AN11073

Using LPC11Axx EEPROM (with IAP)

Rev. 1 — 1 May 2012 **Application note**

Document information

Info	Content
Keywords	LPC11A02UK ; LPC11A04UK; LPC11A11FHN33; LPC11A12FHN33; LPC11A12FBD48; LPC11A13FHI33; LPC11A14FHN33; LPC11A14FBD48; LPC11Axx, EEPROM, IAP, AN11073
Abstract	This application note will detail how to use IAP commands to access and modify the internal non-volatile EEPROM of the LPC11Axx device family.

Revision history

Rev	Date	Description
1	20120501	Initial version.

Contact information

For more information, please visit: http://www.nxp.com

For sales office addresses, please send an email to: salesaddresses@nxp.com

1. Introduction

The LPC11Axx devices can be equipped with up to 4 kB[1] of Electrically Erasable Programmable Read-Only Memory (EEPROM). Unlike the flash memory typically used to store a program image, the EEPROM can be modified in more discrete sizes – down to single byte operations.

EEPROM is commonly used for storing small amounts of data that are used by an application. This can include:

- Calibration information
- A device's hardware configuration settings
- Operating lifetime of a device
- Error logging information
- Service history
- Serial numbers

By integrating EEPROM memory into the LPC11Axx, two major benefits are achieved: reduced BOM on PCBs, and a radically simplified interface for software. A secondary benefit of having a reduced BOM is that designs which use the LPC11Axx can also meet more strenuous size constraints.

To simplify the use of the EEPROM, the LPC11Axx devices feature functionality stored in ROM to enable users to quickly and easily integrate EEPROM into their applications. By using IAP routines, users aren't forced to learn the details of the EEPROM peripheral, and can implement their design requirements in near record time.

2. In-Application Programming – EEPROM functions

The LPC11Axx User manual (UM10527) details the specifics of the IAP calls. Please refer to UM10527 for information about error codes, and argument formatting. The two operations supported by EEPROM are reading and writing. As such, there are two functions that can be used which make use of the IAP calls to read and write.

2.1.1 IAP calling convention

In order to use IAP, a function pointer should be declared which points to the IAP table entry in ROM. This is a relatively simple process, and can be done by using the following lines of code:

```
#define IAP_LOCATION 0x1FFF1FF1

typedef void (*IAP)(unsigned int command[],
                    unsigned int result[]);

static const IAP iap_entry = (IAP) IAP_LOCATION;
```

Fig 1. Definition of IAP function pointer

The symbol *iap_entry* is used in the EEPROM read and write operations that will be detailed below.

1. On fully featured devices, the uppermost 64 bytes are reserved, thus the effective EEPROM size is limited to 4032 bytes.

2.1.2 Reading EEPROM

A simple IAP call has been created to facilitate reading multiple bytes of EEPROM data and storing them into a buffer. A CMSIS compliant wrapper routine can be seen in Fig 2. Because the IAP routines internally control EEPROM timing they use the CMSIS variable *SystemCoreClock*. For applications which are not CMSIS compliant, the last operand of the IAP call should be the core operating frequency in kHz units.

```
void readEEPROM( uint8_t* eeAddress,
                 uint8_t* buffAddress,
                 uint32_t byteCount )
{
    unsigned int command[5], result[4];

    command[0] = 62;
    command[1] = (uint32_t) eeAddress;
    command[2] = (uint32_t) buffAddress;
    command[3] = byteCount;
    command[4] = SystemCoreClock/1000;

    /* Invoke IAP call...*/
    iap_entry( command, result);

    if (0 != result[0])
   {
     //Trap error
     while(1);
   }
    return;
}
```

Fig 2. Routine to invoke IAP command to read EEPROM

2.1.3 Writing EEPROM

A similar IAP call can be used to store data from RAM or flash into EEPROM. Please note that writing to EEPROM does not require an erase cycle; data can be directly written using the IAP command. Again, a simple wrapper routine can be seen in Fig 3.


```
void writeEEPROM( uint8_t* eeAddress,
                  uint8_t* buffAddress,
                  uint32_t byteCount )
{
    unsigned int command[5], result[4];

    command[0] = 61;
    command[1] = (uint32_t) eeAddress;
    command[2] = (uint32_t) buffAddress;
    command[3] = byteCount;
    command[4] = SystemCoreClock/1000;

    /* Invoke IAP call...*/
    iap_entry( command, result);

   if (0 != result[0])
   {
    //Trap error
    while(1);
   }    return;
}
```

Fig 3. Routine to invoke IAP command to write EEPROM

3. Example application

An example project is included with this application note. It demonstrates how to use the wrapper functions detailed above to perform several operations. It can fill the entire EEPROM memory with 0x00, 0xFF, an incrementing pattern of bytes, or the value of a free running timer. For illustrative purposes a portion of the example source is shown in Fig 4.

```
switch(cmd)
{
    case CMD_FF:
        eefill(0xFF);
        eeDump();
        break;

    case CMD_00:
        eefill(0x00);
        eeDump();
        break;
    …
    case CMD_TIME:
        for (i=0; i<EE_SIZE; i+=4)
        {
            LPC_TMR32B0->TCR = 0;
            hold = LPC_TMR32B0->TC;
            LPC_TMR32B0->TCR = 1;
            writeEEPROM( (uint8_t*) i, (uint8_t*) &hold, 4 );
        }
        LPC_TMR32B0->TCR = 2;
        eeDump();
        break;
    …
    }
    …
```

Fig 4. A portion of code to fill EEPROM with different patterns.

4. Conclusion

By applying the concepts outlined in this application note users of the LPC11Axx family of Cortex-M0 devices will be able to rapidly implement designs which make use of the on-chip EEPROM. The EEPROM features of LPC11Axx can open new doors in designs by enabling developers to quickly and easily store and update non-volatile memory on a byte at a time basis.

5. Legal information

5.1 Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

5.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Evaluation products — This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer.

In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out the use of or inability to use the product, whether or not based on tort (including negligence), strict liability, breach of contract, breach of warranty or any other theory, even if advised of the possibility of such damages.

Notwithstanding any damages that customer might incur for any reason whatsoever (including without limitation, all damages referenced above and all direct or general damages), the entire liability of NXP Semiconductors, its affiliates and their suppliers and customer's exclusive remedy for all of the foregoing shall be limited to actual damages incurred by customer based on reasonable reliance up to the greater of the amount actually paid by customer for the product or five dollars (US$5.00). The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails of its essential purpose.

5.3 Trademarks

Notice: All referenced brands, product names, service names and trademarks are property of their respective owners.

6. List of figures

7. Contents

Date of release: 1 May 2012
Document identifier: AN11073