```c
/*============================================================================*/
/* Projekt      :   Olli - eine Holzkiste lernt laufen                        */
/*                                                                            */
/* Hardware     :   Infineon C509                                             */
/*                                                                            */
/* Dateiname    :   seriell_olli_seriell.h                                    */
/*                                                                            */
/* Version      :   1.1 vom 13 Januar 2004                                    */
/*                                                                            */
/* Authoren     :   L. Kulf, J. Roos                                          */
/*                                                                            */
/* Dateigroesse :   xxxx kByte                                                */
/*                                                                            */
/* Datei-                                                                     */
/* beschreibung :   Headerfile                                                */
/*                                                                            */
/*                                                                            */
/*                                                                            */
/*============================================================================*/

// Weitere Informationen -------------------------------------------------------

#ifndef seriell_olli_seriell_h
#define seriell_olli_seriell_h

// Allgemeine Typdefinitionen --------------------------------------------------
#define BUFFER_SIZE_S0_IN   12
#define BUFFER_SIZE_S0_OUT  8
#define BUFFER_SIZE_S1_IN   15
#define BUFFER_SIZE_S1_OUT  10

// Prototypen der Funktionen ---------------------------------------------------
void seriell_S0_init (void);
void interpreter_S0_receive(BYTE my_index, BYTE* s0_inbuf);
void seriell_S0_int_transmit(void);
void seriell_S0_int_receive (void);
void seriell_S0_clear_in_buffer(BYTE* s0_inbuf);
void seriell_S0_clear_out_buffer(void);
void seriell_S0_clear_index(void);
void transmit_S0_vorbereitung(unsigned char wert);

void seriell_S1_init (void);
void interpreter_S1_receive(BYTE my_index, BYTE* s1_inbuf);
void seriell_S1_int_transmit(void);
void seriell_S1_int_receive (void);
void seriell_S1_clear_in_buffer(BYTE* s1_inbuf);
void seriell_S1_clear_out_buffer(void);
void seriell_S1_clear_index(void);
void transmit_S1_vorbereitung(unsigned char Byte2,unsigned char Byte3);

unsigned char paritaetsberechnung(unsigned char x,unsigned char y);
void pause (BYTE i);

#endif
// Programmende ----------------------------------------------------------------
```