Guidelines to move an existing Project from DAVE™ v4.0.0 to DAVE™ v4.1.2

Scope
This document describes incompatibilities between the artifacts between the ecosystem of DAVE v4.0.0 and DAVE v4.1.2. And it describes required actions to move a project created in DAVE v4.0.0 Beta release in DAVE v4.1.2 Productive release.

Notes
All artefacts (DAVE, DAVE SDK, DAVE APPs, Device Descriptions and XMC LIB) now released as productive version are planned to be full upwards compatibility with future version upgrades.

Artifacts that are still released as Beta version (some DAVE APPs) may be incompatible with future version upgrades. For this purpose DAVE APPs released as Beta version will be visually indicted in the relevant APP views in DAVE. DAVE APPs released as Beta version may have not reached the final design status and should be used for evaluation and test purposes and for adaption in the development flow.

Sections in this Document
1. Overview of incompatibilities between DAVE v4.0.0 and DAVE v4.1.2, related APPs and other libraries
2. Detailed information of incompatibilities of DAVE APPs and required actions to move a project from DAVE v4.0.0 to DAVE v4.1.2
3. Detailed incompatibilities of the XMC Lib v1.00 and v2.0.0 and required actions to use the application code developed with XMC Lib v1.0.0 with XMC Lib 2.0.0.
1. Overview of incompatibilities between DAVE v4.0.0 and DAVE v4.1.2, related APPs and other libraries

Due to identified issues and improvement request it was not possible the keep full backward compatibility between the ecosystem of DAVE v4.0.0 and DAVE v4.1.2.

Therefore it is not possible to use workspaces created with DAVE v4.0.0 with DAVE v4.0.0 or local library store installed with DAVE v4.0.0.

It is however possible to import projects developed with DAVE v4.0.0 into DAVE v4.1.2.

There are the following different cases that can be distinguished when importing a project, created with DAVE 4.0.0 into DAVE v4.1.2

1. In case the imported projects includes DAVE APPs and it is not required to add new DAVE APPs. It is recommended to upgrade the project (see section 2) because under some circumstances the solver may fail.
2. In case the imported projects includes DAVE APPs and it is required to add new DAVE APPs. In this case it is required to upgrade the project (see section 2).
3. In case the imported project doesn’t include any DAVE APPs, but the XMC Lib is used. Generally then there are no specific action required. But if the user want to benefit from the improvements of XMC Lib v2.0.0, the XMC Lib should be replaced and some upgrade actions may be required (see section 3).
4. In cases the imported project doesn’t include any DAVE APPs, and the XMC Lib is not used there are no specific action required.

1.1 Overview of incompatibilities related to the DAVE development environment

All functional enhancements in DAVE v4.1.2 are upwards compatible to DAVE v4.0.0.

1.2 Overview of incompatibilities of DAVE APPs

There are in total 18 DAVE APPs that are not upwards compatible. The following different natures of incompatible changes:

- Data type
- Data name
- Members in data type
- API name
- API signature
- Remove of API

Separate to that the following DAVE APPs have been removed from the latest release because the required improvements will take longer with more significant incompatibility changes are expected.
BLDC_SCALAR_CTRL, HALL_POSIF, PMSM_FOC, PMSM_SCALAR_CTRL
Existing project that uses these APPs cannot be upgraded to DAVE v4.1.2.

Also the GUI_LCD APP has been removed. The functionality of this APP has been integrated in the GUISEGGERLIBRARY APP hence the GUI_LCD APP is not required any more. This change is fully upward compatible.

1.3 Overview of incompatibilities between XMC Lib v 1.0.0 and XMC Lib v2.2.0

In total 4 peripheral drivers have been changed in a way that code adaptations are necessary to use them along with existing application code. These are ACMP, CAN, DMA and HRPWM. The natures of the incompatible changes are: API changes, data type and data type members.

2. Detailed information of incompatibilities of DAVE APPs and required actions to move a project from DAVE v4.0.0 to DAVE v4.1.2

As first step the existing project developed in DAVE v4.0.0 should be imported in DAVE v4.2.1

This can be done as follows:

- Press in DAVE v4.1.2 : >File ->Import ->DAVE Project
- Check “Select Root Directory” if the source project is located in its workspace and browse to it location
- Check Select Archive File” if the source project is provided as zip file and browse to its location

In case the “Select Root Directory” is used, don’t forget to check the copy option to full copy the source project into the current workspace.

After the project is imported in the current workspace of DAVE v4.1.2 the project upgrade functionality should be started.
This can be done as follows:

Right mouse click in the project that should be upgraded to the latest APP versions:

- Select APPs Upgrade
- Press OK when the warning pops up

The APP upgrade functionality creates a new project with the latest version startup files, XMC Lib, header and linker script files. The old project remains unchanged. The DAVE APPs existing in old projects will be added as newest version that can be found in the local library store. The adding sequence goes from top to down considering differences in required APPs. After all required APPs are added the configurations, signal connections and pins assignments are copied from the old project to the new project. As consequence of this concept, all DAVE APPs of the old project must be already installed in the local library store; otherwise the missing APP will not be added in the new project. A detailed migration report will be provided after the migration is completed. If a DAVE APP is upward compatible to an older version no further actions are required. If the DAVE APP is not upward compatible some manual interactions depending on the nature of the incompatibilities are required. For the incompatible APPs, the required actions are described below.

<table>
<thead>
<tr>
<th>DAVE APPs</th>
<th>Detailed Change Description that may lead to incompatibilities</th>
<th>Manual project upgrade steps</th>
</tr>
</thead>
</table>
| GLOBAL_HRPWM  | 1. Enum GLOBAL_HRPWM_STATUS_t member name GLOBAL_HRPWM_SUCCESS changed to GLOBAL_HRPWM_STATUS_SUCCESS  
2. Enum GLOBAL_HRPWM_STATUS_t member name GLOBAL_HRPWM_FAILURE changed to GLOBAL_HRPWM_STATUS_FAILURE | Use DAVE's "APPs upgrade" feature and manually correct the parameters with respect to latest APP. |
### HRPWM

1. Data structure member name module_freq changed to frequency_tclk.
2. Data structure member name module_ptr changed to ccu8_module_ptr.
3. Data structure member name slice_ptr changed to hrc_slice_ptr.
4. Enum HRPWM_STATE_t elements name changed.
5. In Struct HRPWM_CONFIG_t variable hrpwm_lr_enable_mask changed to hrpwm_lr_enable_path_mask.
6. In Struct HRPWM_CONFIG_t variable hrpwm_hr_enable_mask changed to hrpwm_hr_enable_path_mask.
7. HRPWM_AcknowledgeInterrupt() api is renamed as HRPWM_ClearEvent().

**Use DAVE's "APPs upgrade" feature and manually correct the parameters with respect to latest APP.**

### POWER_CONV_LIB

1. API POWER_CONV_LIB_PI_FIX() is changed to POWER_CONV_LIB_PI_FIX_Q15()
2. API POWER_CONV_LIB_2P2Z_FIX() is changed to POWER_CONV_LIB_2P2Z_FIX_Q15()
3. API POWER_CONV_LIB_3P3Z_FIX() is changed to POWER_CONV_LIB_3P3Z_FIX_Q15()
4. Structure POWER_CONV_LIB_PI_FIX_COEFFS_t is changed to POWER_CONV_LIB_PI_FIX_Q15_COEFFS_t()
5. Structure POWER_CONV_LIB_PI_FIX_VARS_t is changed to POWER_CONV_LIB_PI_FIX_Q15_VARS_t
6. Structure POWER_CONV_LIB_2P2Z_FIX_COEFFS_t is changed to POWER_CONV_LIB_2P2Z_FIX_Q15_COEFFS_t
7. Structure POWER_CONV_LIB_2P2Z_FIX_VARS_t is changed to POWER_CONV_LIB_2P2Z_FIX_Q15_VARS_t
8. Structure POWER_CONV_LIB_3P3Z_FIX_COEFFS_t is changed to POWER_CONV_LIB_3P3Z_FIX_Q15_COEFFS_t
9. Structure POWER_CONV_LIB_3P3Z_FIX_VARS_t is changed to POWER_CONV_LIB_3P3Z_FIX_Q15_VARS_t
10. Members of above structures from 4 to 9 are changed to uint16_t.

**Change to new API name and change to new data type name.**

### DIM_BCCU

- DIM_BCCU_DitherDisable()
- DIM_BCCU_DitherEnable()
- DIM_BCCU_GetDimCurv() are removed.

**API should not be used**

### PDM_BCCU

- PDM_BCCU_SelDimEngine() has removed.

**API should not be used**

### CAN_NODE

1. CAN_NODE_EnableEvent() and CAN_NODE_ConfigBaudrate() APIs signature have modified. (new argument added)

**Change API name as in PR version of APP**
<table>
<thead>
<tr>
<th>Module</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAC_LUT</td>
<td>2. DAC_LUT_t wave_lut_ptr and wave_lut_coupled_ptr data type updated to uint32. Change to new data type.</td>
</tr>
<tr>
<td>I2C_MASTER</td>
<td>API Name changed for ABORT API's to make it forward compatible with DMA support.</td>
</tr>
<tr>
<td></td>
<td>I2C_MASTER_Abort_Transmit() to I2C_MASTER_AbortTransmit()</td>
</tr>
<tr>
<td></td>
<td>I2C_MASTER_Abort_Transmit() to I2C_MASTER_AbortReceive().</td>
</tr>
<tr>
<td>MANCHESTER_SW</td>
<td>1. Data structure members name have changed.</td>
</tr>
<tr>
<td></td>
<td>MANCHESTER_SW_ioconfig_t to MANCHESTER_SW_ENCODER_IO_SYSTIMER_t</td>
</tr>
<tr>
<td></td>
<td>MANCHESTER_SW_spiconfig_t to MANCHESTER_SW_ENCODER_SPI_CONFIG_t</td>
</tr>
<tr>
<td></td>
<td>MANCHESTER_SW_encoderconfig_t to MANCHESTER_SW_ENCODER_t</td>
</tr>
<tr>
<td></td>
<td>MANCHESTER_SW_decoderconfig_t to MANCHESTER_SW_DECODER_CONFIG_t</td>
</tr>
<tr>
<td></td>
<td>2. Enum name changed MANCHESTER_SW_EncodeTxState_t to MANCHESTER_SW_ENCODER_IO_SYSTIMER_STATE_t</td>
</tr>
<tr>
<td></td>
<td>MANCHESTER_SW_LOGIC_t to MANCHESTER_SW_ENCODER_TECHNIQUE_t</td>
</tr>
<tr>
<td></td>
<td>MANCHESTER_SW_TECHNIQUE_t to MANCHESTER_SW_DECODER_STATE_t</td>
</tr>
<tr>
<td></td>
<td>3. Enum item removed MANCHESTER_SW_SPI_INPUT_t</td>
</tr>
<tr>
<td></td>
<td>4. Enhanced encoding operation using SPI with 32bit data wide and 21bit starting zeros</td>
</tr>
<tr>
<td>SPI_MASTER</td>
<td>API Name changed for ABORT API's to make it forward compatible with DMA support.</td>
</tr>
<tr>
<td></td>
<td>SPI_MASTER_Abort_Transmit() to SPI_MASTER_AbortTransmit()</td>
</tr>
<tr>
<td></td>
<td>SPI_MASTER_Abort_Transmit() to SPI_MASTER_AbortReceive()</td>
</tr>
<tr>
<td>UART</td>
<td>1. removed underscore('_') from the API name.</td>
</tr>
<tr>
<td></td>
<td>UART_Abort_Transmit() to UART_AbortTransmit()</td>
</tr>
<tr>
<td></td>
<td>UART_Abort_Transmit() to UART_AbortReceive()</td>
</tr>
<tr>
<td></td>
<td>2. APIs are renamed: a)UART_GetProtocolStatus() to UART_GetFlagStatus()</td>
</tr>
<tr>
<td></td>
<td>b) UART_ClearProtocolStatus() to UART_ClearFlag()</td>
</tr>
<tr>
<td>DISPLAY_7SEG</td>
<td>API DISPLAY_7SEG_SetBrightness() functionality changed (brightness raw value -&gt; percentage).</td>
</tr>
<tr>
<td></td>
<td>Must pass brightness in percentage rather raw value</td>
</tr>
<tr>
<td>LED_MATRIX</td>
<td>Brightness API functionality changed (brightness raw value -&gt; percentage)</td>
</tr>
<tr>
<td></td>
<td>User must pass brightness in percentage rather raw value</td>
</tr>
<tr>
<td>ADC_MEASUREMENT</td>
<td>In Channel structure name one word changed to capital letters.</td>
</tr>
<tr>
<td></td>
<td>Change to new data structure name</td>
</tr>
</tbody>
</table>
## Guidelines to move from DAVE™ v4.0.0 to DAVE™ v4.1.2

### COUNTER
- 1. UI Layout improved to support additional gating event.
- 2. API: COUNTER_AcknowledgeInterrupt change to COUNTER_ClearEvent

Migration will work without fail. New feature added to support raising of event on various edges. Because of this, user need to change the option in UI to get the event, if different from the default option provided in UI.

Use DAVE's "APPSs upgrade" feature and Configure parameters which are missed out by feature. Change API name as in PR version of APP.

### CRYPTO_AES
- One API signature (return type) modified

Change to new API return type.

### DMA.CH
- Data type for source and destination address to be redefined from uint32_t * to uint32_t

Change the usage of data type as in PR version of APP.

### E_EEPROM_XMC1
- E_EEPROM_XMC1_ReadBlockWithCRC() API name changed to E_EEPROM_XMC1_VerifyBlockCRCStatus()

Change to new API name.

### GLOBAL_CCU4
- API name and internal functionality changed for sync start. SyncStart_TriggerHigh to _SyncStartTriggerHigh

Change to new API name.

### GLOBAL_CCU8
- API name and internal functionality changed for sync start. SyncStart_TriggerHigh to _SyncStartTriggerHigh

Change to new API name.

### PWM
- 1. API PWM_AcknowledgeInterrupt change to PWM_ClearEvent
- 2. PWM_AcknowledgeInterrupt(), PWM_GetInterruptStatus() and PWM_SetPassiveLevel() API prototype modified for the new enum.

Change to new API name and usage of enum items.

### PWM.CCU4
- API PWM.CCU4.AcknowledgeInterrupt changed to PWM.CCU4_ClearEvent

Change to new API name.

### PWM.CCU8
- API PWM.CCU8.AcknowledgeInterrupt changed to PWM.CCU8_ClearEvent

Change to new API name.

### TIMER
- API TIMER_AcknowledgeInterrupt changed to TIMER_ClearEvent

Change to new API name.

---

Note:
In the migration report, the changes related to GUI widget, resource group, virtual signals can be ignored. User need to take corrective actions in case of failures due to removal of user configurations for UI data, Pins, HW resource instance, signal connection.

The build errors will indicate that these modified source code items are used in user application code. User need to correct it by referring the above table and revision history of source files.

In cases of incompatibilities due to API changes, API signature change, data names or type changes and there is no build error, no changes in the application code are required because then the application code doesn’t uses features that are not compatible any more.
3. Detailed incompatibilities of the XMC Lib v1.00 and v2.0.0 and required actions to use the application code developed with XMC Lib v1.0.0 with XMC Lib 2.0.0.

When using the DAVE APPs Upgrade functionality the new project contains then already the new XMC Lib v2.0.0 also the new version of the DAVE APPs adapted to the incompatible changes to the XMC Lib if required.

If the user code in the project also uses the above mentioned peripheral drivers some manual adjustments as described in below table are required.

In case the source project does not included DAVE APPs and hence DAVE APP Upgrade functionality has not been performed, the imported project still contains the XMC Lib v1.0.0. If in such a case the XMC Lib v2.0.0 should be used we recommend creating a new project for the used target MCU and copying the user code of the DAVE v4.0.0 project into the new project. After that the below described manual actions should be performed if the respective peripheral library is used.

<table>
<thead>
<tr>
<th>XMC Libs</th>
<th>Detailed Descriptions of Changes that may cause incompatibilities</th>
<th>Manual project upgrade steps</th>
</tr>
</thead>
</table>
| ACMP     | 1. Split XMC_ACMP_SetInput() API into 3 APIs. XMC_ACMP_EnableReferenceDivider() and DisableReferenceDivider and XMC_ACMP_SetInput().  
2. default enable functionality of comparator from XMC_ACMP_Init() API removed  
3. Enumeration values changed. | 1. change API as in new version of XMC™ Lib  
2. Enable ACMP by respective API  
3. change enumeration value |
| CAN      | 1) XMC_CAN_MO_Config() signature has changed ( 2 arguments removed, default values used)  
2) Setting TXRQ bit in XMC_CAN_TXFIFO_ConfigMOSlaveObject() has removed  
3) XMC_CAN_STATUS_t enum structure has modified (order of 2 member changed) | 1) remove the two relevant arguments when using the API  
2) use XMC_CAN_MO_SetStatus API to set TXRQ  
3 change only required if compare against value is used (instead against enum member) |
| DMA      | Source and destination address type has changed from uint32_t* -> uint32_t in XMC Lib data structure | Change data type for source and destination address definition |
| HRPWM    | Renaming the enum item: From XMC_HRPWM_SHADOW_TX_t -> XMC_HRPWM_SHADOW_TX_DAC_t | Change to new enum data type |

Legal Disclaimer The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics. With respect to any examples or hints given herein, any typical values stated herein and/or any information regarding the application of the device, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation, warranties of non-infringement of intellectual property rights of any third party.

Information For further information on technology, delivery terms and conditions and prices, please contact the nearest Infineon Technologies Office (www.infineon.com).

Warnings Due to technical requirements, components may contain dangerous substances. For information on the types in question, please contact the nearest Infineon Technologies Office. Infineon Technologies components may be used in life-support devices or systems only with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.