

```

/* ***** */
/*
/* Nome do arquivo:    tacometro.c
/*
/*
/* Descricao:        Arquivo contendo as funcoes de interface do uC
/*                    com o encoder do kit, para leitura da rotacao do
/*                    cooler
/*
/*
/* Autores:          Gustavo Lino e Giacomo Dollevedo
/* Criado em:        08/05/2020
/* Ultima revisao em: 31/07/2020
/* ***** */

/* REVISÃO: */
/* ALTERADO A LIBERACAO DO CLOCK PARA PORTA "E" ["SCGC6 -> SCGC5"]*/
/* ALTERADA A MASCARA "TPM0_CLOCK_GATE" no board.h */
/* ALTERADO PARA "&=" no CLOCK DIVIDER */
/* ALTERADA A FORMA COM QUE RPM EH CALCULADA*/

```

```

#include "board.h"
#include "tacometro.h"

```

```

/* ***** */
/* Nome do metodo:    tachometer_init
/* Descricao:        Inicializa os registradores para funcionamento do TPM0
/*                    como contador de pulsos
/*
/*
/* Parametros de entrada:  n/a
/*
/*
/* Parametros de saida:   n/a
/*
/* ***** */

```

```

void tachometer_init(){

```

```

/*Liberar Clock para TPM 0*/
SIM_SCGC6 |= TPM0_CLOCK_GATE;

```

```

/*Configurar o divisor de clock em 1*/
TPM0_SC &= CLOCK_DIVIDER_1;

```

```

/*Liberar o Clock para porta E (encoder)*/
SIM_SCGC5 |= PORTE_CLOCK_GATE;

```

```

/*Configurar o pino PTE29 como external clock (ALT4) e o CLKIN0 como entrada*/
PORTE_PCR29 |= MUX_ALT4;
SIM_SOPT4 &= TPM0CLKSEL_AS_CLKIN0;

```

```

/*Configurar contador para clock externo*/
TPM0_SC |= TPM_EXTERNAL_CLOCK;

```

```

}

```

```

/* ***** */
/* Nome do metodo:    tachometer_readSensor
/* Descricao:        Le a velocidade do cooler (RPM) e a retorna
/*
/*
/* Parametros de entrada:  uiPeriod -> periodo da janela de contagem (LPTMR0)
/*
/*
/* Parametros de saida:   Um unsigned int indicando a rotacao (RPM) do cooler
/*
/* ***** */

```

```

unsigned int tachometer_readSensor(unsigned int uiPeriod){

```

```

/*Numero de pulsos contados*/

```

```
unsigned int uiCounted = TPM0_CNT;
/*Reseta o contador*/
TPM0_CNT &= CLEAR_16;

/*7 pas => A cada 7 pulsos contados, temos 1 rotacao completa*/
unsigned int uiRotations = uiCounted/7;

/*Convertendo a leitura na janela para RPM*/
unsigned int uiCoolerRps = uiRotations/(uiPeriod*1000000);
unsigned int uiCoolerRpM = uiCoolerRps*60;

return uiCoolerRpM;

}
```