**Data transmission**

**top.c**

#include <stdio.h>

#include <stdlib.h>

#include "../Transmitter/tx.h"

#include "../Channel/ch.h"

#include "../Receiver/rx.h"

#define SIZE 10000

int main()

{

int i, txData, rxData;

float txSignal, rxSignal, count = 0;

for (i = 0; i < SIZE; i++) {

txData = rand()%2;

txSignal = TX\_Transmitter(txData);

rxSignal = CH\_Channel(txSignal);

rxData = RX\_Receiver(rxSignal);

count += (txData != rxData);

}

printf("Number of errors = %.2f", count/SIZE);

return 0;

}

**tx.h**

float TX\_Transmitter(int txData);

**tx.c**

float TX\_Transmitter(int txData)

{

return ((txData == 1) ? 1.0 : -1.0);

}

**ch.h**

float CH\_Channel(float txSignal);

**ch.c**

float CH\_Channel(float txSignal)

{

return txSignal;

}

**rx.h**

int RX\_Receiver(float rxSignal);

**rx.c**

int RX\_Receiver(float rxSignal)

{

return ((rxSignal > 0.0) ? 1 : 0);

}

**Collision**

**Collision.c**

#include <stdio.h>

#include "Station.h"

int main()

{

Station s1,s2;

int n1,n2;

Station\_setCW(&s1,8);

Station\_setCW(&s2,8);

n1=Station\_RandomDelay(s1);

n1=Station\_RandomDelay(s2);

if(n1==n2) printf("Collision occurs");

else if (n1<n2) printf("Station 1 transmits");

else (n1<n2) printf("Station 2 transmits");

return 0;

}

**Station.h**

typedef struct Station{

int cw;

} Station;

void Station\_setCW(Station \*s, int n);

int Station\_RandomDelay(Station s);

**Station.c**

#include <stdlib.h>

#include "Station.h"

#define DEFAULT\_CW 15

void Station\_setCW(Station \*s, int n)

{

s->cw=(n>0)?n:DEFAULT\_CW;

}

int Station\_RandomDelay(Station s)

{

return rand()%s.cw;

}

**Debugging**

**debug.h**

#define DEBUG\_MODE 1

#define DEBUG\_BEGIN(a) if (DEBUG\_MODE) if (a) {

#define DEBUG\_END }

#define DEBUG\_PRINT(a) if(a) printf

#define DEBUG\_PAUSE fprintf(stderr,"Press RETURN");getchar();

**main.c**

#include <stdio.h>

#include "debug.h"

int main()

{

int a = 5, b=7,sum=a+b;

#ifdef DEBUG\_MODE

printf("a = %d; b=%d\n", a,b);

#endif

/\*\*/DEBUG\_BEGIN(1)

/\*\*/DEBUG\_BEGIN(1)("a=%d;",a);

/\*\*/DEBUG\_BEGIN(1)("b=%d\n",b);

/\*\*/DEBUG\_END

printf("sum=%d\n",sum);

return 0;

}