|  |  |  |  |
| --- | --- | --- | --- |
| 1K | 1024 | 0400h | 2^10 |
| 2K | 2048 | 0800h | 2^11 |
| 4K | 4096 | 1000h | 2^12 |
| 8k | 8192 | 2000h | 2^13 |
| 16k | 16384 | 4000h | 2^14 |
| 32k | 32768 | 8000h | 2^15 |
| 64k | 65536 | FFFFh + 1 | 2^16 |

|  |  |
| --- | --- |
| Normal Mode | Stop clock |
| Stop the clock | Calculate ticks/s and number needed to countup |
| Clear flags | Scale for desired time unit((F/(pre))^-1) |
| Clear TOV  (write 1) | 1024 pre -> 15.625 tick/ms; 8 pre -> 0.5 tick/us |
|  | countupVal = desired \* above |
|  | Timer start = (MAX + 1) – (long) countupVal |
|  | (If writing to both registers, point at low one) |
|  | Clear TOV flag (1) |
|  | Start Timer |
|  | Wait for TOV flag  (while( \*TCInterrFlagReg & 0x01) == 0) ) |
|  | Stop Clock |
|  | Clear TOV |

|  |  |
| --- | --- |
| baudVal =(F / (16\*desiredbaud))-1 | kb\_hit: check truth with  ReadDataAvailable flag  (no bool) |
| Set Data Register Empty  and Frame Error bits | getch: read from the data register |
| Clear all other flags | putch: wait for truth of TxBuffEmpty  (while == 0),write to register |
| Only enable Tx and Rx,  disable all interrupts | Tx/Rx data register are same |
| 5?, then 8 bit char |  |
| Asynchronous, disable |  |
| 1 bit stop bit |  |
| Sample on falling, send on rise |  |
| Load baud rate register in pieces  >>8 for H, &0x for LFF |  |

|  |  |
| --- | --- |
| Enable, not start, disable autotrigger, cleaf conv. Complete flag, disable interrupt, prescalar | Start conversion, (clear conversion complete flag first?) |
| Write 1 to prevent switching ADC off, analog comparator mode | Wait for conversion to complete |
| Select internal (5v?), justify, reserved, channel | Sample data registers separately (do high last, sampling high gives signal to move on) |
| Disable all digital, 1 is in use, others to save power | Stop conversion and clear conversion complete flag |