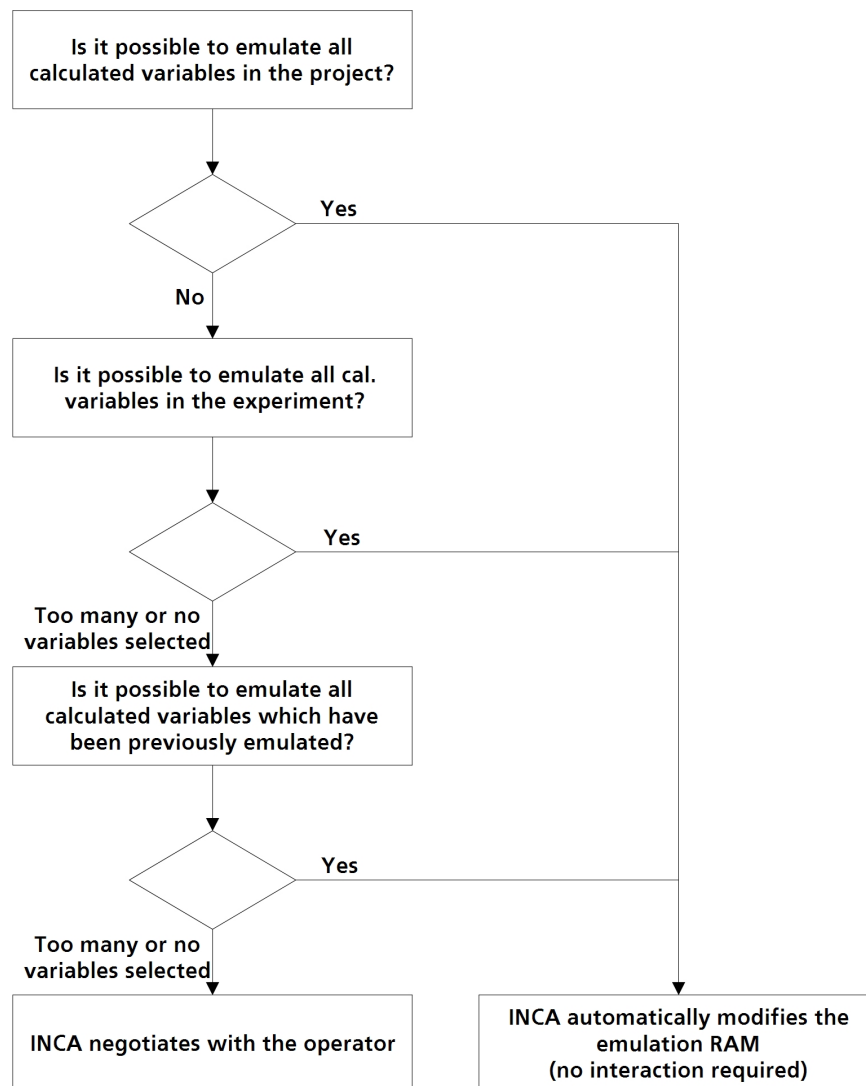
To be able to use emulation RAM with the manual preparation method, the A2L file must contain the CALIBRATION\_METHOD

`FixedSizeMoveableEmuRAM`. For details see "[A2L Examples](#)" on page 38.

Moreover, whether this function is supported depends on type and driver version of the (X)ETK device. If you need a driver update, please contact your local ETAS support.

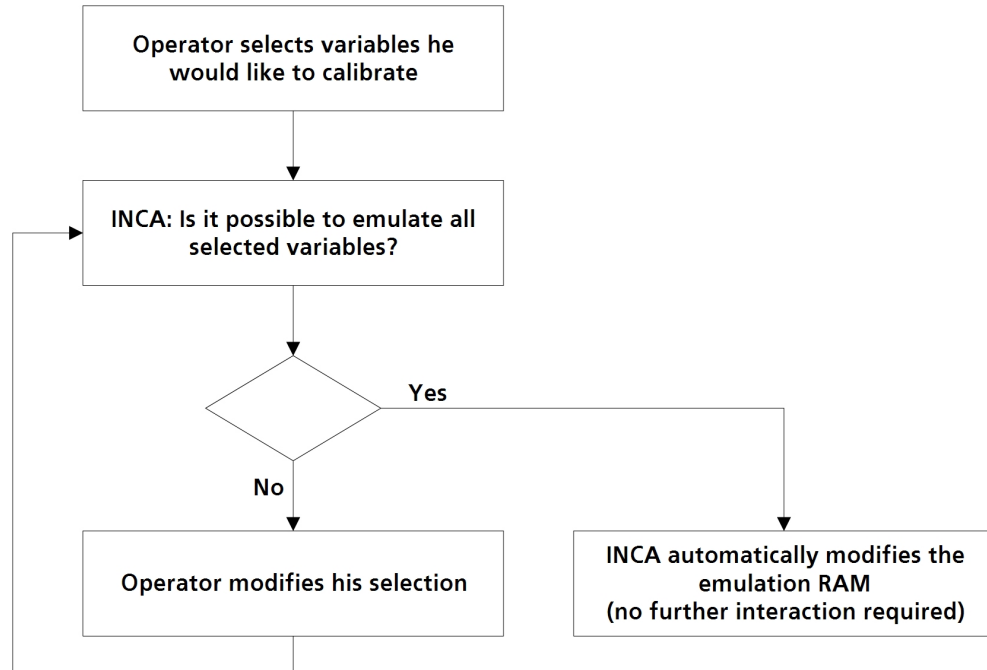
## 3.3.1 Auto-configuration of the Emulation RAM through INCA

The INCA operator starts an experiment or explicitly initializes the hardware. If possible, INCA tries to configure the emulation RAM automatically (auto-configuration). Thereby it follows the logical flow shown below:



### 3.3.2 Experiment Preparation by Interacting with the Operator

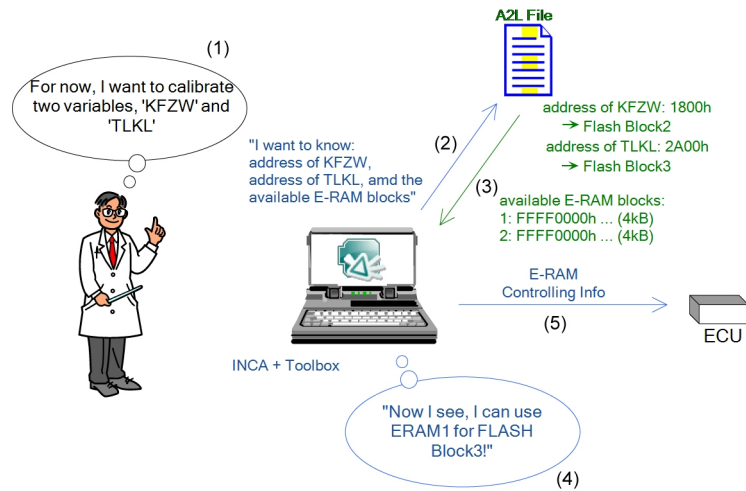
If the size of the flash memory areas which include the variables to be calibrated is larger than the size of emulation RAM areas available, it is not possible to emulate all variables selected by the operator at once. In this case, INCA displays a warning message and prompts the operator to decrease the number of the variables so that it can properly set the control information onto the ECU.



#### Example 1: Emulating All Selected Variables, No Modification of Selection Required

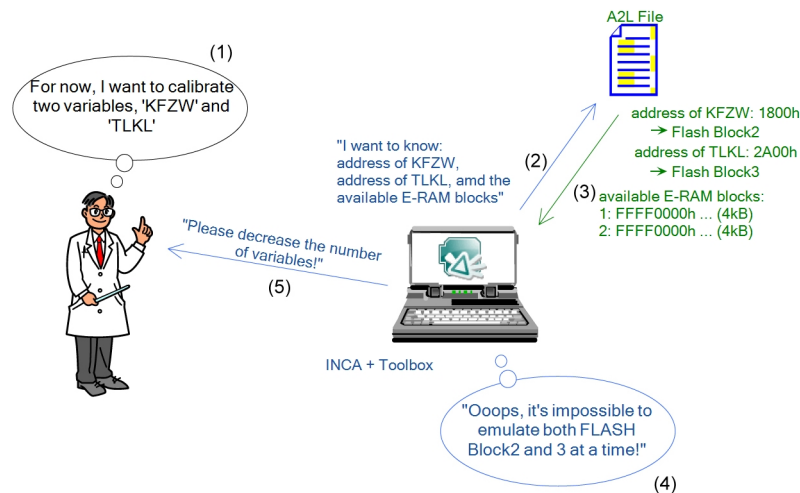
The INCA operator opens an experiment which contains some calibration variables. The Limited Emulation RAM functionality reads the addresses for those variables and the emulation RAM information from the ASAM-2MC file, and sets the appropriate value(s) onto the control register(s) of the microcontroller in the ECU.

Note that the first auto-configuration step which checks whether all variables in the project can be emulated at once is skipped in this illustration.



An explanation of the relationship between block sizes and addresses is given in "A2L Examples" on page 38.

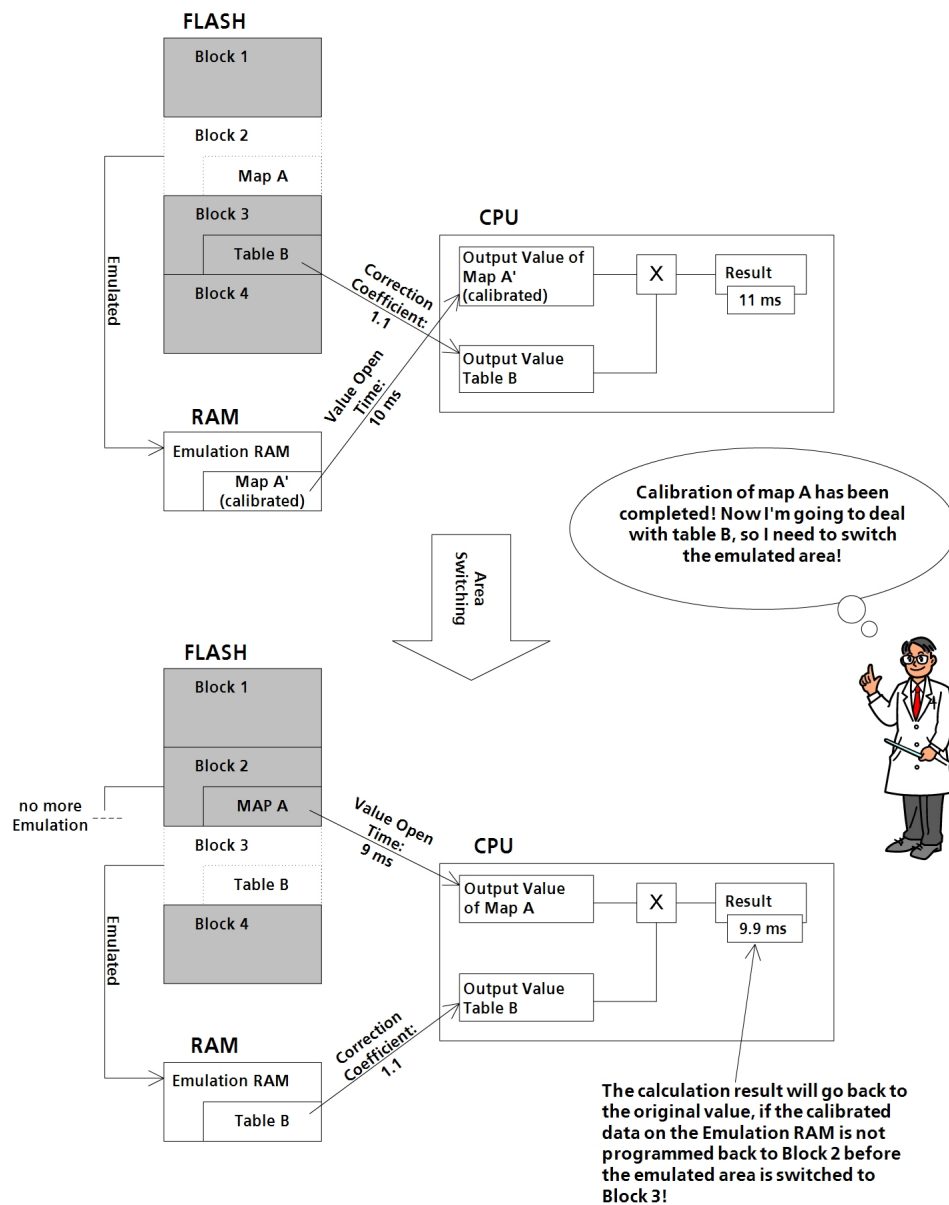
### Example 2: Negotiating with the Operator in Case of the Shortage of Emulation RAM



### 3.3.3 Programming Back Calibrated Data upon Switching of Emulated Area

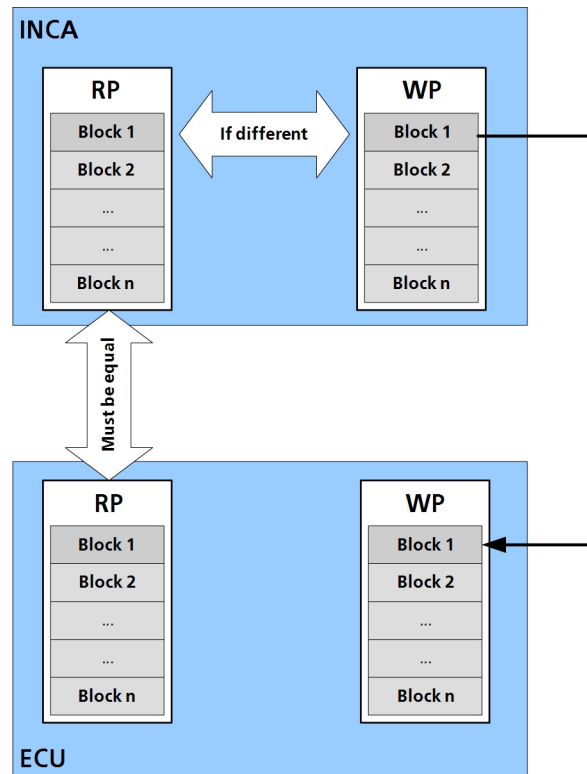
If the emulated flash memory area is changed so that another block of the flash memory can be calibrated, all data which have been calibrated on the emulation RAM are lost, because the data on the newly selected block of the flash memory will be copied into the emulation RAM. If the user wants to keep those calibrated data so that he can continuously use them also for further calibration work, it is possible to write them back to the original flash memory block before the emulated area is switched.

In this case, INCA will prompt the operator to program back by flashing to the Reference Page, as shown in the figure below.



### 3.4 Dynamic Emulation with Fixed Block Sizes in Emulation RAM

With the concept of dynamic emulation with fixed block sizes, INCA determines the differences between Reference Page and Working Page on the PC when the operator opens an experiment. Depending on the emulation resources defined in the A2L file, INCA uses a set of memory RAM blocks; their sizes match the corresponding sizes defined in the A2L file and can not be changed. Only those variables that are changed in INCA (compared to the Reference Page) get emulated. Every time a variable is changed, the emulation is automatically adapted using one or more blocks from the initial setup.



### Note

This function is only available if the A2L file contains the `CALIBRATION_METHOD FixedSizeMovableEmuRAM`.

For details see "[A2L Examples](#)" on page 38.

Moreover, whether this function is supported depends on type and driver version of the (X)ETK device. If you need a driver update, please contact your local ETAS support.

This leads to the fact that not only a dedicated subset of variables, but **all** variables are available for calibration. Whenever a variable is changed, INCA checks whether it is already emulated. If not INCA tries to emulate it.

When the remaining variables can not be emulated in the last remaining blocks, all calibration variables that can no longer be emulated get write-protected.

The operator can now use the Memory Page Manager to flash the Working Page to the Reference Page, thereby freeing the emulation RAM for the emulation of further variables to be changed.



### Note

If Dynamic Emulation is selected in the user options and if there is enough emulation RAM available to emulate all memory segments from the A2L file with the program types DATA and OFFLINE\_DATA at once, INCA will activate a special "Project Full Emulation" mode. In Project Full Emulation mode, INCA automatically configures the emulation RAM to emulate all DATA and OFFLINE\_DATA memory segments and thus all calibration variables. No further reconfiguration is necessary when calibration variables are modified, and no calibration variables become write-protected during calibration.

Only if Project Full Emulation is active, it is possible to use an interpolation when calibrating group axes. If Project Full Emulation is not possible, group axes are write-protected if an interpolation is configured.

## 3.5 Dynamic Emulation with Reconfigurable Block Sizes in Emulation RAM

This concept is similar to dynamic emulation with fixed block sizes. However, it further optimizes the usage of the emulation RAM by using blocks of different and reconfigurable sizes in the emulation RAM. As a result, the operator may calibrate more variables in an equally-sized project.

### Note

This function is only available if the A2L file contains the CALIBRATION\_METHOD ReconfigurableSizeMovableEmuRAM.

For details see "[A2L Examples](#)" on page 38.

Moreover, whether this function is supported depends on type and driver version of the (X)ETK device. If you need a driver update, please contact your local ETAS support.

The emulation behavior when opening an experiment is done in steps. At first INCA tries to do a full emulation of all calibrations of the complete project. If this is not possible INCA behaves dependent upon whether **experiment full emulation** of the experiment has been enabled in the user options. With **experiment full emulation = yes**, INCA tries to emulate all calibrations used in the experiment at once. This prohibits time consuming emulation while calibration in the experiment. If **experiment full emulation** is not possible INCA tries to emulate the calibrations that have different values on working and reference page.

The behavior when opening an experiment depends upon whether full emulation of the experiment has been enabled in the user options. With full emulation, INCA tries to emulate all calibrations at once. This is useful if all needed calibrations in the experiment are already configured.

If the operator would like to add further calibrations after opening the experiment, full emulation should be disabled. In this case, when opening the experiment, INCA determines the differences between Reference Page and Working Page on the PC. Based on the emulation resources defined in the A2L file, INCA then calculates a set of memory blocks with different sizes which are used as initial setup. Only those variables that are changed (compared to the Reference Page) get emulated. Every time a variable is changed, the emulation is automatically adapted using one or more blocks from the initial setup.

The operator goes on changing the values of calibration variables. When INCA runs out of free emulation RAM blocks, the variables that are currently not emulated get write-protected. Now the operator can initiate a reconfiguration of the emulation RAM. INCA calculates a new emulation RAM layout based on the actual set of modified calibration variables which uses as little blocks and as little emulation memory as possible. The goal is to obtain one or more free emulation RAM blocks for further calibration.

Only when it is not possible to further improve the memory usage, the operator must use the Memory Page Manager to flash the Working Page to the Reference Page, thereby freeing the emulation RAM for the emulation of further variables to be changed.

## 3.6 Summary: Comparison of the Concepts

The following table compares the most important features of the old concept with manual experiment preparation and the concepts of dynamic emulation.

Manual Experiment Preparation	Dynamic Emulation
<p>Depending on the size of the project, only a subset of all calibration variables can be calibrated at a time because all variables which are selected in the experiment get emulated.</p> <p>The subset of variables to be calibrated is defined by the location of the emulation range</p>	<p>All variables at once are available for calibration in the Experiment Environment.</p> <p>Only the variables which are calibrated (i.e. only the differences) actually get emulated.</p>
<p>Experiment preparation is required: the user has to determine in advance which variables are to be emulated; INCA checks whether it is possible to emulate all of them at once. If it is not, the variable selection has to be modified in an iterative process.</p>	<p>No experiment preparation is necessary: the user does not have to configure the memory blocks. During calibration and while the system is running, automatic reconfiguration takes place as required.</p>
<p>The concept is not suitable for remote use cases for test benches and Hardware-in-the-Loop application because user interaction is required</p>	<p>The concept is suitable for remote use cases for test benches and Hardware-in-the-Loop application because less user interaction is required.</p> <ul style="list-style-type: none"> <li>– Moveable emulation RAM method using fixed block sizes:           <p>When all blocks are used the difference must be flashed to free emulation RAM for further calibration.</p> </li> <li>– Moveable emulation RAM method using reconfigurable block sizes:           <p>Operator can initiate a reconfiguration. Only when no further optimization is possible, the differences must be flashed to free emulation RAM for further calibration.</p> </li> </ul>
<p>Work can be planned ahead.</p>	<p>The user gets informed just when emulation RAM is no longer available.</p>

## 4 Using Limited Emulation with Experiment Preparation for Serial (X)ETKs

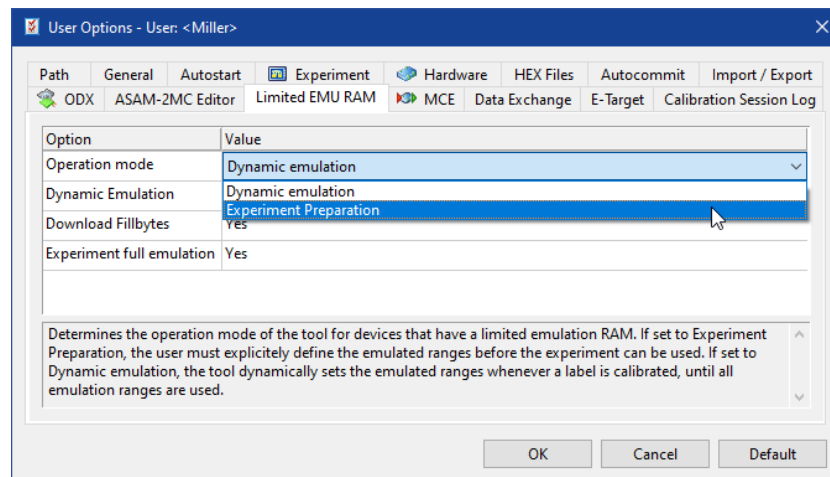
This chapter describes how to use the concept of Limited Emulation RAM in INCA.

### 4.1 Setting the User Option for Emulation with Manual Preparation of the Experiment

In order to use the limited emulation of serial (X)ETKs with manual preparation of the experiment, you have to set the corresponding user option.

#### To set the User Option for Emulation with Experiment Preparation

1. In the INCA Database Manager select **Options > User Options > Open**.
2. In the "User Option" dialog select the "Limited EMU RAM" tab.
3. Select the value **Experiment Preparation** for the **Operation mode** parameter.



4. Exit INCA and restart it to make your changes become valid.

### 4.2 Preparing the Workspace

Before starting the experiment, make sure to complete the following preparation steps for the experiments as usual.

1. Read the A2L file and HEX file for the project and dataset.

### Note

To be able to use the memory block emulation mechanism employed by the Limited Emulation RAM functionality, the A2L file must contain the `CALIBRATION_METHOD FixedSizeMoveableEmuRAM`. For details see "[A2L Examples](#)" on page 38.

Moreover, whether this function is supported depends on type and driver version of the (X)ETK device. If you need a driver update, please contact your local ETAS support.

2. Create a workspace, and set up the hardware configuration.

### Note

For tips on optimizing your work by means of parameter settings in the Hardware Configuration Editor see "[Tips and Tricks](#)" on page 43.

## 4.3 Working on the Experiment

### 4.3.1 Initialization

The initialization procedure for the hardware communication can be started as usual. Please refer to the "INCA Getting Started" manual for details.

At the beginning of the initialization procedure, INCA always calculates the checksums of the memory pages to verify the consistency of the pages. In case emulation RAM is used, the Reference Page must be activated (i.e. the Reference Page must be the "active page") to perform the checksum calculation of both Reference Page and Working Page for this purpose. If you start the initialization with the Working Page activated, INCA will inform you that the active page will be switched to the Reference Page. Once you clicked the <Yes> button of this message box, INCA will activate the Reference Page and proceed the initialization procedure, and the Working Page will automatically be activated again after the procedure is completed.

If INCA detects an inconsistency between the checksums of the PC and the ECU, the memory page management window will open. In this case, you need to perform the appropriate memory operations.

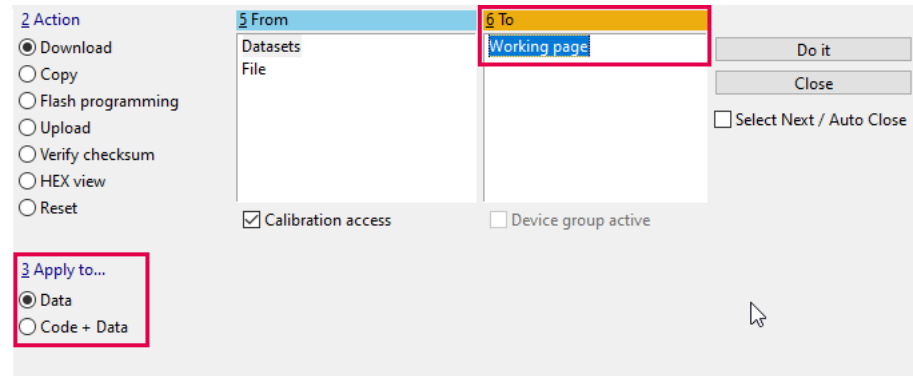
### 4.3.2 Limitations on the Operation of the Memory Pages

When using the emulation RAM, there are some limitations in the following actions for memory management:

- Download

Downloading can only be performed to the Working Page (the emulation RAM blocks to be used can be selected in the Experiment Environment. Please refer to "Settings for the Emulation RAM Area" below).

If you want to transfer the Code Page and the Reference Page to the ECU, you need to perform flash programming.



- Verify checksum

When performing the checksum verification, a switch to the Reference Page might be necessary. Confirm the page if prompted by the following message box.

- Copy

For most (X)ETK devices, copying from Working Page to Reference Page is not possible on the ECU, and only copying from Reference Page to Working Page is possible.

Therefore, to make the data on the Working Page available on the Reference Page, you need to use the menu command **Dataset > Freeze working data**. Afterwards you have to flash the new reference page to the ECU. Please refer to the INCA Online Help for further explanations.

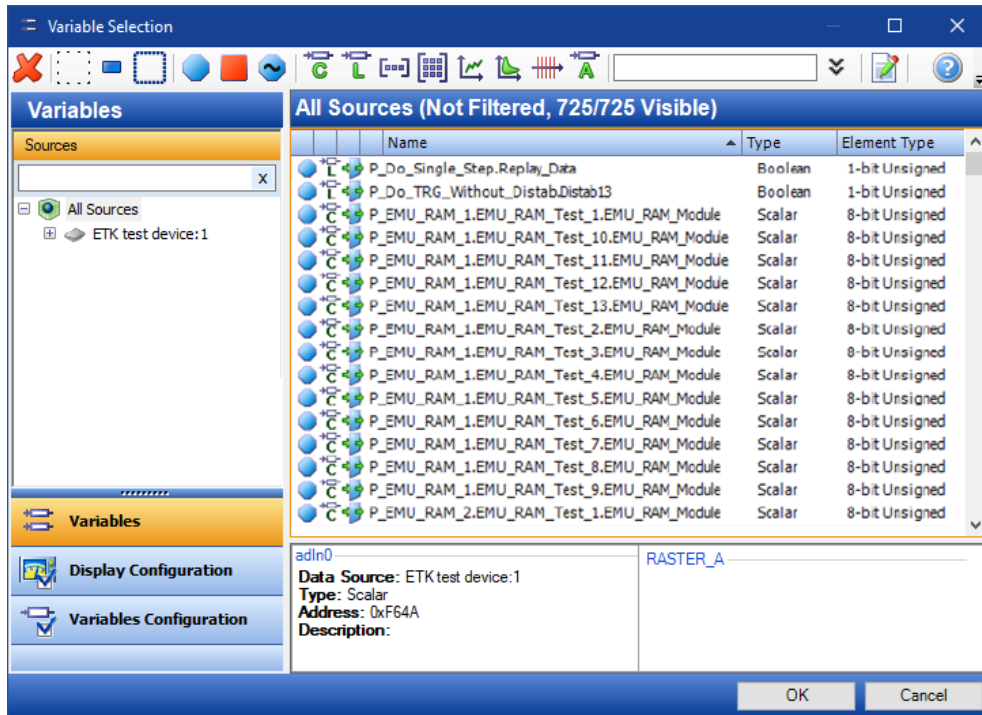
**Note**

This restriction does not apply to combinations of ECU and (X)ETK supporting data freeze. In this case, the behavior is the same as when selecting the functionality Freeze working data: INCA offers copying the Working Page to the Reference Page, and an ECU re-programming of the DATA regions in the Flash memory is performed in the background.

### 4.3.3 Settings for the Emulation RAM Area

Before selecting variables which are used for the experiment, you need to set up the configuration for the emulation RAM in order to determine which flash memory block(s) should be emulated by emulation RAM block(s).

INCA tries to auto-configure the emulation RAM as shown in section "Auto-configuration of the Emulation RAM through INCA" on page 12. Most of the time there will be too many variables to be emulated so that INCA cannot configure the emulation RAM without interaction with the user. In this case the "Experiment preparation to configure emulation RAM" dialog box is automatically displayed:



### Note

For the following case, there is no need to set up the emulation RAM, and therefore the above dialog box is not available:

- Emulation RAM size  $\geq$  data area size

### Note

The "Experiment preparation to configure emulation RAM" dialog box which is shown above is similar to the "Variable Selection" dialog box, but the two dialog boxes are used for different purposes:

- "Experiment preparation..." dialog box:

Here, you can select the calibration variables to be available for calibration and set up the emulation RAM configuration.

- "Variable Selection" dialog box:

Here, you can select the measurement and calibration variables to be used in the Experiment Environment.

It is possible to select calibration variables which are located in areas which are not emulated. However, those variables will be grayed out in the Experiment Environment window, and they cannot be edited, even when the Working Page is activated.

If required you can set up the emulation RAM as follows:

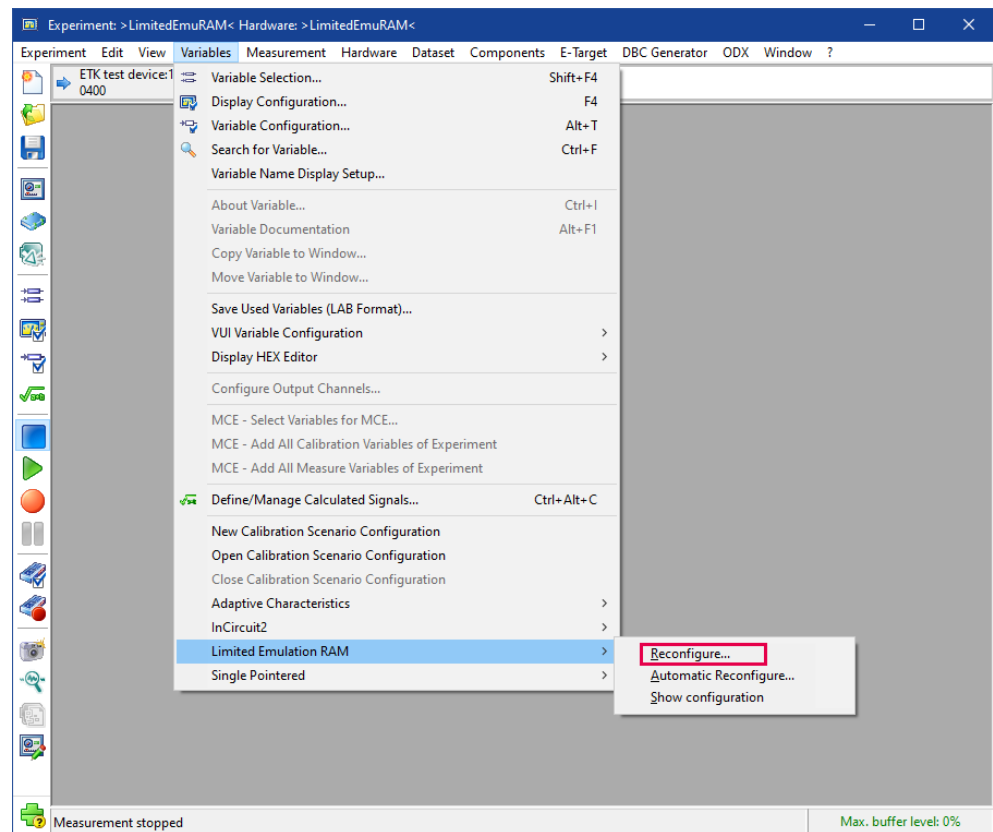
### To set up the emulation RAM configuration

The emulation RAM configuration has to be set up in the "Experiment preparation to configure emulation RAM" dialog box in the Experiment Environment. This dialog box can be opened in one of the following two ways:

1. If INCA detects a difference between the settings stored on the (X)ETK and the settings in the Experiment upon initialization, INCA automatically attempts to open the dialog box before you can work in the Experiment Environment. The settings stored on the (X)ETK remain even after the power-down.

or

If the dialog box is not opened automatically at opening the Experiment, you can open it manually by selecting **Variables > Limited Emulation RAM > Reconfigure**.



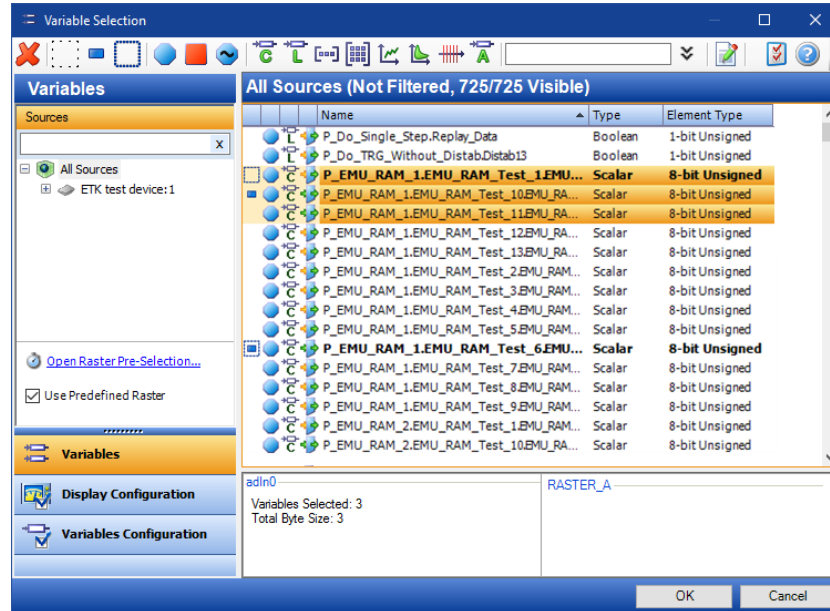
2. If the working data do not equal the reference data, you are prompted by a message box to freeze the working data and write it onto the flash memory.
3. Make sure to click **Yes** to perform the flash programming.

### **Note**

If you click **No**, the consistency of the working data on the ECU and in INCA may be lost.



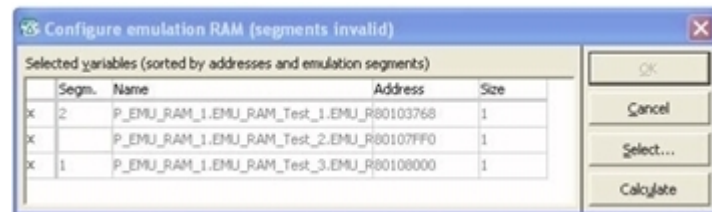
Working data are programmed onto the flash memory, and the "Experiment preparation to configure emulation RAM" dialog box opens as soon as this process is completed. All calibration variables are displayed in the view of the variables pane.



4. Select the variables you need to calibrate.
5. Click **OK**.

If it is possible to emulate all selected variables on the emulation RAM, the setting is completed here.

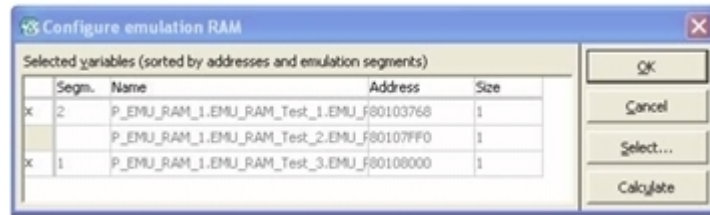
If it is not possible to emulate all of the selected variables (i.e. if the size of emulation RAM is not sufficient for all selected variables), the "Configure emulation RAM" dialog box opens.



In the example above, there are only two emulation RAM blocks available, and therefore it is not possible to emulate all of the three variables selected. Thus, the **OK** button is disabled.

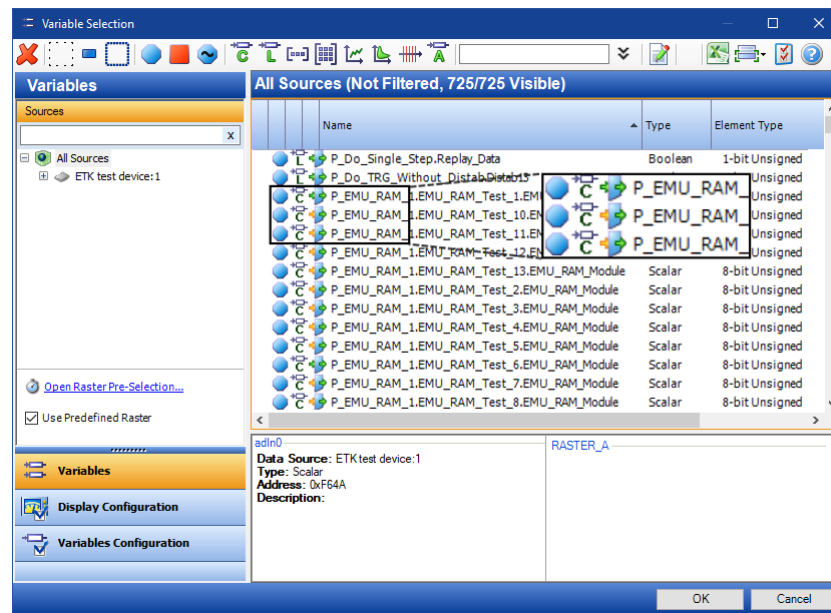
6. To set the proper configuration, remove the "x" mark of any of the variables to disable it.
7. Click the **Calculate** button.

If the configuration is appropriate, the **OK** button will get enabled:



If the **OK** button still does not get enabled, disable more variable(s) and start re-calculation as described above.

8. Click **OK** to complete the setting of the configuration.  
The emulation RAM configuration gets accepted.
9. Open the "Variable Selection" dialog box as usual.



As shown in the example above, the calibration variables which cannot be calibrated are marked with an orange arrow on the left of the symbol. Variables that can be calibrated are marked with a green arrow.

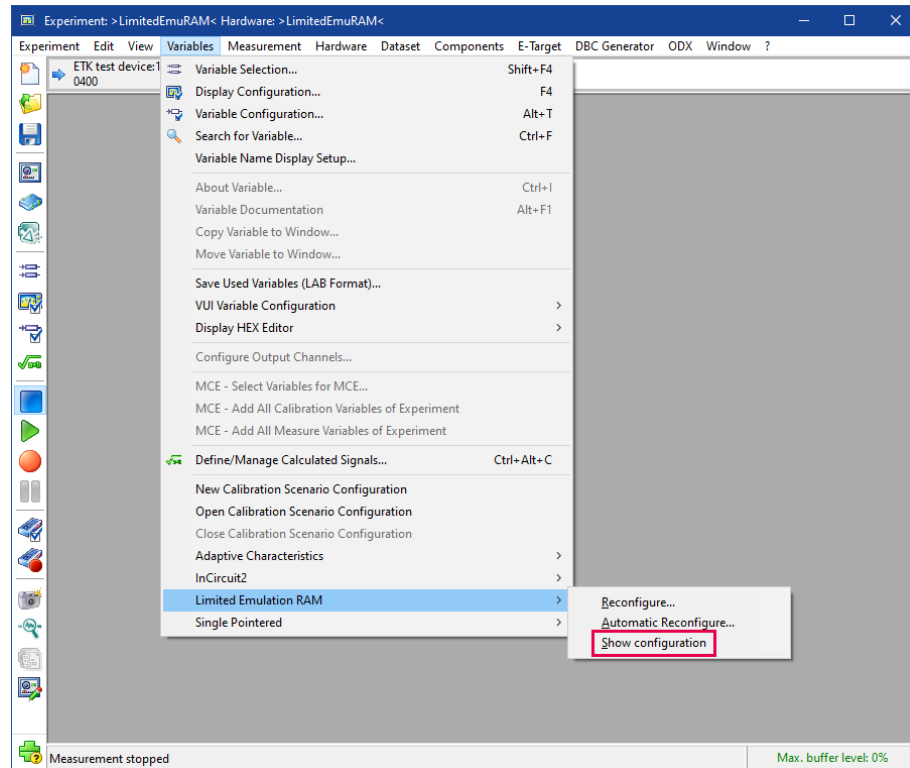
10. Select the variables to be used for the experiment as usual.

#### 4.3.4 Displaying the Emulation RAM Configuration

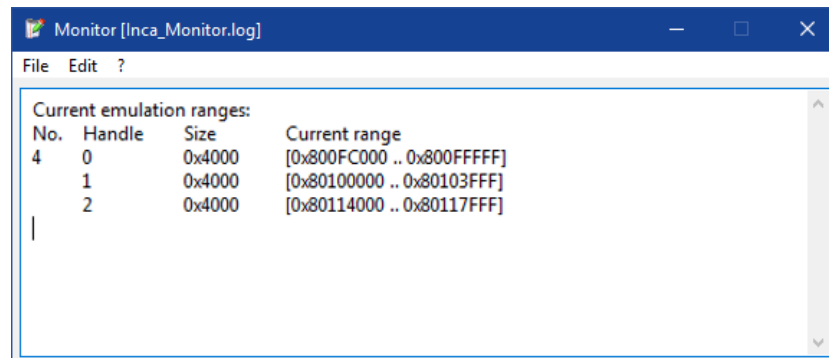
You can check the current emulation RAM configuration (i.e., the emulation RAM block number and the address of the emulated flash memory block). Proceed as follows:

### To display the current configuration of the emulation RAM

1. Open the Experiment Environment.
2. Select **Variables > Limited Emulation RAM > Show configuration**.



Information on the current configuration is displayed in the INCA Monitor window.



## 5 Using Dynamic Emulation for Serial (X)ETKs

This chapter describes how to use the concept of Dynamic Emulation in INCA.

### 5.1 Setting the User Options for Dynamic Emulation

In order to use dynamic emulation of serial (X)ETKs, you have to set the user options described in the following.

#### To set the user options required for dynamic emulation

1. In the INCA Database Manager select **Options > User Options > Open**.
2. In the "User Options" dialog select the "Limited EMU RAM" tab.
3. Select the value **Dynamic emulation** for the **Operation mode** parameter.

#### **Note**

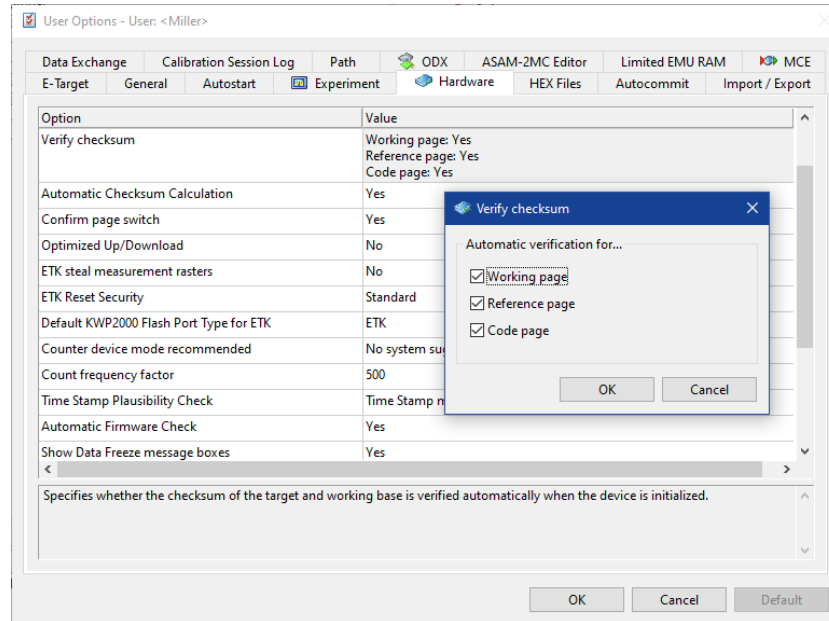
If you change the operation mode towards dynamic emulation and use an (X)ETK type which does not support this type of operation, INCA automatically switches the mode:

If the A2L file contains the CALIBRATION\_METHOD `FixedSizeMoveableEmuRAM`, INCA will switch to the default mode with manual experiment preparation, if it contains the CALIBRATION\_METHOD `ReconfigurableSizeMovableEmuRAM`, INCA will switch to offline mode. In both cases INCA shows a message which informs you about the mode switch.

4. Determine whether you want to use full emulation of the experiment:
  - Enable full emulation if you are working with an INCA experiment where all needed calibrations are already configured: in this case you will get the best emulation RAM configuration after opening the experiment.
  - It is recommended that you disable full emulation if you would like to add further calibrations after opening the experiment; otherwise, the system might run out of free emulation RAM segments, because all calibrations in the experiment are already emulated, including those that are not necessarily needed for your task.

Moreover, you have to activate the checksum calculation on Reference Page and Working Page upon initialization when you want to use dynamic emulation.

5. In the "User Options" dialog select the "Hardware" tab.
6. In the field for the **Verify checksum** parameter, make sure that the checksum calculation for both Working and Reference page is activated (value `Yes`).



### Note

If this parameter is set to **No** and if therefore checksums are not verified, the Limited Emulation RAM functionality assumes that the Reference pages are different and disables calibration access.

7. Confirm your settings with **OK**. Then exit INCA and restart it to make your changes become valid.

You can further facilitate your work by setting additional user options. For detailed information see "[Tips and Tricks](#)" on page 43.

## 5.2 Preparing the Workspace

Before starting the experiment, make sure to complete the following preparation steps for the experiments as usual:

1. Read the A2L file and HEX file for the project and dataset.

 **Note**

To be able to use the memory block emulation mechanism employed by the Limited Emulation RAM functionality, the A2L file must contain the CALIBRATION\_METHOD `FixedSizeMoveableEmuRAM` or `ReconfigurableSizeMovableEmuRAM`.  
For details see "[A2L Examples](#)" on page 38.

2. Create a workspace, and set up the hardware configuration.

 **Note**

For tips on optimizing your work by means of parameter settings in the Hardware Configuration Editor see "[Tips and Tricks](#)" on page 43.

## 5.3 Working with the Experiment

### 5.3.1 Initialization

The initialization procedure for the hardware communication can be started as usual. Please refer to the "INCA Getting Started" manual and to section "[Initialization](#)" on page 21 in this document.

 **Note**

If below listed dynamic emulation methods are defined in the A2L file, INCA uses `ReconfigurableSizeMovableEmuRAM`.

Defined dynamic emulation methods:

`FixedSizeMoveableEmuRAM`  
`FixedSizeNonMoveableEmuRAM`  
`FlashRangeSpecificEmuRAM`  
`ReconfigurableSizeMoveableEmuRAM`

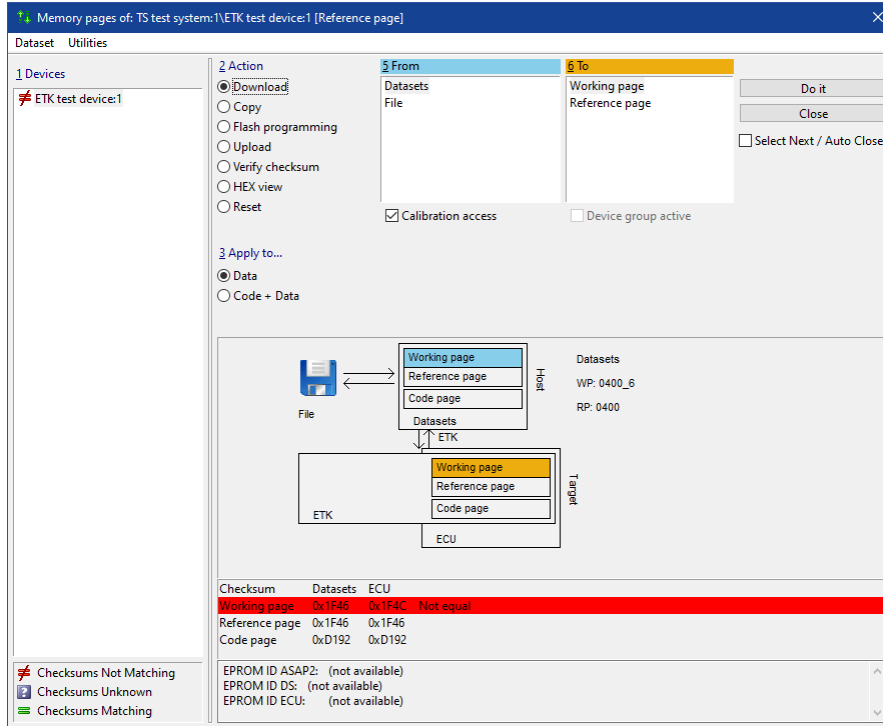
Upon initialization INCA starts to compare the checksums of the memory pages. If Working Pages and Reference Pages in INCA and on the ECU are identical, you can immediately start with calibrating (see "[Calibrating Variables in the Experiment Environment using Dynamic Emulation](#)" on page 32).

If, however, INCA detects an inconsistency between the checksums of the PC and the ECU, the Memory Page Manager will be displayed. In this case, you need to perform the appropriate memory operations.

### 5.3.2 Resolving Differences by Means of Memory Page Operations

Three different situations may occur:

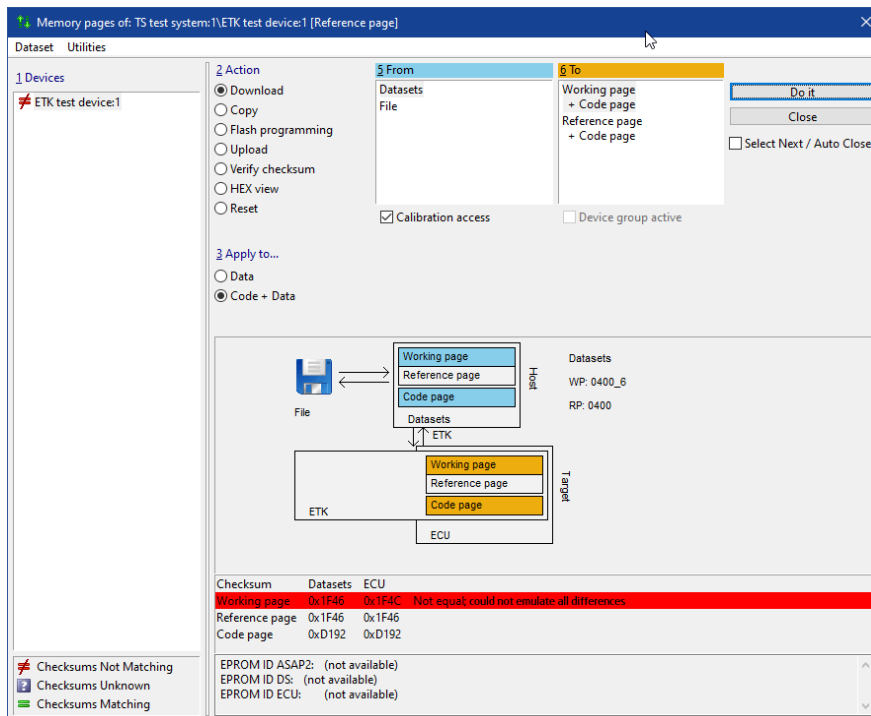
## Working Pages in INCA and in ECU are different



### To resolve the differences

1. Download the INCA Working Page to the ECU, thereby enabling calibration in the experiment.

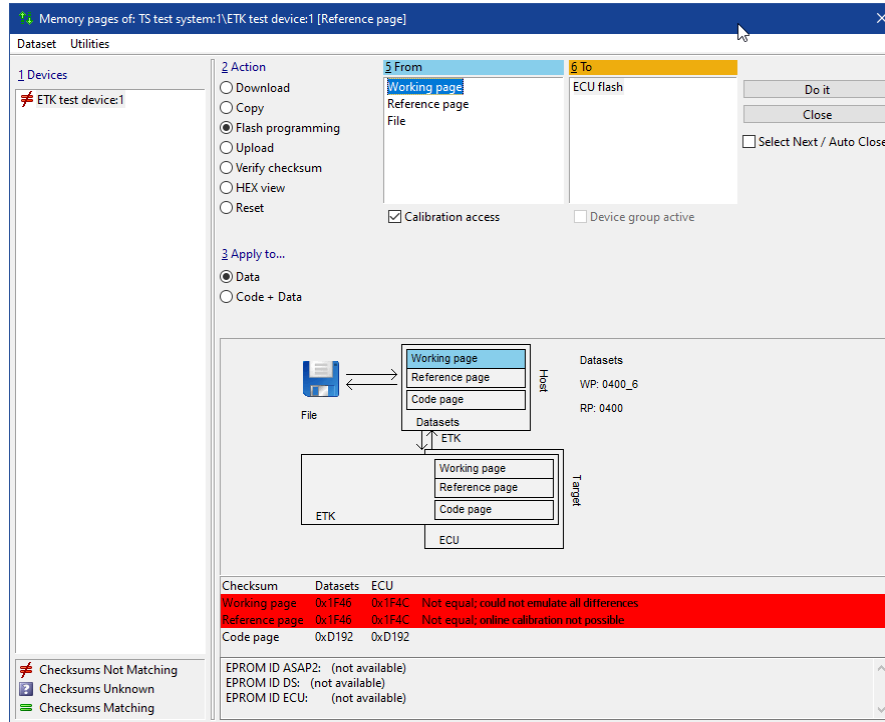
## Working Pages in INCA and in ECU are different, Working Page in INCA contains too many differences



### To resolve the differences

1. Flash the INCA Working Page to the ECU Flash memory, thereby enabling calibration in the experiment.

### Reference Pages in INCA and in ECU are different



### To resolve the differences

1. Flash the INCA Working Page to the ECU Flash memory, thereby enabling calibration in the experiment.

OR

Flash the INCA Reference Page into the ECU Flash memory and download the INCA Working Page. In this case, Working and Reference Page remain different from each other.

## 5.3.3 Calibrating Variables in the Experiment Environment using Dynamic Emulation

You can start selecting, configuring and calibrating variables in the Experiment Environment as usual. Apart from the steps described in the previous sections, there is no additional experiment preparation necessary.

You might have more variables in your experiment than can be emulated at a time. In this case all variables which have not been emulated get write protected and are displayed in gray:



Variable Name	Value
P_EMU_RAM_1.EMU_RAM_Test_1.EMU_R...	1.0000
P_EMU_RAM_1.EMU_RAM_Test_10.EMU_...	2.0000
P_EMU_RAM_1.EMU_RAM_Test_11.EMU_...	2.0000
P_EMU_RAM_1.EMU_RAM_Test_12.EMU_...	2.0000
P_EMU_RAM_1.EMU_RAM_Test_13.EMU_...	4.0000
P_EMU_RAM_1.EMU_RAM_Test_2.EMU_R...	4.0000
P_EMU_RAM_1.EMU_RAM_Test_3.EMU_R...	0.0000
P_EMU_RAM_1.EMU_RAM_Test_4.EMU_R...	0.0000
P_EMU_RAM_1.EMU_RAM_Test_5.EMU_R...	0.0000
P_EMU_RAM_1.EMU_RAM_Test_6.EMU_R...	0.0000
P_EMU_RAM_1.EMU_RAM_Test_7.EMU_R...	0.0000
P_EMU_RAM_1.EMU_RAM_Test_8.EMU_R...	0.0000
P_EMU_RAM_1.EMU_RAM_Test_9.EMU_R...	0.0000

You have the following options:

- You can reconfigure the emulation RAM so that specific variables of a calibration instruments get emulated.
- You can use dynamic reconfiguration in order to optimize the emulation RAM layout.

### Note

These two functions are only available if the A2L file contains the CALIBRATION\_METHOD ReconfigurableSizeMovableEmuRAM and FlashRangeSpecificEmuRAM. For details see "[A2L Examples](#)" on [page 38](#).

- You can free emulation RAM either by reverting calibration values to their reference page value or by flashing the INCA Working Page to the ECU.

#### To reconfigure the emulation RAM with focus on specific variables

If you want to calibrate currently write-protected variables, you can use this method to reconfigure the emulation RAM with focus on the corresponding variables. This type of reconfiguration tries to emulate all or selected variables of a specific calibration instruments.

#### To calibrate only some variables of a calibration instrument

1. Mark these variables in the calibration instrument.
2. Open the context menu of the calibration instrument and select **Limited Emulation RAM > Try to emulate Selected Variables**.

#### To calibrate all variables of a calibration instrument

1. Open the context menu of the calibration instrument and select **Limited Emulation RAM > Try to emulate All Variables**.

INCA will only reconfigure the emulation RAM if all of the corresponding variables can be emulated; if this is not possible, no reconfiguration will take place. In this case you can free the emulation RAM as described below.

 **Note**

When the emulation RAM shall be changed, INCA asks you to switch to the ECU reference page beforehand.

#### To optimize the emulation RAM by a general reconfiguration

If you want to optimize the emulation RAM in general, without special focus on selected variables, you can use the following method.

1. Select **Variables > Limited Emulation RAM > Automatic Reconfigure**.

Based on the actual set of variables that have differences between Working Page and Reference Page, INCA calculates a better emulation RAM layout. The emulation RAM freed through this procedure can now be used for emulating additional variables.

If dynamic reconfiguration does not provide sufficient memory for the required calibrations or is not possible, you can free the emulation RAM as described below.

 **Note**

When the emulation RAM shall be changed, INCA asks you to switch to the ECU reference page beforehand.

#### To free the emulation RAM for further calibration through reverting to the reference page value or flashing

You have the following options to free emulation RAM for further calibration:

1. Revert individual calibration values to their reference page value by selecting **Reset to Reference Page Value** from the context menu or the **Edit** menu.

As there is no longer a difference between working page and reference page for this calibration variable, the emulation RAM is now available for emulating other calibrations.

*or*

Open the Memory Page Manager and flash the INCA Working Page to the ECU.

The Working Page in INCA and the ECU are now identical again.

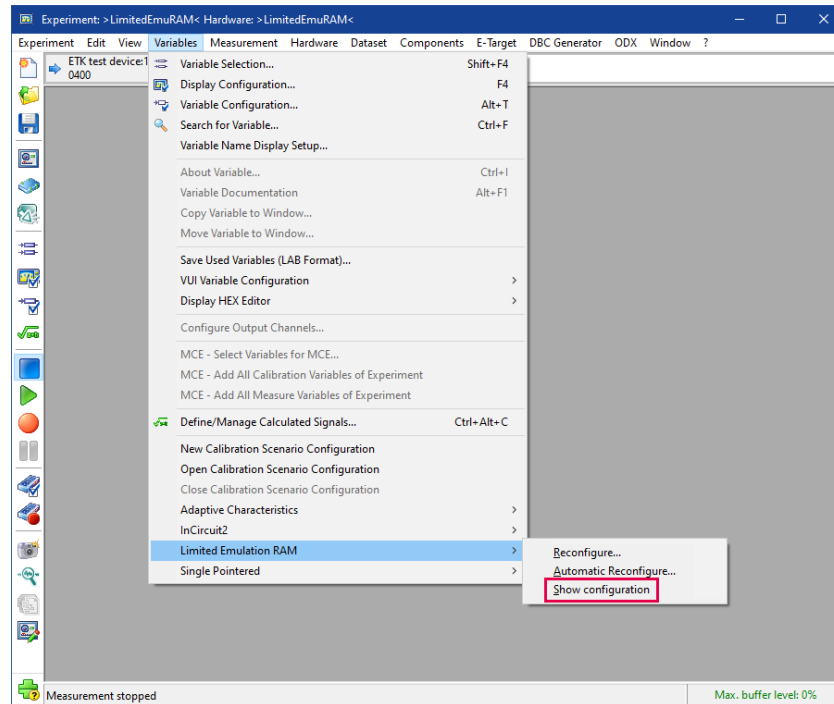
The variables in the INCA experiment are no longer write protected, and you can go on calibrating, whereas the calibrated variables are emulated dynamically.

### 5.3.4 Displaying the Emulation RAM Configuration

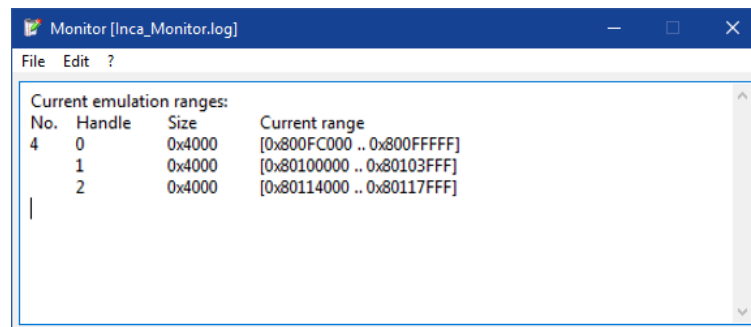
You can check the current emulation RAM configuration (i.e. the emulation RAM block number and the address of the emulated flash memory block). Proceed as follows:

To display the current configuration of the emulation RAM

1. Open the Experiment Environment.
2. Select **Variables > Limited Emulation RAM > Show configuration**.



Information on the current configuration is displayed in the INCA Monitor window.



## 6 Restrictions

This product has been manufactured with the utmost care for quality, usability and completeness in terms of users' expectations.

Nevertheless there are constraints that need to be considered when using this product.

### 6.1 General Restrictions

- The following types of calibration variables are not supported by the Limited Emulation RAM functionality:
  - Dependent
  - Virtual
  - Calibration variables that are marked as read-only
  - Calibration variables that are located outside of DATA memory segments and (if optionally active) outside of OFFLINE\_DATA memory segments
  - Calibration variables that are located in a RESERVED memory segment
- For some ETK devices (e.g. the ETK-S20.0), the emulation RAM layout is not stored in a nonvolatile memory of the ETK device itself, but in the RAM of the ETAS Interface Module (e.g. an ES592).

If the ETAS Interface Module is switched off or enters power saving mode, the content of the emulation RAM layout is lost. As a consequence, different checksums between the working page in INCA and on the ECU will be detected during the next initialization, requiring a download from INCA to the ECU.

A comparable behavior applies to XETK devices: If the power-supply of the XETK device is interrupted or the XETK device enters standby mode, different working page checksums will be reported for INCA and the ECU during the next initialization.

### 6.2 Restrictions using Limited Emulation with Experiment Preparation

- The concept is not suitable for remote use cases (e.g. test benches, Hardware-in-the-Loop application) because the selection of variables in the experiment preparation step cannot be automated via ASAP3.

### 6.3 Restrictions using Dynamic Emulation

- Write protection for variables outside of ASAP2 data ranges: variables with an address in memory blocks that are located outside the ASAP2 data ranges are write protected for online access. Depending on their location in the dataset (memory block type offline) they can be edited offline. If the

address of calibration variables is located outside the dataset, the variables can not be edited at all.

- Downloading from file to the Working Page is not supported in the Memory Page Manager.
- Grouped ECUs: In addition to the usual requirements which have to be fulfilled to enable calibration on all ECUs in the group, it must be possible to display the variable, which is to be calibrated, separately for each ECU. If this is not possible, the variable is calibrated only for one ECU. This situation may occur when a different number of emulation ranges is defined in the ECU projects or when the differences between Working Page and Reference Page are not identical on the individual ECUs in the group.
- Dynamic Emulation with reconfigurable block sizes is only available if the A2L file contains the `CALIBRATION_METHOD ReconfigurableSizeMovableEmuRAM`.

For details see "[A2L Examples](#)" on page 38.

Moreover, whether this function is supported depends on type and driver version of the (X)ETK device. If you need a driver update, please contact your local ETAS support.

## 7 A2L Examples

The Limited Emulation RAM functionality uses a memory block overlay mechanism (emulation) for the calibration with serial (X)ETKs. To allow a calibration tool the use of this mechanism, it must be described in the A2L File.

For that purpose the A2L file must contain one of the following entries below  
CALIBRATION\_METHOD:

- FixedSizeMoveableEmuRAM
- FixedSizeNonMoveableEmuRAM
- FlashRangeSpecificEmuRAM.
- ReconfigurableSizeMoveableEmuRAM

The individual emulation blocks are defined by a handle number, the start address and the block size. The possible values depend on the microcontroller used in the ECU and the (X)ETK device.

### 7.1 A2L Example using FixedSizeMoveableEmuRAM

When using the calibration method **FixedSizeMoveableEmuRAM**, typically all blocks have the same size. However, some microcontrollers can use different block sizes (which are fixed in the sense that they can not be changed dynamically). In both cases, the block sizes must be defined in the A2L file.



#### Note

The A2L file contains an extra CALIBRATION\_HANDLE for each emulation memory block.

Block sizes and start addresses of the blocks that are defined in the A2L file are directly related to each other: The sizes of the emulation blocks (e.g. "0x4000" in the example below) determine the segment borders of the Reference Page, i.e. if the block size is 0x4000, each block must begin at an address which can be divided by 0x4000.

#### *Example with all blocks having the same size*

```

/begin CALIBRATION_METHOD
    "FixedSizeMoveableEmuRAM"    /* Method name                */
    1                            /* Method version            */
    /begin CALIBRATION_HANDLE
        0                        /* Handle no. (emu register no. of CPU) */
        0xFFFF0000             /* Start address of emu RAM (in RAM area) */
        0x4000                 /* Length of emu RAM          */
    /end CALIBRATION_HANDLE
    /begin CALIBRATION_HANDLE
        1                        /* Handle no. (emu register no. of CPU) */

```

```

        0xFFFF4000 /* Start address of emu RAM (in RAM area) */
        0x4000      /* Length of emu RAM */
    /end CALIBRATION_HANDLE

...

/end CALIBRATION_METHOD

```

### Example with blocks having different sizes

```

/begin CALIBRATION_METHOD
    "FixedSizeMoveableEmuRAM" /* Method name */
    1                          /* Method version */
    /begin CALIBRATION_HANDLE
        0 /* Handle no. (emu register no. of CPU) */
        0xFFFF0000 /* Start address of emu RAM (in RAM area) */
        0x8000     /* Length of emu RAM */
    /end CALIBRATION_HANDLE
    /begin CALIBRATION_HANDLE
        1 /* Handle no. (emu register no. of CPU) */
        0xFFFF8000 /* Start address of emu RAM (in RAM area) */
        0x4000     /* Length of emu RAM */
    /end CALIBRATION_HANDLE

...

/end CALIBRATION_METHOD

```



#### Note

Whether different block sizes are supported depends on type and driver version of the (X)ETK device.

Moreover, there may be specific restrictions on the possible sizes of the blocks (e.g. the microcontroller supports 2 different block sizes, where the size of the second block is twice the size of the first block).

## 7.2 A2L Example using FixedSizeNonMoveableEmuRAM

### Example

```

/begin CALIBRATION_METHOD
    "FixedSizeMoveableEmuRAM" /* Method name */
    1                          /* Method version */
    /begin CALIBRATION_HANDLE
        0 /* Handle */
        0xBF400000 /* Emu RAM start address */
        0x20000    /* Emu RAM size */
        0x00300000 /* Flash start address */
    /end CALIBRATION_HANDLE

```

```

        /end CALIBRATION_HANDLE
    ...
/end CALIBRATION_METHOD

```



### Note

Whether fixed sized non moveable blocks are supported depends on type and driver version of the (X)ETK device. Moreover, there may be microcontroller specific restrictions on the possible sizes of the blocks, number of blocks, and flash ranges each block is usable with.

## 7.3 A2L Example using FlashRangeSpecificEmuRAM

```

/begin CALIBRATION_METHOD
    "FlashRangeSpecificEmuRAM"          /* Method name          */
    1                                     /* Method version       */
        /begin CALIBRATION_HANDLE
            0                             /* Overlay type: 0->
                ClusterOverlay            */
            32                             /* Handle start index   */
            32                             /* Handle count         */
            4                               /* Handle group size    */
            0x8000                          /* Handle page size     */
            0x8000                          /* Handle alignment     */
            1                               /* Number of Emu RAM ranges*/
            0xFB000000                      /* Emu RAM start address */
            0x100000                        /* Emu RAM size         */
                /* If more than one Emu RAM
                range is specified, start
                address and size is to be
                added here          */
            2                               /* Number of Flash Ranges */
            0x00000000                      /* Flash Bank A start
                address            */
            0x00400000                      /* Flash Bank A size     */
            0x00400000                      /* Flash Bank B start
                address            */
            0x00400000                      /* Flash Bank B size     */
                /* If more flash ranges are
                specified, start address
                and size is to be added
                here          */
        /end CALIBRATION_HANDLE

```



```

/begin CALIBRATION_HANDLE
  1 /* Overlay type: 1->
      GlobalOverlay */
  0 /* Handle start index */
  32 /* Handle count */
  0x2000 /* Overlay min size */
  0x20000 /* Overlay max size */
  1 /* Number of Emu RAM ranges */
  0xFB400000 /* Emu RAM start address */
  0x180000 /* Emu RAM size */
  1 /* Number of Flash Ranges */
  0x00000000 /* Flash start address */
  0x00800000 /* Flash size */
/end CALIBRATION_HANDLE
/end CALIBRATION_METHOD

```



### Note

Whether flash range specific blocks are supported depends on type and driver version of the (X)ETK device.

Moreover, there may be microcontroller specific restrictions on the possible sizes of the blocks, number of blocks, and flash ranges each block is usable with.

## 7.4 A2L Example using ReconfigurableSizeMoveableEmuRAM

```

/begin CALIBRATION_METHOD
  "ReconfigurableSizeMoveableEmuRAM" /* Method name */
  1 /* Method version */
  /begin CALIBRATION_HANDLE
    0 /* Handle Range Lower */
    15 /* Handle start index */
    0x20 /* Handle count */
    0x20000 /* Handle group size */
    0xBF000000 /* Handle page size */
    0xC0000 /* Handle alignment */
    /* If more than one Emu RAM
       range is specified for
       each Emu RAM, start
       address and size is to be
       added here */
  /end CALIBRATION_HANDLE

```

```
/end CALIBRATION_METHOD
```

**Note**

Whether reconfigurable block sizes are supported depends on type and driver version of the (X)ETK device.  
Moreover, there may be microcontroller specific restrictions on the possible sizes of the blocks.

An AML description including ETAS specific enhancements can be obtained from ETAS.

## 8 Tips and Tricks

This chapter provides some help information for tasks which might be difficult to understand or can be optimized by means of little tips and tricks.

### Optimizing Your Work by Settings in the User Options and Hardware Configuration Editor

You can optimize your work by setting certain parameters in the User Options and in the Hardware Configuration Editor as described below.

#### To optimize the settings in the user options

1. In the INCA Database Manager select **Options > User Options > Open**.
2. In the **User Options** dialog select the **Limited Emu RAM** tab.
3. Optional: You can set the parameter **Download Fillbytes** to **No** if you want to make sure that address ranges with fill bytes will not be downloaded to the ECU memory.

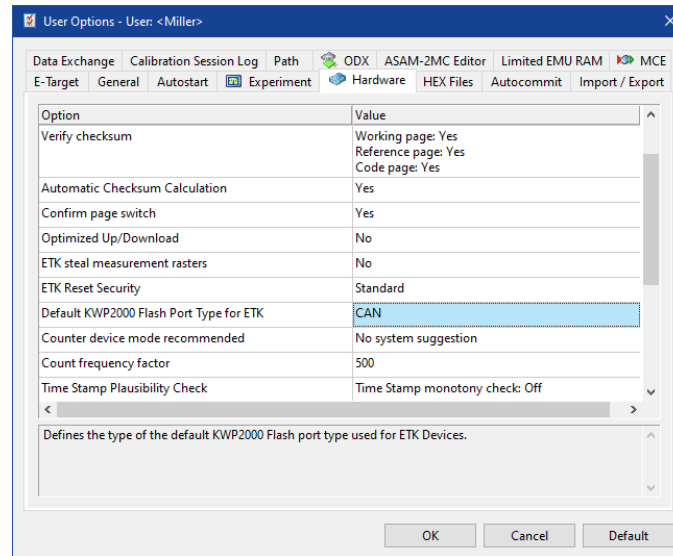
INCA requires that some address ranges that are not defined in the A2L file are initialized with fill bytes. However, it is not always desirable that these fill bytes get written into the ECU memory upon download of the working page. Therefore, INCA offers a user option **Download Fillbytes**. With this option you can define whether the download of the working page to the ECU only writes address ranges that are defined in the A2L file as MEMORY\_SEGMENT with the memory type FLASH, or whether address ranges with other memory types, that are filled with fill bytes, are to be downloaded as well.

4. In the **User Options** dialog select the **Limited EMU RAM** tab.
5. Optional: You can set the parameter **Dynamic Emulation** to **Allow calibration with different RPs**.

Calibration with different RPs is possible if all calibrations of the used ECU project can be emulated (full ECU project emulation).

6. In the **User Options** dialog select the **Hardware** tab.
7. Optional: You can set the parameter **Confirm page switch** to **No** if you would like to skip the confirmation of the memory page switch at initialization.
8. Optional: Under **Default KWP2000 Flash Port Type for ETK**, you can select your preferred type of flash port (ETK, CAN or K-Line) which will be

used as the default type for all new ECU devices.



9. Confirm your settings with **OK**.

#### To optimize the Parameter Settings in the Hardware Configuration Editor

1. In the **Hardware Devices** list of the Hardware Configuration Editor, select the serial ETK you are using for your calibration task.
2. In the **Parameters** list on the right side of the window, click twice in the entry field for the **Flash Port** and select the flash port you would like to use by default.
3. With the parameter **Confirm page switch** in the **Parameters** list, you can also define whether you have to confirm explicitly each page switch.

## 9 Contact Information

### Technical Support

For details of your local sales office as well as your local technical support team and product hotlines, take a look at the ETAS website:

[www.etas.com/hotlines](http://www.etas.com/hotlines)

ETAS offers trainings for its products:

[www.etas.com/academy](http://www.etas.com/academy)



### ETAS Headquarters

ETAS GmbH

Borsigstraße 24	Phone:	+49 711 3423-0
70469 Stuttgart	Fax:	+49 711 3423-2106
Germany	Internet:	<a href="http://www.etas.com">www.etas.com</a>