4.5.5.3 EEPROM ACCESS FUNCTIONS

The library functions `eeprom_read()` and `eeprom_write()`, can be called to read from, and write to, the EEPROM during program execution.

These functions are available for all Mid-range devices that implement EEPROM (described in Section "eeprom_read" and Section "eeprom_write").

For convenience, the macro `_EEPROMSIZE` represents the number of bytes of EEPROM available on the target device.

4.5.5.4 EEPROM ACCESS MACROS

Macro versions of the EEPROM access functions are also provided (described in Section "EEPROM_READ (macro)" and Section "EEPROM_WRITE (macro)").

4.5.6 Variables in Registers

With MPLAB XC8, there is no direct control of placement of variables in registers. The `register` keyword (which can only be used with `auto` variables) is silently ignored and has no effect on the allocation of variables.

Some arguments are passed to functions in the W register rather than in a memory location; however, these values will typically be stored back to memory by code inside the function so that W can be used by code associated with that function. See Section 4.8.6 "Function Parameters" for more information as to which parameter variables can use registers.

4.5.7 Dynamic Memory Allocation

Dynamic memory allocation, (heap-based allocation using `malloc`, etc.) is not supported on any 8-bit device. This is due to the limited amount of data memory available, the memory banks which divide the memory, and the wasteful nature of dynamic memory allocation.

4.5.8 Memory Models

MPLAB XC8 C Compiler does not use fixed memory models to alter allocation of variables to memory. Memory allocation is fully automatic and there are no memory model controls.