Version 1.1

eRIC LowPowerMode:

CPU LowPowerMode:

There are generally three CPU Low power modes and three to exit CPU low power modes available on eRIC. When any one of the CPU Low power modes are called, the program execution stops and only runs code which are based on interrupts. So before entering CPU LPM modes, the code which needs to be executed should be in interrupt and enabled. Similarly to exit any CPU Low power mode, the exit function can only be called in interrupt since only interrupts work in CPU LPM modes. When it exits, the program immediately starts from where it entered into CPU LPM mode before.

Refer eRIC_eROS_Developers_Manual_1.2 on wards for complete list of CPU LPM definitions. Refer SLAU259E document by Texas Instruments to use core registers.

The following are the functions which can be used to achieve CPU Lowpowermodes.

eRIC_LPM_Level0();In this level MCLK is turned off and SMCLK, ACLK works normally. This doesn't save much power, but most of the modules can be assessed as they use SMCLK, ACLK as source clocks.

eRIC_LPM_ExitLevel0(); This is used to exit eRIC_LPM_Level0().

eRIC_LPM_Level1();In this level MCLK,SMCLK is turned off and ACLK works normally. So any module like UART,WDT,TimerA0 etc should be sourced from ACLK to run in this mode. Depending on the modules used, the current draw varies but generally it takes 5uA when all radio and other modules are off, as 32768Hz(ACLK) takes ~6uA.

eRIC_LPM_ExitLevel1(); This is used to exit eRIC_LPM_Level1().

eRIC_LPM_Level2();In this level all clocks are Off. So eRIC would be in sleep and will be drawing only 2uA if Radio is off. Radio receiver should be turned off and put in to sleep before using this, as it might generate PUC(reset) or consume more current since all clocks are off. Few Interrupts which are not based on any clocks still work, so interrupts can be used to exit this LPM mode.

eRIC_LPM_ExitLevel2(); This is used to exit eRIC_LPM_Level2().

Code Example:

```
#include<cc430f5137.h>
#include"eRIC.h"
int main(void)
{
       eRIC_WDT_Stop();                    //Stops Watch dog timer
       eRIC_GlobalInterruptDisable();     //Global interrupt is stopped
       eROS_Initialise(434000000);        //Eros is initialised with 434MHz
       eRIC_Power = 12;                   //eRIC Power is set to 12 dbm
       eRIC_TxPowerLevel = 8;             //Rx disable and radio sleep
       eRIC_RxPowerLevel = 8;             //Tx disable
       eRIC_RadioUpdate();                //Radio is updated
        Pin1_PullDownEnable();           //Pin1 is pulled down
       Pin1_InterruptLow2High();          //Pin1 interrupt direction is
   selected low to high
       Pin1_InterruptEnable();           //Pin1 interrupt is enabled
       eRIC_RadioTx_Buffer[0] = 'L';      //Radio transmit packet is LPM2
       eRIC_RadioTx_Buffer[1] = 'P';
       eRIC_RadioTx_Buffer[2] = 'M';
       eRIC_RadioTx_Buffer[3] = '2';
       eRIC_GlobalInterruptEnable();     //Global interrupts enabled
       while(1)
       {
             eRIC_RadioTx_BuffCount = 4;       //Radio packet size is 4
             eRIC_RfSenddata();                //Packet is sent over air
             eRIC_LPM_Level2();                //Enters low power mode 2
       }
}

#pragma vector= PORT2_VECTOR
 __interrupt void PORT2_ISR(void)
 {
        if(P2IV==0x0A)
        {
              Pin1_ClearInterruptFlag();       //Clears Pin1 interrupt flag
              eRIC_LPM_ExitLevel2();           //Exits the Low power mode2
   and continuous code in main
        }
  }

void eRIC_RfDataReceivedInterrupt()   //V1.5.4 Add code here to deal with
   available received data..This is triggered when interrupt is enabled and a
   packet is received
 {
  }
```

The above example will transmit “LPM2” message whenever Pin1 is pulled Low- high and goes back into Low power mode 2 which consumes only 2uA.

Line1 and line2 includes cc430F5137 and eRIC.h , which is must for any program code. Main starts at line3.

Watchdogtimer is stopped in Line5. Global interrupts are disabled in Line6, any interrupts even radio interrupts in eros will be disabled.

Radio is initialised with 434MHz, 12 dbm power in Line7 and 8.

Rx is disabled in Line9 and radio is put into sleep in Line10 when radio updates in Line11.

Pin1 interrupt is enabled with low to high direction in Line 12-14.

Message “LPM2” is loaded into Radio tx buffer in Line 15-18.

eRIC enters in to LPM2 at Line24 after transmitting a packet “LPM2” and program execution stops here.

Low power mode 2 exits when Pin1 is held Low-high in Line 28 - 36 and transmits data and goes back into LPM2 drawing only 2uA.

eRIC_RfDataReceivedInterrupt() is copied from eRIC.c which is removed and pasted in main at line 38-40. This has no effect in this example as there is no receiver enabled. But this code is needed for compiler to compile without error or un-remove this same code in eRIC.c.

RadioLowPowerMode:

There are two variables which defines Radio Low power mode.

1) eRIC_RXPowerLevel :
2) eRIC_TXPowerLevel:

There are 9(eRIC_RXPowerLevel = 0-8) different Radio receiver power saving levels for which there are 9(eRIC_TXPowerLevel = 0-8)transmitter levels correspondingly which are explained below.

eRIC_RXPowerLevel = 0; Radio receiver is always On. This draws a continuous current of 16mA

eRIC_RXPowerLevel = 1; Radio receiver is 12.5% On. This draws an average current of 2mA

eRIC_RXPowerLevel = 2; Radio receiver is 6.25% On. This draws an average current of 1mA

eRIC_RXPowerLevel = 3; Radio receiver is 3.13% On. This draws an average current of 500uA

eRIC_RXPowerLevel = 4; Radio receiver is 1.56% On. This draws an average current of 250uA

eRIC_RXPowerLevel = 5; Radio receiver is 0.78% On. This draws an average current of 125uA

eRIC_RXPowerLevel = 6; Radio is 0.39% On. This draws an average current of 62.5uA

eRIC_RXPowerLevel = 7; Radio is 0.2% On. This draws an average current of 32uA

eRIC_RXPowerLevel = 8; Radio is Off.

eRIC_TXPowerLevel is chosen in such a way that it is always greater than or equal to eRIC_RXPowerLevel.

eRIC_TXPowerLevel>=eRIC_RXPowerLevel

For example if eRIC_RXPowerLevel is set at 5 on receiver and if transmitter eRIC_TXPowerLevel is set at 4, then receiver will not receive data. For successful communications, transmitter eRIC_TXPowerLevel should be set at 5 or above or receiver eRIC_RXPowerLevel should be set at 4 or below.

The exact timings of Receiver being On/Off in different eRIC_RXPowerLevels and at different over-air baud rates are explained below.

The below table explains eRIC_RXPowerLevel settings								
Receiver On, On+Off time in Seconds								
RfBaud\RxPoweravinglevel	**RX ON time(25bits +400 us)**	**12.5%(eRIC_RXPowerLevel = 1)**	**6.25%(eRIC_RXPowerLevel = 2)**	**3.13%(eRIC_RXPowerLevel = 3)**	**1.56%(eRIC_RXPowerLevel = 4)**	**0.78%(eRIC_RXPowerLevel = 5)**	**0.39%(eRIC_RXPowerLevel = 6)**	**0.2%(eRIC_RXPowerLevel = 7)**
1200(eRIC_RfBaudRate = 0)	0.021233	0.169867	0.339733	0.678381	1.361111	2.722222	5.444444	10.616667
2400(eRIC_RfBaudRate = 1)	0.010817	0.086533	0.173067	0.345580	0.693376	1.386752	2.773504	5.408333
4800(eRIC_RfBaudRate = 2)	0.005608	0.044867	0.089733	0.179180	0.359509	0.719017	1.438034	2.804167
9600(eRIC_RfBaudRate = 3)	0.003004	0.024033	0.048067	0.095980	0.192575	0.385150	0.770299	1.502083
19200(eRIC_RfBaudRate = 4)	0.001702	0.013617	0.027233	0.054380	0.109108	0.218216	0.436432	0.851042
38400(eRIC_RfBaudRate = 5)	0.001051	0.008408	0.016817	0.033580	0.067374	0.134749	0.269498	0.525521

76800(eRIC_RfBaudRate = 6)	0.000726	0.005804	0.011608	0.023180	0.046508	0.093015	0.186031	0.362760
100000(eRIC_RfBaudRate = 7)	0.000650	0.005200	0.010400	0.020767	0.041667	0.083333	0.166667	0.325000
250000(eRIC_RfBaudRate = 8)	0.000500	0.004000	0.008000	0.015974	0.032051	0.064103	0.128205	0.250000
500000(eRIC_RfBaudRate = 9)	0.000450	0.003600	0.007200	0.014377	0.028846	0.057692	0.115385	0.225000

The Receiver On time for a particular baud rate is calculated based on sum of 25Bits and 400uS.

For example at 1200 Baudrate:

25bits +400us = ((25*1000000)/1200)+400 = 20833.333uS +400uS = 21233.333uS

So 21233.333uS is the fixed receiver On time at 1200 baud rate for different eRIC_RXPowerLevels (1-7).

So at eRIC_RXPowerLevel = 1, the On time is 12.5% duty cycle, which would mean receiver is off for 87.5%.

So receiver off time at eRIC_RXPowerLevel = 1 is (87.5*21233.333)/12.5 = 148633.331.

So total time between receiver coming on or Receiver On+Off time is 21233.333+148633.331 = 169866.664.

The receiver comes On every 169866.664uS for 21233.333uS and if it detects any transmitter preamble then receiver will stay On until it receives complete packet. So for the receiver to receive a data from transmitter, the transmitter preamble should be atleast 169866.664 uS. Choosing eRIC_TXPowerLevel = 1 at 1200 baud rate ,a preamble time of 190700 uS is added to the transmit packet in eROS. The transmitter preamble time tables are shown further below.

Similarly for 500kHz baud rate:

25bits +400us = ((25*1000000)/500000) + 400 = 50uS +400uS = 450 uS

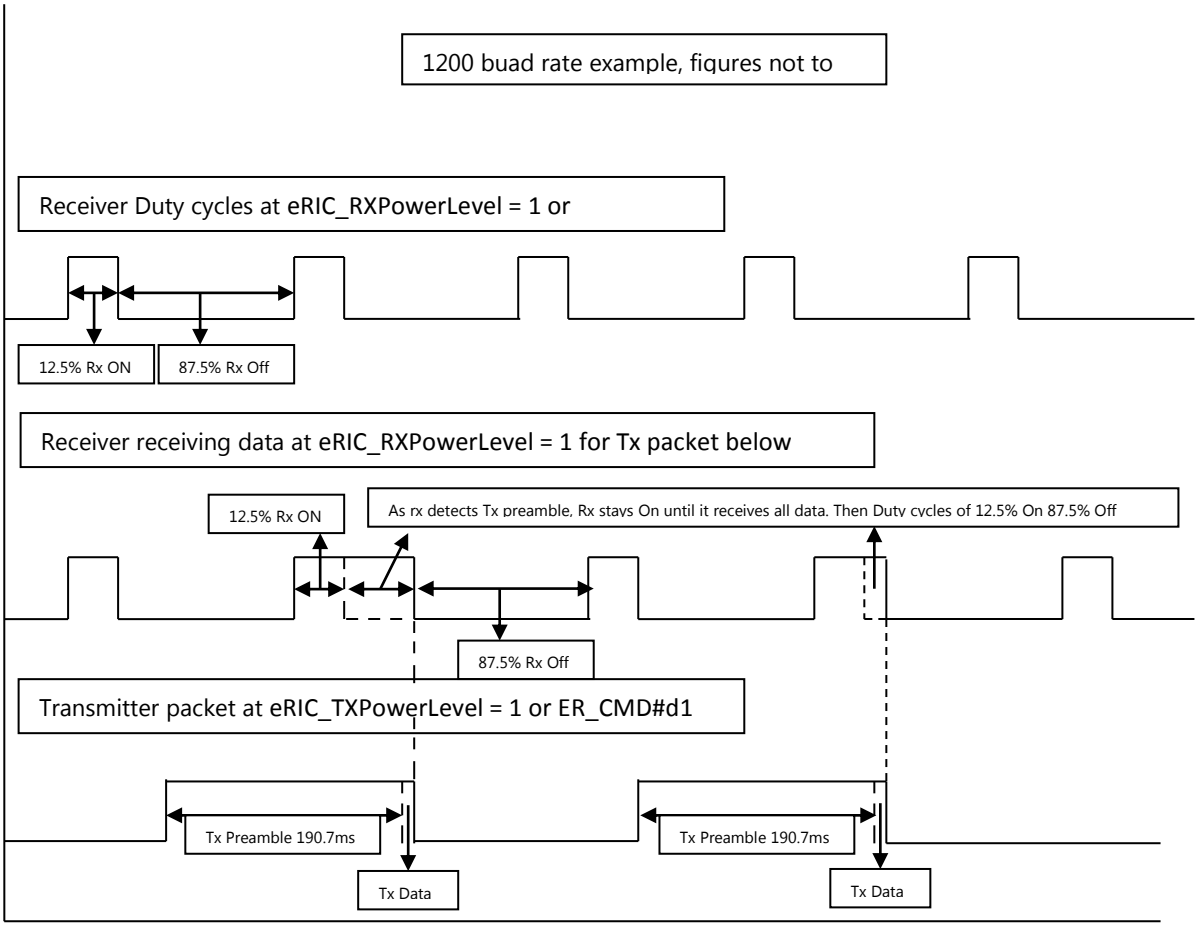
So 450uS is the fixed receiver On time at 500kHz baud rate for different eRIC_RXPowerLevels (1-7).

So at eRIC_RXPowerLevel = 7, the On time is 0.2% duty cycle, which would mean receiver is off for 99.8%.

So receiver off time at eRIC_RXPowerLevel = 7 is (99.8*450)/0.2 = 224550 uS.

So total time between receiver coming on or Receiver On+Off time is 450+224550= 225000 uS..

The receiver comes On every 225000uS for 450uS and if it detects any transmitter preamble then receiver will stay On until it receives complete packet. So for the receiver to receive a data from transmitter, the transmitter preamble should be atleast 225000uS. Choosing eRIC_TXPowerLevel = 7 at 500kHzbaud rate, a preamble time of 225050 uS is added to the transmit packet in eROS. The transmitter preamble time tables are shown further below.



eRIC_RfSenddata() takes roughly 950 clock cycles and 1150 clock cycles when eRIC_RXPowerLevels are 1-7. These clock cycles will need to be considered on top of preamble time.

The transmitter preamble time periods are given below:

The below table explains eRIC_TXPowerLevel settings							
Transmitter Preamble time in Seconds Cpu clock speed should also be atleast 9*current Over-Airbaud rate.							
RfBaud/TxPowerSavinglevel	**eRIC_TXPowerLevel = 1**	**eRIC_TXPowerLevel = 2**	**eRIC_TXPowerLevel = 3**	**eRIC_TXPowerLevel = 4**	**eRIC_TXPowerLevel = 5**	**eRIC_TXPowerLevel = 6**	**eRIC_TXPowerLevel = 7**
1200(eRIC_RfBaudRate = 0)	0.190700	0.360567	0.699215	1.381944	2.743056	5.465278	10.637500
2400(eRIC_RfBaudRate = 1)	0.096950	0.183483	0.355997	0.703793	1.397169	2.783921	5.418750
4800(eRIC_RfBaudRate = 2)	0.050075	0.094942	0.184388	0.364717	0.724225	1.443243	2.809375
9600(eRIC_RfBaudRate = 3)	0.026638	0.050671	0.098584	0.195179	0.387754	0.772903	1.504688
19200(eRIC_RfBaudRate = 4)	0.014919	0.028535	0.055682	0.110410	0.219518	0.437734	0.852344
38400(eRIC_RfBaudRate = 5)	0.009059	0.017468	0.034231	0.068026	0.135400	0.270149	0.526172
76800(eRIC_RfBaudRate = 6)	0.006130	0.011934	0.023505	0.046833	0.093341	0.186357	0.363086
100000(eRIC_RfBaudRate	0.005450	0.010650	0.021017	0.041917	0.083583	0.166917	0.325250

= 7)							
250000(eRIC_RfBaudRate = 8)	0.004100	0.008100	0.016074	0.032151	0.064203	0.128305	0.250100
500000(eRIC_RfBaudRate = 9)	0.003650	0.007250	0.014427	0.028896	0.057742	0.115435	0.225050

The combination of CPULowPowerModes and RadioLowPowerModes can be used to achieve best results of communication at lowest power. An example project is available online on LPRS website.

Using Commands: (For eRICxeasyRadioV1.5.5 and above):

There are 5 CPULowPowerMode commands, 20RadioLowpowercommands(10 for receiver and 10 for transmitter)and 5 commands which has the combination of CPULowPowerMode and RadioLowPoweMode settings.

CPULowPowerModeCommands:

1) ER_CMD#A80: All CPU Low power modes are disabled

2) ER_CMD#A81: This is same as eRIC_LPM_Level0().In this level MCLK is turned off and SMCLK, ACLK works normally. This doesn't save much power, but most of the modules can be assessed as they use SMCLK, ACLK as source clocks.

3) ER_CMD#A82: This is same as eRIC_LPM_Level1().In this level MCLK,SMCLK is turned off and ACLK works normally. Depending on the modes used, the current draw varies but generally it takes ~6uA when all radio and other modules are off, as 32768Hz(ACLK) takes ~6uA.

4) ER_CMD#A83: This is same as eRIC_LPM_Level2().In this level all clocks are Off. So eRIC would be in sleep and will be drawing only 2uA. Radio receiver is turned off and put in to sleep in this, as it might generate PUC(reset) or consume more current since all clocks are off.

5) ER_CMD#A8?: This returns what mode the module is in. Eg 1,2,3 or 0

   ACK is required for all commands above. After the module is set in one of the above modes, Pin22 is used to either enter or exit these modes. Pin22 when pulled low, will enter in the above set modes and when it is pulled low –high, module exits the mode. It is not recommended to send any more commands when Pin22 is pulled low and module is in one of the above low power modes. All commands must be sent before module going into lowpowermodes.

RadioLowPowerModeCommands:

1) ER_CMD#D0: This is same as eRIC_RXPowerLevel = 0; Radio receiver is always On. This draws a continuous current of 16mA

2) ER_CMD#D1: This is same as eRIC_RXPowerLevel = 1; Radio receiver is 12.5% On. This draws an average current of 2mA

3) ER_CMD#D2: This is same as eRIC_RXPowerLevel = 2; Radio receiver is 6.25% On. This draws an average current of 1mA

4) ER_CMD#D3: This is same as eRIC_RXPowerLevel = 3; Radio receiver is 3.13% On. This draws an average current of 500uA

5) ER_CMD#D4: This is same as eRIC_RXPowerLevel = 4; Radio receiver is 1.56% On. This draws an average current of 250uA

6) ER_CMD#D5: This is same as eRIC_RXPowerLevel = 5; Radio receiver is 0.78% On. This draws an average current of 125uA

7) ER_CMD#D6: This is same as eRIC_RXPowerLevel = 6; Radio is 0.39% On. This draws an average current of 62.5uA

8) ER_CMD#D7: This is same as eRIC_RXPowerLevel = 7; Radio is 0.2% On. This draws an average current of 32uA

9) ER_CMD#D8: This is same as eRIC_RXPowerLevel = 8; Radio is Off.

10) ER_CMD#D?: This returns the current D setting.Eg 0,1,2,,,8.

    ACK is required to send after sending above commands except for ER_CMD#D? Eeprom write is disabled for D1- 7 settings. Say for example if the module is in ER_CMD#D2 settings and if channel is changed by command ER_CMD#C3, then channel will be changed but it will not be stored in EEPROM. So on next power up or reset the channel will not be set to 3 as it is not stored in EEPROM. So for the commands, which needs to be stored in EEPROM must be sent when D is 0 or 8.

The other 10 commands are for setting transmitter in according to receiver settings for communications.

ER_CMD#dx where x is 0-8 is same as eRIC_TXPowerLevel = x where x is 0 -8.

ER_CMD#dx is chosen in such a way that x in ER_CMD#dx is always greater than or equal to y in ER_CMD#Dy.

For example, if receiver is set with ER_CMD#D5 and if transmitter is set with ER_CMD#d4, then receiver will not receive data. For successful communications, transmitter should be set with ER_CMD#d5 or above or receiver should be set with ER_CMD#D4 or below.

The exact timings of Receiver being On/Off in different ER_CMD#Dx and at different over-air baud rates are already explained above.

ER_CMD#d? will return the current d setting. Eg 0,1,2,,8. This doesn't require sending ACK after command.

ER_CMD#d8 will set radio in to sleep mode if ER_CMD#D8 is already sent.

Combination of CPULowPowerMode and RadioLowPowerMode commands:

1) ER_CMD#A20: LowPowerModes are disabled.

2) ER_CMD#A21: In this mode eRIC_LPM_Level0() is enabled. Cpu Clock is set to 1048576Hz if the current Cpu clock is more than 108576Hz and ER_CMD#D7 (eRIC_RXPowerLevel = 7) is set if current eRIC_RXPowerLevel (Dx) is less than 7. Uart Baud rate (ER_CMD#U4)19200 is also set if current Uart baud rate is greater than 19200 (U4). Pin22 is used to enter/exit this mode. Pull down enabled on Pin22. It enters this Lowpowermode when Pin22 is pulled low and exits when it pulled Low-High. It is not recommended to change commands or any radio setting when Pin22 is low and after this command is sent.

   All other interrupts are disabled except UART, Pin22 and Radio. eRIC can send data and receive data.

   When it exits this mode, Cpu Clock, eRIC_RXPowerLevel, Uart is changed back to whatever it was set before.

   Current consumption is roughly ~104-180 uA

3) ER_CMD#A22: In this mode eRIC_LPM_Level1() is enabled. Cpu Clock is set to 32768Hz if current Cpu clock is more than 32768 and ER_CMD#D7 (eRIC_RXPowerLevel = 7) is set if current eRIC_RXPowerLevel is less than 7. Uart Baud rate (ER_CMD#U0)1200 is also set. Pin22 is used to enter/exit this mode. Pull down enabled on Pin22. It enters Lowpowermode when Pin22 is pulled low and exits when it pulled Low-High. It is not recommended to change commands or any radio setting when Pin22 is low and after this command is sent.

   All other interrupts are disabled except UART, Pin22 and Radio. eRIC can send data and receive data.

   When it exits this mode, Cpu Clock, eRIC_RXPowerLevel, Uart is changed back to whatever it was set before.

   Current consumption is around ~37uA. If ER_CMD#D8 (Radio receiver Off) (Can send data but not receive data) then Current consumption would be around ~6uA

4) ER_CMD#A23: This is same as ER_CMD#A83.In this mode eRIC_LPM_Level2() is enabled. All clocks are turned off. Radio is off. Pin22 is used to enter/exit this mode. Pull down enabled on Pin22. It enters Lowpowermode when Pin22 is pulled low and exits when it pulled Low-High. It is not recommended to change commands or any radio setting when Pin22 is low and after this command is sent.

   When it exits this mode, radio resumes its functions as set before.

   Current consumption ~2uA

5) ER_CMD#A2?: Returns current mode setting. Eg 0,1,2,3

The following table gives current consumption of different commands:

Mode	Condition	Min Current	Typical current	Max current	Unit	Notes
Active mode	eRICxeasyRadioV1.5.5 and eROS V4.5	18	19.6	23	mA	
ER_CMD#A81(eRIC_LPM_Level0())			18.8		mA	
ER_CMD#A82(eRIC_LPM_Level1())			18.8		mA	
ER_CMD#A83(eRIC_LPM_Level2())			2		uA	
ER_CMD#D8	Active mode with receiver Off		1.08		mA	
ER_CMD#A21	ER_CMD#D7,ER_CMD#U4,clock is set at 1048576Hz	100	110	180	uA	
ER_CMD#A22	ER_CMD#D7,ER_CMD#U0,clock is set at 32768Hz	5.7	37	38	uA	
ER_CMD#A23	Radio is off		2		uA	
ER_CMD#U0,ER_CMD#f0007A120,ER_CMD#A82(Pin22 Held high),ER_CMD#D7	Afters sending command in the same series, then when Pin22 is puled low then module will be set at Uart 1200,clock 500kHZ and D7.	80	112	150	uA	

ER_CMD#U0,ER_CMD #f0007A120,ER_CMD# A82(Pin22 Held high),ER_CMD#B0,ER_ CMD#D7	Afters sending command in the same series, then when Pin22 is puled low then module will be set at Uart 1200,clock 500kHZ,RF buad 1200 B0 and D7.		82		uA	
ER_CMD#T0/ER_CMD #T1/ER_CMD#T2/ER_ CMD#Z01/ ER_CMD#Z02/ ER_CMD#Z03	All test modes and carrier on		32.5		mA	