Может кому пригодится инициализация GSM модуля SIM900.

Модуль может принимать целый ряд команд в одном пакете.

UART0\_Write\_Text("AT+IPR=9600;+IFC=0,0;+ICF=3,3;\r\n",32); // Задаем параметры UART if(!(Sim\_r\_otv())) stop();

UART0\_Write\_Text("AT+CMGF=1;+CLIP=1;+CNMI=2;E0\r\n",30); // Отключаем эхо, ;E0

// включаем текстовый режим смс, включаем аон, // включение оповещения о смс

if(!(Sim\_r\_otv())) stop();

UART0\_Write\_Text("AT+CMGDA=\"DEL ALL\"\r\n", 20); // Удалям все смс

if(!(Sim\_r\_otv())) stop();

По просьбе Кирилла «Кирилл 10.05.2013 13:14 #

Интерес в основном для меня представляет кусок кода отвечающий за распознование ответов модема. Моя система распознования мне не нравится, ищу идеи получше.»

выкладываю блок чтения популярных ответа от GSM модуля.

Остальные ответы распознаю в тупую

UART0\_Write\_Text("AT+CSQ\r\n",8); // Команда проверки качества связи

UART0\_Load(); // ждем ответ

if( UCSR0A & (1<<RXC0)) // если есть не прочитанные данные

{

rx\_com(); // принимаем их по UART\_0

if(rx\_tx[4] == 'S' && rx\_tx[5] == 'Q') //дешифрирование принятого ответа по 2-м буквам

{

ur\_sig[0] = rx\_tx[8]; // Запись значения уровня связи от 0-31, 99 - нет связи.

ur\_sig[1] = rx\_tx[9];

}

}

/\*=================================================================================

Функция - unsigned char Sim\_r\_otv(void) читаем ответ от sim900

Вход - нет

Выход - 0 - ошибка, 1 - OK, 2 - ERROR, 3 - "> "готовность написания смс

=================================================================================\*/

unsigned char Sim\_r\_otv(void)

{

unsigned char at\_1, at\_2, stat1 = 0,

ot\_rx[40],

ot\_m[8], k\_ot, k\_nn;

if(!UART0\_Load()) return 0;

for(at\_1 = 0; at\_1 < 40; at\_1++)

{

ot\_rx[at\_1] = UART0\_Read(); // Считываем 40 байт

if(ot\_rx[at\_1] == 0) at\_1 = 40; // Если считали 0х00 прекращаем считывать ответ

}

for(at\_2 = 0; at\_2 < 3; at\_2++)

{

switch (at\_2) // Перебираем все возможные ответы

{

case 0: ot\_m[0] = 'O'; ot\_m[1] = 'K'; k\_ot = 2; break; // ОК

case 1: ot\_m[0] = 'E'; ot\_m[1] = 'R'; ot\_m[2] = 'R'; // ERROR

ot\_m[3] = 'O'; ot\_m[4] = 'R'; k\_ot = 5;break;

case 2: ot\_m[0] = '>'; ot\_m[1] = 0x20; k\_ot = 2; break; // >' ' - ответ для написания смс

}

at\_1 = 0;

while(at\_1<40) // Ищим совпадения в принятом массиве

{

if(ot\_rx[at\_1] == ot\_m[0]) // Если мы дошли до текста

{

for(k\_nn = 1; k\_nn < k\_ot; k\_nn++)

{

if(ot\_rx[at\_1 + k\_nn] != ot\_m[k\_nn]) k\_nn = 100;

}

if(k\_nn == k\_ot) return (at\_2 + 1); // Если нашли ответ возвращаем номер его

}

at\_1 ++;

}

}

return 0;

}

/\*=====================================================================================================================

Функция - unsigned char Sim\_out\_sms(char text\_sms[], unsigned char data\_sms, char text\_nom[], unsigned char data\_nom)

отправка смс сообщения

Вход - char text\_sms[] - текст смс

- unsigned char data\_sms - количество символов в смс

- char text\_nom[] - номер отправки смс

- unsigned char data\_nom - количество символов в номере

Выход - 0 - смс не отправленно

- 1 - смс успешно отправленно

=====================================================================================================================\*/

unsigned char Sim\_out\_sms(char text\_sms[], unsigned char data\_sms, char text\_nom[], unsigned char data\_nom)

{

unsigned char n\_p = 0;

while(n\_p < 3) // Три попытки на отправку смс

{

UART0\_Clear(); // Обнуляем остатки пакета(если есть)

UART0\_Write\_Text("AT+CMGS=\"",9); // Команда отправки смс

UART0\_Write\_Text(text\_nom, data\_nom); // Номер отправки смс

UART0\_Write\_Text("\"\r\n",3);

if(Sim\_r\_otv() == 0x03) // Если есть приглашение '>'

{

UART0\_Write\_Text(text\_sms, data\_sms); // Пишем текст смс

UART0\_Write(0x1A); // Отправляем смс

if(Sim\_r\_otv() == 0x01) return 1; // Читаем статус отправки смс

}

n\_p ++;

}

return 0;

}