Para ajustar el temporizador de precisión máxima que se utilizaría para medir el tiempo de cálculo que invierte la IA en realizar su jugada se han modificado los valores de los registros rTCFG0 que es el encargado del pre escalado, el cual se ha reducido a 0 para minimizar el periodo de interrupción del temporizador siguiendo la fórmula que posteriormente se mostrará, el divisor (rTCFG1) se ha mantenido como estaba, a .

Fórmula y explicación de la misma:

Timer input clock Frequency = MCLK / {prescaler value + 1} / {divider value}

MCLK es fijo por lo cual solo podemos modificar el pre escalado y el valor del divisor para maximizar la frecuencia y por tanto reducir el periodo de interrupción del temporizador.

Para pasarlo a microsegundos a partir de las interrupciones generadas se ha seguido la tabla que nos proporcionan para el pre escalado y divisor seleccionados ( tabla en la guía).

Se hace una regla de tres, si interrumpe cada vez en 'y' tiempo, 1 microsegundo equivaldrá a 'x' interrupciones.

|  |
| --- |
| void timer2\_inicializar(void) |
|  | { |
|  | /\* Configuraion controlador de interrupciones \*/ |
|  | rINTMSK &= ~(BIT\_GLOBAL | BIT\_TIMER2); // Emascara todas las lineas excepto Timer0 y el bit global (bits 26 y 13, BIT\_GLOBAL y BIT\_TIMER0 están definidos en 44b.h) |
|  |  |
|  | /\* Establece la rutina de servicio para TIMER2\*/ |
|  | pISR\_FIQ = (unsigned\*) timer2\_ISR; |
|  |  |
|  | /\* Configura el Timer2 \*/ |
|  | rTCFG0 |= 0x0000; // PREESCALADO ajusta el preescalado |
|  | rTCFG1 |= 0x00; // DIVISOR selecciona la entrada del mux que proporciona el reloj. La 00 corresponde a un divisor de 1/2. |
|  | rTCNTB2 = 65535;// CADA PAR DE REG valor inicial de cuenta (la cuenta es descendente) |
|  | rTCMPB2 = 12800;// valor de comparación |
|  | /\* establecer update=manual (bit 1) + inverter=on (¿? será inverter off un cero en el bit 2 pone el inverter en off)\*/ |
|  |  |
|  | } |