

DEFINICIONES NECESARIAS PARA LOS PROTOCOLOS QUE SE ANALIZARAN

```
#define MAX_PKT 1024

/* determina el tamaño del paquete
en bytes */

typedef enum {false, true} boolean;          /* tipo booleano */
typedef unsigned int seq_nr;                 /* números de secuencia o
confirmación */

typedef struct {unsigned char data[MAX_PKT];} packet; /*definición de paquete */
typedef enum {data, ack, nak} frame_kind;    /* definición de frame_kind */

typedef struct {
    frame_kind kind;          /* las tramas se transportan en
esta capa */
    seq_nr seq;              /* ¿qué clase de trama es? */
    seq_nr ack;              /* número de secuencia */
                                /*número de confirmación de recepción*/

    packet info;              /* paquete de la capa de red*/
} frame;

/* Espera que ocurra un evento; devuelve el tipo en la variable event. */
void wait_for_event(event_type *event);

/* Obtiene un paquete de la capa de red para transmitirlo por el canal. */
void from_network_layer(packet *p);

/* Entrega información de una trama entrante a la capa de red. */
void to_network_layer(packet *p);

/* Obtiene una trama entrante de la capa física y la copia en r. */
void from_physical_layer(frame *r);

/* Pasa la trama a la capa física para transmitirla. */
void to_physical_layer(frame *s);

/* Arranca el reloj y habilita el evento de expiración de temporizador. */
void start_timer(seq_nr k);

/* Detiene el reloj e inhabilita el evento de expiración de temporizador.*/
void stop_timer(seq_nr k);

/* Inicia un temporizador auxiliar y habilita el evento ack_timeout. */
void start_ack_timer(void);

/* Detiene el temporizador auxiliar e inhabilita el evento ack_timeout. */
void stop_ack_timer(void);

/* Permite que la capa de red cause un evento network_layer_ready. */
void enable_network_layer(void);

/* Evita que la capa de red cause un evento network_layer_ready. */
void disable_network_layer(void);

/* La macro inc se expande en línea: incrementa circularmente k. */
#define inc(k) if (k < MAX_SEQ) k = k + 1; else k = 0
```