```
/* fooCfg.c to be included in foo.c */
/* Creation date: Tue Mar  7 18:09:19 MET 2000 */
/* Author: andre */
/* Creation directory: /extension/tools/MAQUETTE/TPRegs */
/* Generator: BIC (Bit-Interface Configurator) v2.6 - 1999 */
/* BIC (Bit-Interface Configurator) has been created and developed
by */
/* Charles ANDRE - I3S Laboratory - University of Nice (F) */
/* e-mail:  andre@unice.fr */


Binary A;
Binary B;
Binary C;

void REG_edit(FILE *f,char format)
{
  char buf[80];
  fprintf(f,"B = %s\n",Binary2text(B,format,buf));
  fprintf(f,"A = %s\n",Binary2text(A,format,buf));
  fprintf(f,"C = %s\n",Binary2text(C,format,buf));
}

void REG_fill()
{
  char buf[80];
  fprintf(stderr,"Give a value for A:");
  fgets(buf,75,stdin);
  text2Binary(buf,&A);
  fprintf(stderr,"Give a value for C:");
  fgets(buf,75,stdin);
  text2Binary(buf,&C);
}

void RTL_params_init()
{
  Binary_init(16);
  RTL_largestSize = 2;
  RTL_nbBin = 7;
  RTL_nbBout = 9;
}

void RTL_get()
{
  Binary tmp;
  if (RTL_read()) {
    RTL_shutdown(); exit(3);
  }
  B = ClientBin[6];
  RTL_status = ClientBin[0];
  tmp = ClientBin[5];
  A = (tmp & 0xe0) | (A & 0xff1f);
  tmp = ClientBin[7]|(ClientBin[8]<<8);
  C = (tmp & 0xf033) | (C & 0xfcc);
}

void RTL_put()
{
  ClientBout[0] = (RTL_status & 0xff) >> 0;
  ClientBout[4] = (A & 0xff) >> 0;
  ClientBout[5] = (C & 0xff) >> 0;
  ClientBout[6] = (C & 0xff00) >> 8;
  if (RTL_write()) {
    RTL_shutdown(); exit(4);
  }
}

void RTL_putRow(Binary b)
{
  A &= ~0x0f;
  A |= (0x0f & (b << 0));
}

Binary RTL_getColumn(void)
{
  Binary tmp;
```

```
  tmp = A & 0xe0;
  tmp >>= 5;
  return tmp;
}


void Sensor_edit(FILE *f,char format)
{
  char buf[80];
  Binary tmp;
  tmp = RTL_getColumn();
  fprintf(f,"Column = %s\n",Binary2text(tmp,format,buf));
}

void Actuator_set(void)
{
  char buf[80];
  Binary tmp;
  fprintf(stderr,"Give a value for Row:");
  fgets(buf,75,stdin);
  text2Binary(buf,&tmp);
  RTL_putRow(tmp);
}
```