|  |  |
| --- | --- |
| osThreadTerminate(NULL);  osThreadTerminate(myTask03Handle); | Позволяет уничтожить задачу, которая выполняется (NULL) или ту на которую указывает хэндл. |
| osThreadDef(myTask03, StartTask03, osPriorityIdle, 0, 128);  myTask03Handle = osThreadCreate(osThread(myTask03), NULL); | Создает задачу:   1. Объявляем   #define osThreadDef(name, thread, priority, instances, stacksz)  Name – имя задачи  Thread – указатель на адрес функции в памяти (название ф-ции)  Priority – начальный приоритет  Instances – количество возможных экземпляров потока (0 – не ограничено)  Stacksz – Размер стека в байтах   1. Создаем экземпляр задачи   osThreadId osThreadCreate (const osThreadDef\_t \*thread\_def, void \*argument)  osThreadDef\_t – имя потока  argument – аргумент |
| osThreadList((unsigned char \*)str\_buf); | Вернет в массив str\_buf информацию о текущих процессах Работает только при включенных режимах отладки операционной системы, конкретно  GENERATE\_RUN\_TIME\_STATS  USE\_TRACE\_FACILITY  USE\_STATS\_FORMATTING\_FUNCTIONS  Описание данных: (B : Blocked, R : Ready, D : Deleted, S : Suspended), далее приоритет, далее стек, далее ID задачи |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |