

CosechadoraAgro

Introducción teorica

Estados: definición correspondiente

Transicion: definicion correspondiente

Evetos: definicion correspondiente

Acciones: definicion correspondiente

Reset: definicion correspondiente

Descripción del proyecto

En este proyecto se describe el funconamiento de una maquina agro-industrial para la cosecha y plantacion de plantas en un campo. Consta de tres maquinas de estados:

- Máquina A

- Máquina B

- Máquina C

ect etc etc

Eventos, Acciones y Variables

Eventos:

- s_configCosecha
- s_configPlantar
- s_finCosecha
- s_finPlantar
- e_tiempoOn
- s_sinSemillas
- s_agujereado
- s_semillaTirada
- s_agujeroTapado
- s_enPosicion

Acciones:

- t_tiempoOn_001_S()
- t_tiempoOn_001_stop()
- m_apagarLed()
- m_prenderLed()
- m_agujerearPiso()
- m_avanzar()
- m_tirarSemilla()
- m_tapar()

Variables:

- plantaciones

Máquinas de estados**MdE2****Diagrama de estados:**

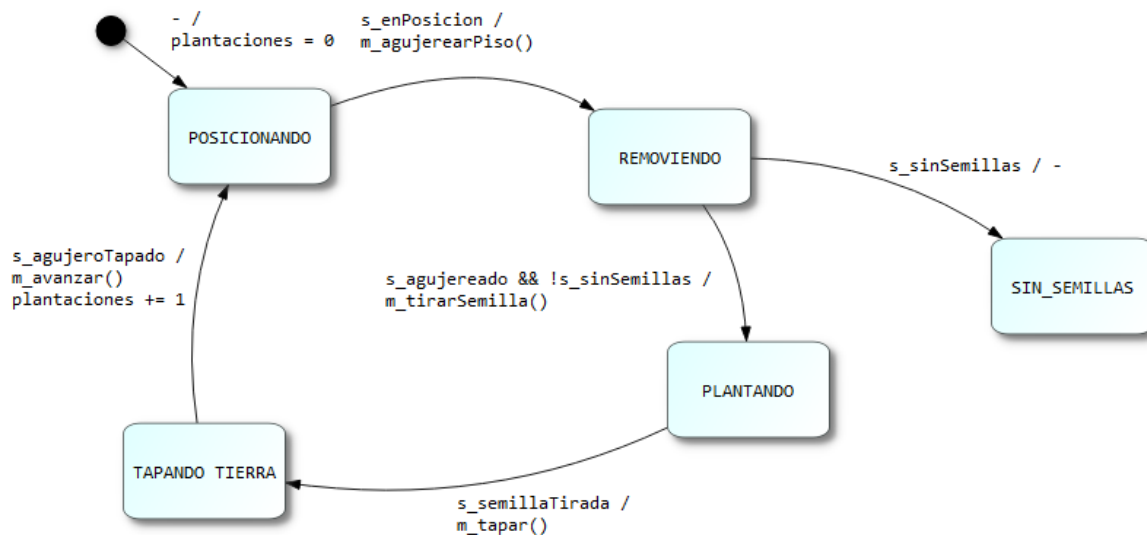


Tabla de estados y transiciones:

| Estado actual | Estado futuro | Eventos | Acciones |
|----------------|----------------|--------------------------------|--------------------------------|
| - | POSICIONANDO | (reset) | plantaciones = 0 |
| TAPANDO TIERRA | POSICIONANDO | s_agujeroTapado | m_avanzar(), plantaciones += 1 |
| POSICIONANDO | REMOVIENDO | s_enPosicion | m_agujerearPiso() |
| REMOVIENDO | PLANTANDO | s_agujereado && !s_sinSemillas | m_tirarSemilla() |
| REMOVIENDO | SIN_SEMILLAS | s_sinSemillas | - |
| PLANTANDO | TAPANDO TIERRA | s_semillaTirada | m_tapar() |

MdE3

Diagrama de estados:

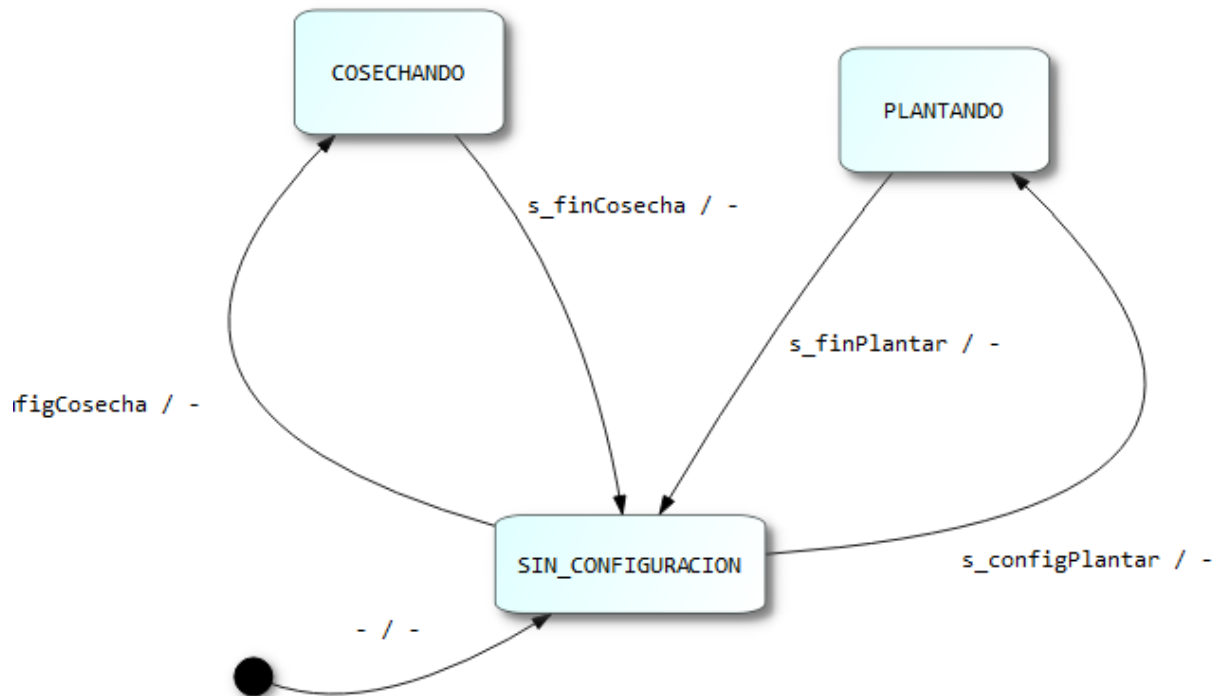


Tabla de estados y transiciones:

| Estado actual | Estado futuro | Eventos | Acciones |
|-------------------|-------------------|-----------------|----------|
| - | SIN_CONFIGURACION | (reset) | |
| COSECHANDO | SIN_CONFIGURACION | s_finCosecha | - |
| PLANTANDO | SIN_CONFIGURACION | s_finPlantar | - |
| SIN_CONFIGURACION | COSECHANDO | s_configCosecha | - |
| SIN_CONFIGURACION | PLANTANDO | s_configPlantar | - |

MdE4

Diagrama de estados:

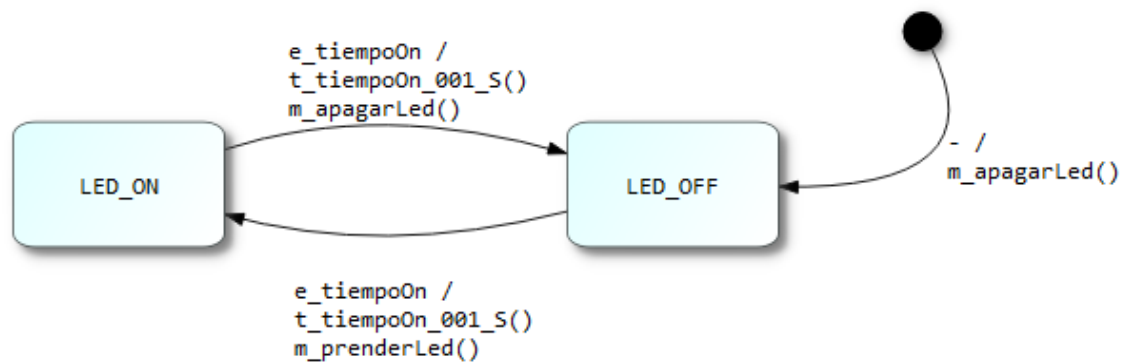


Tabla de estados y transiciones:

| Estado actual | Estado futuro | Eventos | Acciones |
|---------------|---------------|------------|-----------------------------------|
| - | LED_OFF | (reset) | m_apagarLed() |
| LED_ON | LED_OFF | e_tiempoOn | t_tiempoOn_001_S(),m_apagarLed() |
| LED_OFF | LED_ON | e_tiempoOn | t_tiempoOn_001_S(),m_prenderLed() |

Implementación en C:

| AP_FuncionesMDE.c | |
|-------------------|--|
| /** | |
| * | \file AP_FuncionesMDE.c |
| * | \brief Funciones de Eventos y Acciones asociadas a las Maquinas de estados |
| * | \details Descripcion detallada del archivo |

```
*      \author      Almaraz
*      \date   01-05-2023  14:06:29
*/
```

```
/*****
****
```

```
*** INCLUDES
```

```
****
****/
```

```
#include "AP_FuncionesMDE.h"
```

```
/*****
****
```

```
*** DEFINES PRIVADOS AL MODULO
```

```
****
****/
```

```
/*****
****
```

```
*** VARIABLES GLOBALES PUBLICAS
```

```
****
****/
```

```
/*****
****
```

```
*** FUNCIONES GLOBALES AL MODULO
```

```
*****
*****/

/*----- EVENTOS -----
-----*/

/**
 *   \fn      int s_Config ( void )
 *   \brief Coloque aqui su descripcion
 *   \details Amplie la descripcion
 *   \author Almaraz
 *   \date 01-05-2023 14:06:29
 *   \param void
 *   \return int : TRUE y FALSE
 */

int s_Config ( void )
{
    int res = 0 ;

    // Coloque aqui su codigo

    return res ;
}
```

```
/**  
*   \fn      int s_ConfigCosecha ( void )  
*   \brief   Coloque aqui su descripcion  
*   \details Amplie la descripcion  
*   \author   Almaraz  
*   \date    01-05-2023 14:06:29  
*   \param   void  
*   \return  int : TRUE y FALSE  
*/
```

```
int s_ConfigCosecha ( void )  
{  
    int res = 0 ;  
  
    // Coloque aqui su codigo  
  
    return res ;  
}
```

```
/**  
*   \fn      int s_ConfigPlantado ( void )  
*   \brief   Coloque aqui su descripcion  
*   \details Amplie la descripcion  
*   \author   Almaraz
```



```
*      \date 01-05-2023 14:06:29
* \param void
*      \return int : TRUE y FALSE
*/

int s_ConfigPlantado ( void )
{
    int res = 0 ;

    // Coloque aqui su codigo

    return res ;
}

/**
*      \fn      int s_ErrorConfig ( void )
*      \brief Coloque aqui su descripcion
*      \details Amplie la descripcion
*      \author Almaraz
*      \date 01-05-2023 14:06:29
* \param void
*      \return int : TRUE y FALSE
*/
```

```
int s_ErrorConfig ( void )
{
    int res = 0 ;

    // Coloque aqui su codigo

    return res ;
}

/**
 * \fn      int s_ErrorProceso ( void )
 * \brief   Coloque aqui su descripcion
 * \details Amplie la descripcion
 * \author  Almaraz
 * \date    01-05-2023 14:06:29
 * \param   void
 * \return  int : TRUE y FALSE
 */

int s_ErrorProceso ( void )
{
    int res = 0 ;
```

```
// Coloque aqui su codigo

return res ;

}

/**
 * \fn      int s_HaySemillas ( void )
 * \brief   Coloque aqui su descripcion
 * \details Amplie la descripcion
 * \author  Almaraz
 * \date    01-05-2023 14:06:29
 * \param   void
 * \return  int : TRUE y FALSE
 */

int s_HaySemillas ( void )
{
    int res = 0 ;

    // Coloque aqui su codigo

    return res ;
}
```

```
/**  
*   \fn      int s_HayFertilizante ( void )  
*   \brief   Coloque aqui su descripcion  
*   \details Amplie la descripcion  
*   \author   Almaraz  
*   \date    01-05-2023 14:06:29  
*   \param   void  
*   \return  int : TRUE y FALSE  
*/
```

```
int s_HayFertilizante ( void )  
{  
    int res = 0 ;  
  
    // Coloque aqui su codigo  
  
    return res ;  
}
```

```
/**  
*   \fn      int s_TierraRemovida ( void )  
*   \brief   Coloque aqui su descripcion  
*   \details Amplie la descripcion  
*   \author   Almaraz
```

```
*      \date 01-05-2023 14:06:29
* \param void
*      \return int : TRUE y FALSE
*/

int s_TierraRemovida ( void )
{
    int res = 0 ;

    // Coloque aqui su codigo

    return res ;
}

/**
*      \fn      int s_FertilizanteTirado ( void )
*      \brief Coloque aqui su descripcion
*      \details Amplie la descripcion
*      \author Almaraz
*      \date 01-05-2023 14:06:29
* \param void
*      \return int : TRUE y FALSE
*/
```

```
int s_FertilizanteTirado ( void )
{
    int res = 0 ;

    // Coloque aqui su codigo

    return res ;
}

/**
 * \fn      int s_TierraAgujereada ( void )
 * \brief   Coloque aqui su descripcion
 * \details Amplie la descripcion
 * \author  Almaraz
 * \date    01-05-2023 14:06:29
 * \param   void
 * \return  int : TRUE y FALSE
 */

int s_TierraAgujereada ( void )
{
    int res = 0 ;
```

```
// Coloque aqui su codigo

return res ;

}

/**
 * \fn      int s_SemillasPlantadas ( void )
 * \brief   Coloque aqui su descripcion
 * \details Amplie la descripcion
 * \author  Almaraz
 * \date    01-05-2023 14:06:29
 * \param   void
 * \return  int : TRUE y FALSE
 */

int s_SemillasPlantadas ( void )
{
    int res = 0 ;

    // Coloque aqui su codigo

    return res ;
}
```

```
/**
 * \fn      int s_AgujeroTapado ( void )
 * \brief   Coloque aqui su descripcion
 * \details Amplie la descripcion
 * \author  Almaraz
 * \date    01-05-2023 14:06:29
 * \param   void
 * \return  int : TRUE y FALSE
 */

int s_AgujeroTapado ( void )
{
    int res = 0 ;

    // Coloque aqui su codigo

    return res ;
}

/**
 * \fn      int s_PlantacionPendiente ( void )
 * \brief   Coloque aqui su descripcion
 * \details Amplie la descripcion
 * \author  Almaraz
```



```
*      \date 01-05-2023 14:06:29
* \param void
*      \return int : TRUE y FALSE
*/

int s_PlantacionPendiente ( void )
{
    int res = 0 ;

    // Coloque aqui su codigo

    return res ;
}

/**
*      \fn      int s_HayLugar ( void )
*      \brief Coloque aqui su descripcion
*      \details Amplie la descripcion
*      \author Almaraz
*      \date 01-05-2023 14:06:29
* \param void
*      \return int : TRUE y FALSE
*/
```

```
int s_HayLugar ( void )
{
    int res = 0 ;

    // Coloque aqui su codigo

    return res ;
}

/**
 * \fn      int s_FrutaRetirada ( void )
 * \brief   Coloque aqui su descripcion
 * \details Amplie la descripcion
 * \author  Almaraz
 * \date    01-05-2023 14:06:29
 * \param   void
 * \return  int : TRUE y FALSE
 */

int s_FrutaRetirada ( void )
{
    int res = 0 ;
```

```
// Coloque aqui su codigo

return res ;

}

/**
 * \fn      int s_FrutaGuardada ( void )
 * \brief   Coloque aqui su descripcion
 * \details Amplie la descripcion
 * \author  Almaraz
 * \date    01-05-2023 14:06:29
 * \param   void
 * \return  int : TRUE y FALSE
 */

int s_FrutaGuardada ( void )
{
    int res = 0 ;

    // Coloque aqui su codigo

    return res ;
}
```

```
/**
 * \fn      int s_CosechasPendientes ( void )
 * \brief   Coloque aqui su descripcion
 * \details Amplie la descripcion
 * \author  Almaraz
 * \date    01-05-2023 14:06:29
 * \param   void
 * \return  int : TRUE y FALSE
 */
```

```
int s_CosechasPendientes ( void )
{
    int res = 0 ;

    // Coloque aqui su codigo

    return res ;
}
```

```
/*----- ACCIONES -----
-----*/
```

```
/*----- TEMPORIZADORES -----
-----*/
```

AP_FuncionesMDE.h

```
/**

*   \file   AP_FuncionesMDE.h

*   \brief   Prototipos, defines, etc. de los Eventos y Acciones
asociadas a la M de E

*   \details   Descripcion detallada del archivo

*   \author   Almaraz

*   \date    01-05-2023 14:06:29

*/

#ifndef INC_AP_FUNCIONESMDEE_H_
#define INC_AP_FUNCIONESMDEE_H_

/*****
*****

*** INCLUDES GLOBALES

*****
*****/

/*****
*****

*** DEFINES GLOBALES

*****
*****/

#define TRUE 1

#define FALSE 0
```

```
#define          ON          1

#define          OFF         0


/*****
*****

*** VARIABLES GLOBALES PUBLICAS

*****
*****/


/*****
*****

*** PROTOTIPO DE FUNCIONES GLOBALES

*****
*****/


/*----- EVENTOS -----
-----*/


int s_Config ( void );

int s_ConfigCosecha ( void );

int s_ConfigPlantado ( void );

int s_ErrorConfig ( void );

int s_ErrorProceso ( void );

int s_HaySemillas ( void );

int s_HayFertilizante ( void );

int s_TierraRemovida ( void );

int s_FertilizanteTirado ( void );
```

```
int s_TierraAgujereada ( void );  
int s_SemillasPlantadas ( void );  
int s_AgujeroTapado ( void );  
int s_PlantacionPendiente ( void );  
int s_HayLugar ( void );  
int s_FrutaRetirada ( void );  
int s_FrutaGuardada ( void );  
int s_CosechasPendientes ( void );
```

```
/*----- ACCIONES -----  
-----*/
```

```
/*-----TEMPORIZADORES -----  
-----*/
```

```
#endif /* INC_AP_FUNCIONESMDEE_H_ */
```

| |
|----------------------------|
| AP_Inicializacion.c |
|----------------------------|

```
/**  
* \file AP_Inicializacion.c  
* \brief Funciones de inicializacion de todo el proyecto  
* \details Descripcion detallada del archivo  
* \author Almaraz  
* \date 01-05-2023 14:06:29  
*/
```

```

/*****
*****

*** INCLUDES

*****
*****/

#include "AP_Inicializacion.h"

#include "AP_MDE.h"

/*****
*****

*** DEFINES PRIVADOS AL MODULO

*****
*****/

/*****
*****

*** PROTOTIPO DE FUNCIONES PRIVADAS AL MODULO

*****
*****/

static void Inicializar_Cosecha( void ) ;

static void Inicializar_Plantacion( void ) ;

static void Inicializar_General( void ) ;

/*****
*****

*** FUNCIONES PRIVADAS AL MODULO
```



```
*****
*****/

/**
 *  \fn      void Inicializar_Cosecha( void )
 *  \brief   Inicializacion de las diferentes maquinas de estado
 *  \details Amplie la descripcion
 *  \author  Almaraz
 *  \date    01-05-2023 14:06:29
 *  \param   void
 *  \return  void
 */
static void Inicializar_Cosecha( void )
{
    //!< Coloque aqui su codigo

    return ;
}

/**
 *  \fn      void Inicializar_Plantacion( void )
 *  \brief   Inicializacion de las diferentes maquinas de estado
 *  \details Amplie la descripcion
 *  \author  Almaraz
 *  \date    01-05-2023 14:06:29
```

```
* \param void
*      \return void
*/

static void Inicializar_Plantacion( void )
{
    //!< Coloque aqui su codigo

    return ;
}

/**
*      \fn      void Inicializar_General( void )
*      \brief Inicializacion de las diferentes maquinas de estado
*      \details Amplie la descripcion
*      \author Almaraz
*      \date 01-05-2023 14:06:29
*      \param void
*      \return void
*/

static void Inicializar_General( void )
{
    //!< Coloque aqui su codigo

    return ;
}
```

```

/*****
**** FUNCIONES GLOBALES AL MODULO

*****/

/**
 *  \fn      void Inicializar( void )
 *  \brief   Inicializacion de las diferentes maquinas de estado
 *  \details Amplie la descripcion
 *  \author  Almaraz
 *  \date    01-05-2023 14:06:29
 *  \param   void
 *  \return  void
 */
void Inicializar( void )
{
    //Coloque aqui su codigo

    Inicializar_Cosecha( );

    Inicializar_Plantacion( );

    Inicializar_General( );

```

```
        return ;  
    }
```

AP_Inicializacion.h

```
/**  
  
 *   \file   AP_Inicializacion.h  
  
 *   \brief   prototipo de la Funcion general de inicializaciones  
  
 *   \details   Descripcion detallada del archivo  
  
 *   \author   Almaraz  
  
 *   \date    01-05-2023 14:06:29  
  
 */  
  
#ifndef INC_AP_INICIALIZACION_H_  
#define INC_AP_INICIALIZACION_H_  
  
/*****  
*****  
  
*** INCLUDES GLOBALES  
  
*****  
*****/  
  
/*****  
*****  
  
*** DEFINES GLOBALES
```

```
*****
*****/

/*****
*****

*** PROTOTIPO DE FUNCIONES GLOBALES

*****
*****/

void Inicializar( void ) ;

#endif /* INC_AP_INICIALIZACION_H_ */
```

| |
|-----------------|
| AP_MDE.c |
|-----------------|

```
/**

*   \file   AP_MDE.c
*
*   \brief   Maquina/s de Estado
*
*   \details  Descripcion detallada del archivo
*
*   \author   Almaraz
*
*   \date    01-05-2023 14:06:29
*
*/

/*****
*****

*** INCLUDES

*****
*****/

#include "AP_MDE.h"
```

```
#include "AP_FuncionesMDE.h"
```

```

/*****
*****/

```

*** DEFINES PRIVADOS AL MODULO

***** /

```
#define RESET 0
```

```
//Maquina: Cosecha
```

```
#define Reposo 1 // < Maquina: Cosecha >
#define Avanzar 2 // < Maquina: Cosecha >
#define Retirar_Fruta 3 // < Maquina: Cosecha >
#define Depositar_Fruta 4 // < Maquina: Cosecha >
```

```
//Maquina: Plantacion
```

```
#define Reposo 1 // < Maquina: Plantacion >

#define Agujerear_Tierra 2 // < Maquina: Plantacion >

#define Plantar_Semilla 3 // < Maquina: Plantacion >

#define Remover_Tierra 4 // < Maquina: Plantacion >

#define Tirar_Fertilizante 5 // < Maquina: Plantacion >

#define Tapar_Agujero 6 // < Maquina: Plantacion >

#define Avanzar 7 // < Maquina: Plantacion >

#define Cargar_Insumos 8 // < Maquina: Plantacion >
```

```
//Maquina: General
```

```
#define      Esperando Configuracion      1      // < Maquina: General
>

#define      Configurada      2      // < Maquina: General >

#define      Plantacion_de_semilla      3      // < Maquina: General
>

#define      Cosechado      4      // < Maquina: General >


/*****
*****

*** MACROS PRIVADAS AL MODULO

*****
*****/

#define      m_AgujerearTierra()

#define      m_ApagarActuadores()

#define      m_Avanzar()

#define      m_Avanzar()

#define      m_ChequearAlmacenamiento()

#define      m_ChequearConfig()

#define      m_ChequearFertilizante()
```

```
#define      m ChequearSemillas()
```

```
#define      m_Cosechar()
```

```
#define      m GuardarFruta()
```

```
#define      m_LeerConfig()
```

```
#define      m_Plantar()
```

```
#define      m_RemoverTierra()
```

```
#define      m_RetirarFruta()
```

```
#define      m TaparAgujero()
```

```
#define      m TirarFertilizante()
```

```
#define      m_TirarSemillas()
```

```

/*****
*****

```

*** PROTOTIPO DE FUNCIONES PRIVADAS AL MODULO


```
*****
*****/

static int Cosecha ( int );

static int Plantacion ( int );

static int General ( int );


/*****
*****

*** VARIABLES GLOBALES PUBLICAS


*****
*****/


/*****
*****

*** FUNCIONES PRIVADAS AL MODULO


*****
*****/


/**
*   \fn      static int Cosecha ( int Estado )
*
*   \brief   Coloque aqui su descripcion
*
*   \details Amplie la descripcion
*
*   \author   Almaraz
*
*   \date    01-05-2023 14:06:29
*
*   \param   [in] Estado: caso dentro de la maquina (estado actual)
*
*   \return  int : estado siguiente
```

```
*/
```

```
static int Cosecha ( int Estado )  
{  
    switch ( Estado )  
    {  
  
        case RESET :  
            m_ChequearAlmacenamiento();  
            m_LeerConfig();  
  
            Estado = Reposo;  
  
            break;  
  
        case Reposo :  
            if ( s_HayLugar() )  
            {  
                m_RetirarFruta();  
  
                Estado = Retirar_Fruta;  
            }  
  
            break;
```

```
case Avanzar :  
  
    if ( !s_CosechasPendientes() || !s_HayLugar() )  
    {  
  
        Estado = Reposo;  
    }  
  
    if ( s_HayLugar() && s_CosechasPendientes() )  
    {  
  
        m_RetirarFruta();  
  
        Estado = Retirar_Fruta;  
    }  
  
    break;  
  
case Retirar_Fruta :  
  
    if ( s_FrutaRetirada() )  
    {  
  
        m_GuardarFruta();  
  
        Estado = Depositar_Fruta;  
    }  
  
    break;
```

```
case Depositar_Fruta :  
    if ( s_FrutaGuardada() )  
    {  
        m_Avanzar();  
        m_ChequearConfig();  
        m_ChequearAlmacenamiento();  
  
        Estado = Avanzar;  
    }  
  
    break;  
  
default:  
    Estado = RESET ;  
    break;  
  
}  
return Estado ;  
}  
  
/**  
*   \fn      static int Plantacion ( int Estado )  
*   \brief   Coloque aqui su descripcion  
*   \details Amplie la descripcion
```

```
*      \author Almaraz
*      \date 01-05-2023 14:06:29
*      \param [in] Estado: caso dentro de la maquina (estado actual)
*      \return int : estado siguiente
*/

static int Plantacion ( int Estado )
{
    switch ( Estado )
    {

        case RESET :

            m_ApagarActuadores();
            m_ChequearSemillas();
            m_ChequearFertilizante();
            m_ChequearConfig();

            Estado = Reposo;

            break;

        case Reposo :

            if ( s_HaySemillas() && s_HayFertilizante() &&
s_PlantacionPendiente() )
            {

                m_RemoverTierra();
```

```
        Estado = Remover_Tierra;
    }

    if ( !s_HaySemillas() || !s_HayFertilizante() )
    {

        Estado = Cargar_Insumos;
    }

    break;

case Agujerear_Tierra :
    if ( s_TierraAgujereada() )
    {

        m_TirarSemillas();

        Estado = Plantar_Semilla;
    }

    break;

case Plantar_Semilla :
    if ( s_SemillasPlantadas() )
    {
```

```
m_TaparAgujero();

Estado = Tapar_Agujero;
}

break;

case Remover_Tierra :
    if ( s_TierraRemovida() )
    {
        m_TirarFertilizante();

        Estado = Tirar_Fertilizante;
    }

    break;

case Tirar_Fertilizante :
    if ( s_FertilizanteTirado() )
    {
        m_AgujerearTierra();

        Estado = Agujerear_Tierra;
    }
```

```
        break;

    case Tapar_Agujero :
        if ( s_AgujeroTapado() )
        {
            m_Avanzar();
            m_ChequearFertilizante();
            m_ChequearSemillas();

            Estado = Avanzar;
        }

        break;

    case Avanzar :
        if ( s_PlantacionPendiente() && s_HaySemillas() &&
s_HayFertilizante() )
        {
            m_RemoverTierra();

            Estado = Remover_Tierra;
        }

        if ( !s_PlantacionPendiente() )
        {
```



```
        Estado = Reposo;
    }

    if ( !s_HayFertilizante() && !s_HaySemillas() )
    {

        Estado = Cargar_Insumos;
    }

    break;

case Cargar_Insumos :

    if ( s_HaySemillas() && s_HayFertilizante() )
    {

        m_ChequearConfig();

        Estado = Reposo;
    }

    break;

default:

    Estado = RESET ;

    break;
```

```
    }

    return Estado ;
}

/**
 * \fn      static int General ( int Estado )
 * \brief   Coloque aqui su descripcion
 * \details Amplie la descripcion
 * \author  Almaraz
 * \date    01-05-2023 14:06:29
 * \param [in] Estado: caso dentro de la maquina (estado actual)
 * \return int : estado siguiente
 */

static int General ( int Estado )
{
    switch ( Estado )
    {

        case RESET :

            m_ApagarActuadores();

            Estado = Esperando Configuracion;

            break;
    }
}
```

```
case Esperando Configuracion :  
    if ( s_Config() )  
    {  
        m_LeerConfig();  
  
        Estado = Configurada;  
    }  
  
    break;  
  
case Configurada :  
    if ( s_Config()Cosecha )  
    {  
        m_Cosechar();  
  
        Estado = Cosechado;  
    }  
  
    if ( s_Config()Plantado )  
    {  
        m_Plantar();  
  
        Estado = Plantacion_de_semilla;  
    }
```

```
        if ( s_ErrorConfig() )
        {

            Estado = Esperando Configuracion;

        }

        break;

    case Plantacion_de_semilla :

        if ( s_ErrorProceso() )
        {

            m_ApagarActuadores();

            Estado = Esperando Configuracion;

        }

        break;

    case Cosechado :

        if ( s_ErrorProceso() )
        {

            m_ApagarActuadores();

            Estado = Esperando Configuracion;
```

```
    }

    break;

    default:

        Estado = RESET ;

        break;

    }

    return Estado ;
}

/*****
*****

*** FUNCIONES GLOBALES AL MODULO

*****
*****/

/**
 *   \fn      void MaquinaDeEstados ( void )
 *
 *   \brief   Coloque aqui su descripcion
 *
 *   \details Amplie la descripcion
 *
 *   \author   Almaraz
 *
 *   \date    01-05-2023 14:06:29
 *
 *   \param   void
 *
 *   \return  void
```

```
*/  
  
void MaquinaDeEstados ( void )  
{  
  
    static int estados_Cosecha = RESET;  
  
    static int estados_Plantacion = RESET;  
  
    static int estados_General = RESET;  
  
    // Coloque su codigo aqui  
  
    estados_Cosecha = Cosecha( estados_Cosecha );  
    estados_Plantacion = Plantacion( estados_Plantacion );  
    estados_General = General( estados_General );  
  
    return ;  
}
```

| |
|-----------------|
| AP_MDE.h |
|-----------------|

```
/**  
  
*   \file   AP_MDE.h  
  
*   \brief   Prototipos, defines, etc. de las diferentes la M de E  
  
*   \details   Descripcion detallada del archivo  
  
*   \author   Almaraz  
  
*   \date    01-05-2023 14:06:29  
  
*/
```

```
#ifndef INC_AP_MDEE_H_

#define INC_AP_MDEE_H_

/*****
*****

*** INCLUDES GLOBALES

*****
***** /

/*****
*****

*** DEFINES GLOBALES

*****
***** /

/*****
*****

*** VARIABLES GLOBALES PUBLICAS

*****
***** /

/*****
*****

*** PROTOTIPO DE FUNCIONES GLOBALES

*****
***** /
```

```
void MaquinaDeEstados ( void );
```

```
#endif /* INC_AP_MDEE_H_ */
```

| |
|---------------|
| main.c |
|---------------|

```
/**
```

```
*   \file   main.c
```

```
*   \brief   Proyecto: CosechadoraAgro
```

```
*   \details  Descripcion detallada del archivo
```

```
*   \author   Almaraz
```

```
*   \date    01-05-2023 14:06:29
```

```
*/
```

```
/*  
*****
```

```
*** INCLUDES
```

```
*****  
*****/
```

```
#include "AP_MDE.h"
```

```
#include "AP_Inicializacion.h"
```

```
void main ( void )
```

```
{
```

```
    Inicializar( );
```



```
while ( 1 )  
{  
    MaquinaDeEstados();  
}  
return ;  
}
```