LGT8FX8P Series – Programming Manual 1.0.4J
Errata Page 17
Jake 2019 Oct 19 4:45pm
Discrepancy My edited text ATMEL text Original+Google Translate
---

The LGT8FX8P series of microcontrollers have 3 blocks of register addresses in the first 256 bytes of RAM data memory. The first block section is the Register File. This is the 32 Special Function Register, and is addressed from 0x00 to 0x1F. The next 64 addresses 0x20 to 0x5F are the Direct I/O Register. The third section is the Extended I/O Register. This occupies from 0x60 to 0xFF. In decimal this breaks down to addresses 0 to 31, 32 through 95, and 96 through 255. The 32x8 Register File has already been described in the CPU Architecture section beginning on page 11. The 64 Direct I/O Registers can be accessed via the faster IN/OUT instructions. The following 160 Extended I/O Registers from 0x60-0xFF are mapped to the data storage space. These registers are only accessible through the slower data addressing commands such as ST/STS/STD and LD/LDS/LDD.

NOTE: major rewrite here. I was initially confused while trying to figure out what either datasheet was trying to communicate when I read this for the first time a couple of weeks ago. If we're going to call this an "overview" or "introduction" at least answer the most basic questions of what are we talking about, where does it begin and end, and what the total size is from the start.
The ATmega48P/88P/168P/328P is a complex microcontroller with more peripheral units than can be supported within the 64 locations reserved in the Opcode for the IN and OUT instructions. For the Extended I/O space from 0x60 - 0xFF in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.

The LGT8FX8P family of microcontrollers is a relatively complex microcontroller that supports many different types of peripherals whose controllers are allocated in 64 I/O register spaces. It can be accessed directly via the IN/OUT instruction. The control registers of other peripherals are allocated in the 0x60~0xFF area. Since this part of the space is mapped into the data storage space, it can only be accessed by commands such as ST/STS/STD and LD/LDS/LDD.
LGT8FX8P 系列微控制器是一种相对复杂的微控制器,它支持多种不同类型的外设,这些外设的控制器被分配在 64 个 I/O 寄存器空间内。可以直接通过 IN/OUT 指令访问。另一些外设的控制寄存器分配在 0x60~0xFF 区域内,由于这部分空间是映射到数据存储空间内,只能通过 ST/STS/STD 以及 LD/LDS/LDD 等指令访问。

The first 256 addresses are directly followed by a maximum of 2K bytes of data SRAM. The portion of the space from 0x4000 to the end of 0xBFFF maps the FLASH program memory location.

(this is the area they were replacing but it is completely rewritten)
The lower 768/1280/1280/2303 data memory locations address both the Register File, the I/O memory, Extended I/O memory, and the internal data SRAM. The first 32 locations address the Register File, the next 64 location the standard I/O memory, then 160 locations of Extended I/O memory, and the next 512/1024/1024/2048 locations address the internal data SRAM.

The system data storage space of LGT8FX8P starts from 0 address and maps common working register file, I/O space, extended I/O space and internal data SRAM space. The first 32 byte addresses correspond to the 32 general working registers of the LGT8XM core. The next 64 addresses are standard I/O spaces that can be accessed directly by the IN/OUT instruction. The next 160 addresses are the extended I/O space, followed by a maximum of 2K bytes of data SRAM. The portion of the space from 0x4000 to the end of 0xBFFF maps the FLASH program memory location.
LGT8FX8P 的系统数据存储空间从 0 地址开始,分别映射了通用工作寄存器文件,I/O 空间,扩展 I/O 空间以及内部数据 SRAM 空间。最开始的 32 个字节地址对应 LGT8XM 内核 32 个通用工作寄存器。接下来的 64 个地址是可以通过 IN/OUT 指令直接访问的标准 I/O 空间。然后的 160 个地址是扩展 I/O 空间,在接下来就是最多 2K 字节的数据 SRAM。从 0x4000 开始到 0xBFFF 结束的这部分空间,映射了 FLASH 程序存储单元。

The 1KB (LGT8F88P/LGT8F168P) or 2KB (LGT8F328P) of SRAM is mapped to two separate locations. The first space from 0x0100 to 0x0900 is read and written by the CPU in 8-bit bytes. The second SRAM space starting from 0x2100 to 0x2900 is a 16-bit wide access space. The system RAM is mapped to the high-order address starting from 0x2100 and is designed to work with the uDSU module to achieve efficient 16-bit data storage. When programming, the normal 8-bit address is offset by 0x2000 to switch to 16-bit access mode.

(obviously this ain’t in grandpa’s Atmel datasheet)

The 1K/2K byte SRAM in the system is mapped to two spaces, respectively. This space from 0x0100 to 0x0900 is read and written by the kernel in 8-bit bytes. Starting from 0x2100 to 0x2900, this area is a 16-bit wide access space. The system RAM is mapped to the high-order address starting from 0x2100 and is mainly used to work with the uDSU module to achieve efficient 16-bit data storage. When programming, the normal 8-bit addressing variable address is offset by 0x2000 to switch to 16-bit access mode.
系统内 1K/2K 字节 SRAM 被分别映射到两个空间。从 0x0100 开始到 0x0900 结束的这个空间被内核以 8 位字节的宽度读写。从 0x2100 开始到 0x2900 结束这个区域为 16 位宽度的访问空间。系统 RAM 被映射到 0x2100 开始的高位地址主要用于配合 uDSU 模块工作,实现高效的 16 位数据存储。在编程时,将普通的 8 位寻址变量地址加上 0x2000 的偏移量,即可切换到 16 位访问模式。

{page 18 for continuity}

The five different addressing modes for the data memory are: Direct, Indirect, Indirect with Displacement, Indirect with Pre-decrement, and Indirect with Post-increment. In the Register File, registers R26 to R31 are the indirect addressing pointer registers. *Indirect* access can address the entire data storage space.

The Indirect with Displacement addressing mode incorporates a 6 bit address as part of the the instruction word. These six bits are added to the the address given by the Y or Z Registers to access up to 63 address locations from the Y or Z Register base address.

When using register indirect addressing modes with automatic pre-decrement and post-increment, the address registers X, Y, and Z are decremented or incremented. Please refer to the instruction set description section for details.

The five different addressing modes for the data memory cover: 1Direct, 2Indirect with Displacement, 3Indirect, 4Indirect with Pre-decrement, and 5Indirect with Post-increment. In the Register File, registers R26 to R31 feature the indirect addressing pointer registers.
The direct addressing reaches the entire data space.
The Indirect with Displacement mode reaches 63 address locations from the base address given by the Y- or Z-register.
When using register indirect addressing modes with automatic pre-decrement and post-increment, the address registers X, Y, and Z are decremented or incremented.
The 32 general purpose working registers, 64 I/O Registers, 160 Extended I/O Registers, and the 512/1024/1024/2048 bytes of internal data SRAM in the ATmega48P/88P/168P/328P are all accessible through all these addressing modes. The Register File is described in ”General Purpose Register File” on page 11.

Notes: I could certainly be wrong here but I think this (direct/indirect) may be a mistake in the Atmel datasheet that was caught and corrected in the LGT version or this is an intentional difference. I’m not sure. I’m going with the Indirect version though.
The system supports five different addressing modes that can cover the entire data space: 1direct access, 2indirect access with offset, 3indirect access, 4indirect access to the decremented address before access, and 5indirect access to the incremental address after access. The general working registers R26 to R31 are used for indirect access to the address pointer. Indirect access can address the entire data storage space. Indirect access with offset address can be addressed to 63 address spaces in the vicinity of the Y/Z register.
When using the register indirect access mode that supports address auto-increment/decrement, the address register X/Y/Z is automatically decremented/incremented by hardware before/after the access occurs. Please refer to the instruction set description section for details.
系统支持 5 种不同的寻址模式可以覆盖到整个数据空间:直接访问,带偏移的间接访问,间接访问,访问前递减地址的间接访问,访问后递增地址的间接访问。通用工作寄存器 R26 到 R31 用于间接访问的地址指针。间接访问可以寻址整个数据存储空间。带偏移地址的间接访问能够寻址到以 Y/Z 寄存器为基地址的附近 63 个地址空间。
当使用支持地址自动递增/递减的寄存器间接访问模式,地址寄存器 X/Y/Z 会在访问发生前/后自动由硬件递减/递增。具体请参考指令集描述部分。

The 16-bit register X/Y/Z and its associated automatic addressing mode (increment, decrement) also play a very important role in the 16-bit extended mode. The 16-bit extended mode can use the LD/ST increment/decrement mode to implement auto-increment and decrement addressing with variables. This mode is very effective when performing operations on arrays. For details, please refer to the relevant chapter, "Digital Operational Accelerator (uDSU)".

(obviously this ain’t in grandpa’s Atmel datasheet)

The 16-bit register X/Y/Z and its associated automatic addressing mode (increment, decrement) also play a very important role in the 16-bit extended mode. The 16-bit extended mode can use the LD/ST increment/decrement mode to implement auto-increment and decrement addressing with variables. This mode is very effective when performing operations on arrays. For details, please refer to the relevant chapter of "Digital Operational Accelerator (uDSU)".

16 位寄存器 X/Y/Z 以及与之相关的自动寻址模式(递增、递减),在 16 位扩展模式下也有着非常重要的作用。16 位扩展模式可以使用 LD/ST 的递增/递减模式,实现带变量的自动递增、递减寻址。这种模式在对数组进行运算操作时,将非常有效。具体实现请参考“数字运算加速器(uDSU)”相关章节。