POLYTECH
CLERMONT-FERRAND

RENESAS

January 2014

Study and implementation of RL78 low power mode

Application note

Emeric Bartholin
GE5 YEARS 2013-2014

Summary

Introduction

This application note describe the different operating mode of the microcontroller RL78 which made by the company Renesas. Renesas are a Japanese company leader on the microcontroller market. This company employed 25000 salaries in several country.

The microcontroller RL78 are the true lower power microcontroller of Renesas. It's a polyvalent microcontroller specialize for applications with a strong constraint of electrical consummation.

The RL78 are available in several release (G12, G13, G14…) with more or less peripheries, more or less evolved.

1. Internal architecture of RL78G14

The figure following present the internal contents of the microcontroller RL78G14.

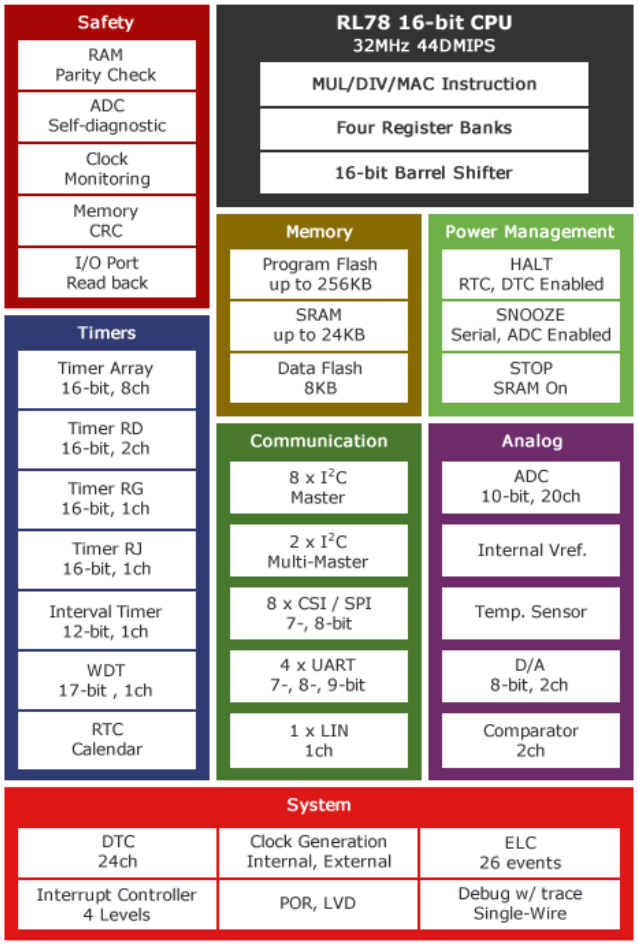


Figure 1: internal architecture of RL78G14

On this figures we can see the diversity of the available peripheries of the RL78 G14.

They are several timer for all application input capture, PWM, delay… They also have communication module and analog module with a 10 bits analog converter. This microcontroller are flash memory dedicate for storage data and 24 channel of data transfers controller for automatize memory transfer.

2. The three mode of low power

The RL78 are four different mode of operation which describe by the following figure.

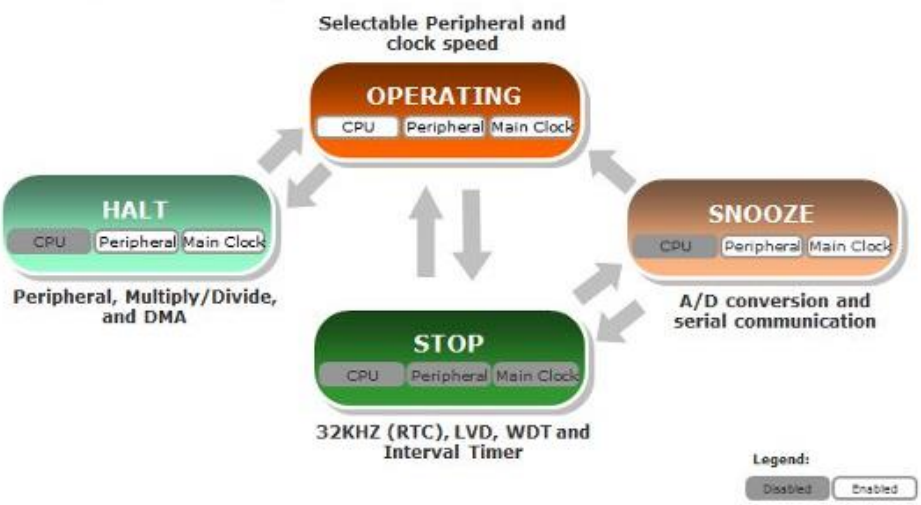


Figure 2: operating mode of RL78

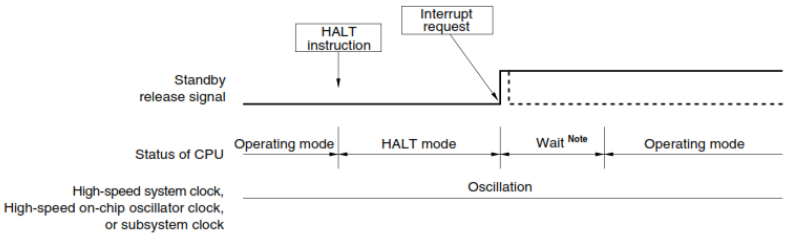
The RL78 also has 4 different operating mode:

- Mode "operating" the microcontroller is in normal operation mode is more about consuming 1.2mA.
- In "Halt" mode the heart of the microcontroller are stop but a lot of peripheries are available. Halt mode are the intermediate mode and reduces power consumption to approximately 0.5mA this value depends on the number of peripheries used by the application.
- The "Stop" mode is the ultimate mode of the point of view of power consumption. This mode stopped all the microcontroller and peripheries unless some timer and interrupt pin to wake up the microcontroller. This mode reduce power consumption of the microcontroller about 0.0005mA.
- The “SNOOZE” mode permit to maximize the time in Stop mode when the application use A/D converter or serial communication because this mode permit to wake up peripheries just during the time they need and switch automatically in STOP mode without using of CPU.

2.1 HALT mode

2.1.1 Detaille description

The first mode of standby function is “Halt”. In Halt mode the CPU operation clock is stopped but the clock of using peripheries still on. In this mode, the operating current is not decreased as much as stop mode, but halt mode is effective for restarting operation immediately after an interrupt request.



Note Wait time for HALT mode release

- When vectored interrupt servicing is carried out
  - Main system clock: 13 to 15 clock
  - Subsystem clock (RTCLPC = 0): 8 to 10 clock
  - Subsystem clock (RTCLPC = 1): 9 to 11 clock
- When vectored interrupt servicing is not carried out
  - Main system clock: 8 to 9 clock
  - Subsystem clock (RTCLPC = 0): 3 to 4 clock
  - Subsystem clock (RTCLPC = 1): 4 to 5 clock

Figure 3: Halt mode Release by interrupt request

For set HALT mode you must execute HALT instruction. For example with IAR compiler insert in your code this line “HALT ();” or the intrinsic function “_halt();”.

Halt mode is best mode if you want to restart CPU operation quickly and for use peripheries without CPU.

2.1.2 Operable peripheries during Halt mode

As mention before halt mode stop operation of the CPU operation but not peripheries with certain condition. All peripheries enable before the setting of HALT mode still on.

Operable peripheries during HALT mode:

- DTC (Data transfer controleur)
- ALL timer
- ADC and DAC
- Serial interface
- RAM and RAM control function are operable with a DTC or stop
- See annex for more information

If the CPU operate on the subsystem clock and if low consumption RTC mode is active the following peripheries are disable:

- ADC, DAC
- serial interface,
- DTC
- Timer array unit, Timer RJ, RG, RD

2.2 STOP mode

2.2.1 Detaille description

To minimize the electric consummation you must use STOP mode. In this mode the high-speed system clock oscillator and high-speed on-chip oscillator stop, stopping the whole system, thereby considerably reducing the CPU operating current. This mode can be clear by an interrupt request but restarting the main clock needs a waiting time for clock stabilization.

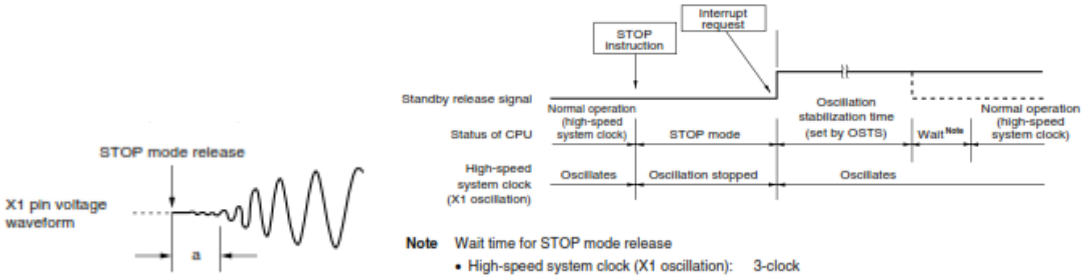


Figure 4: Clock stabilization and STOP mode Release by interrupt request

To set STOP mode you must have to execute STOP instruction. For example with IAR compiler insert in your code this line “STOP ();” or the intrinsic function “_stop();”.Caution the stop instruction is not execute during data flash programming.

To resume the restart time for STOP mode are longer than HALT mode but STOP mode operating current are less than HALT mode.

2.2.2 Operable peripheries during STOP mode

In STOP mode the main clock of the system are stop this implies some peripheries are stop too. Some peripheries still operable during STOP mode but you must active them before entering stop mode:

- RTC, interval timer
- Watchdog timer
- Power-on reset
- External interrupt and Key interrupt

2.3 SNOOZE mode

2.3.1 Detaille description

SNOOZE mode is use when you wait a data reception or A/D conversion request by and interrupt signal the end of the reception or the conversion is the source of a DTC. So STOP mode are clear by the first interrupt the conversion or reception are execute and the result store by the DTC and after this STOP mode are set automatically without CPU operating.

SNOOZE mode can be used only if your system use the high-speed on-chip oscillator for main clock.

To set SNOOZE mode you must have to set up serial standby control register (SSCm) before switching to STOP mode for serial unit. For A/D converter you have to set up A/D converter mode register (ADM2) before switching to STOP mode.

Annex

Table 23-1. Operating Statuses in HALT Mode (1/2)

HALT Mode Setting / Item			When HALT Instruction Is Executed While CPU Is Operating on Main System Clock		
			When CPU Is Operating on High-speed On-chip Oscillator Clock (f_{IH})	When CPU Is Operating on X1 Clock (f_X)	When CPU Is Operating on External Main System Clock (f_{EX})
System clock			Clock supply to the CPU is stopped		
	Main system clock	f_{IH}	Operation continues (cannot be stopped)	Operation disabled	
		f_X	Operation disabled	Operation continues (cannot be stopped)	Cannot operate
		f_{EX}		Cannot operate	Operation continues (cannot be stopped)
	Subsystem clock	f_{XT}	Status before HALT mode was set is retained		
		f_{EXS}			
	f_{IL}		Set by bits 0 (WDSTBYON) and 4 (WDTON) of option byte (000C0H), and WUTMMCK0 bit of operation speed mode control register (OSMC) • WUTMMCK0 = 1: Oscillates • WUTMMCK0 = 0 and WDTON = 0: Stops • WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 1: Oscillates • WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 0: Stops		
CPU			Operation stopped		
Code flash memory					
Data flash memory					
RAM			Operation stopped (Operable while in the DTC is executed)		
Port (latch)			Status before HALT mode was set is retained		
Timer array unit			Operable		
Real-time clock (RTC)					
12-bit interval timer					
Watchdog timer			See CHAPTER 13 WATCHDOG TIMER		
Timer RJ			Operable		
Timer RD					
Timer RG					
Clock output/buzzer output					
A/D converter					
D/A converter Note					
Comparator Note					
Serial array unit (SAU)					
Serial interface (IICA)					
DTC			Operable		
ELC			Operable function blocks can be linked		
Power-on-reset function			Operable		
Voltage detection function					
External interrupt					
Key interrupt function					
CRC operation function	High-speed CRC				
	General-purpose CRC		Operation stopped (Operable while in the DTC is executed)		
Illegal-memory access detection function			Operation stopped (Operable while in the DTC is executed)		
RAM parity check function					
RAM guard function					
SFR guard function					

Note Only for products with 96 KB or more code flash memory.

Remark Operation stopped: Operation is automatically stopped before switching to the HALT mode.
Operation disabled: Operation is stopped before switching to the HALT mode.
f_{IH}: High-speed on-chip oscillator clock
f_X: X1 clock
f_{XT}: XT1 clock
f_{IL}: Low-speed on-chip oscillator clock
f_{EX}: External main system clock
f_{EXS}: External subsystem clock

Table 23-1. Operating Statuses in HALT Mode (2/2)

HALT Mode Setting / Item			When HALT Instruction Is Executed While CPU Is Operating on Subsystem Clock	
			When CPU Is Operating on XT1 Clock (f_{XT})	When CPU Is Operating on External Subsystem Clock (f_{EXS})
System clock			Clock supply to the CPU is stopped	
	Main system clock	f_{IH}	Operation disabled	
		f_X		
		f_{EX}		
	Subsystem clock	f_{XT}	Operation continues (cannot be stopped)	Cannot operate
		f_{EXS}	Cannot operate	Operation continues (cannot be stopped)
	f_{IL}		Set by bits 0 (WDSTBYON) and 4 (WDTON) of option byte (000C0H), and WUTMMCK0 bit of operation speed mode control register (OSMC) • WUTMMCK0 = 1: Oscillates • WUTMMCK0 = 0 and WDTON = 0: Stops • WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 1: Oscillates • WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 0: Stops	
CPU			Operation stopped	
Code flash memory				
Data flash memory				
RAM			Operation stopped (Operable while in the DTC is executed)	
Port (latch)			Status before HALT mode was set is retained	
Timer array unit			Operable (Operation is disabled while in the low consumption RTC mode (when the RTCLPC bit of the OSMC register is 1))	
Real-time clock (RTC)			Operable	
12-bit interval timer				
Watchdog timer			See CHAPTER 13 WATCHDOG TIMER	
Timer RJ			Operable (Operation is disabled while in the low consumption RTC mode (when the RTCLPC bit of the OSMC register is 1))	
Timer RD				
Timer RG				
Clock output/buzzer output				
A/D converter			Operation disabled	
D/A converter Note			Operable (Operation is disabled while in the low consumption RTC mode (when the RTCLPC bit of the OSMC register is 1))	
Comparator Note				
Serial array unit (SAU)				
Serial interface (IICA)			Operation disabled	
DTC			Operable (Operation is disabled while in the low consumption RTC mode (when the RTCLPC bit of the OSMC register is 1))	
ELC			Operable function blocks can be linked	
Power-on-reset function			Operable	
Voltage detection function				
External interrupt				
Key interrupt function				
CRC operation function	High-speed CRC		Operation disabled	
	General-purpose CRC		Operation stopped (Operable while in the DTC is executed)	
Illegal-memory access detection function			Operation stopped (Operable while in the DTC is executed)	
RAM parity check function				
RAM guard function				
SFR guard function				

Note Only for products with 96 KB or more code flash memory.

Remark Operation stopped: Operation is automatically stopped before switching to the HALT mode.
Operation disabled: Operation is stopped before switching to the HALT mode.
f_{IH}: High-speed on-chip oscillator clock
f_X: X1 clock
f_{XT}: XT1 clock
f_{IL}: Low-speed on-chip oscillator clock
f_{EX}: External main system clock
f_{EXS}: External subsystem clock

Table 23-2. Operating Statuses in STOP Mode

STOP Mode Setting / Item			When STOP Instruction Is Executed While CPU Is Operating on Main System Clock		
			When CPU Is Operating on High-speed On-chip Oscillator clock (f_{IH})	When CPU Is Operating on X1 Clock (f_X)	When CPU Is Operating on External Main System Clock (f_{EX})
System clock			Clock supply to the CPU is stopped		
Main system clock	f_{IH}		Stopped		
	f_X				
	f_{EX}				
Subsystem clock	f_{XT}		Status before STOP mode was set is retained		
	f_{EXS}				
f_{IL}			Set by bits 0 (WDSTBYON) and 4 (WDTON) of option byte (000C0H), and WUTMMCK0 bit of operation speed mode control register (OSMC) • WUTMMCK0 = 1: Oscillates • WUTMMCK0 = 0 and WDTON = 0: Stops • WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 1: Oscillates • WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 0: Stops		
CPU			Operation stopped		
Code flash memory					
Data flash memory			Operation stopped (The STOP instruction is not executed during data flash programming)		
RAM			Operation stopped		
Port (latch)			Status before STOP mode was set is retained		
Timer array unit			Operation disabled		
Real-time clock (RTC)			Operable		
12-bit interval timer					
Watchdog timer			See CHAPTER 13 WATCHDOG TIMER		
Timer RJ			Wakeup by event counter mode operable		
Timer RD			Operation disabled		
Timer RG					
Clock output/buzzer output			Operable only when subsystem clock is selected as the count clock		
A/D converter			Wakeup operation is enabled (switching to the SNOOZE mode)		
D/A converter Note			Operable (status before STOP mode was set is retained)		
Comparator Note			Operable (when digital filter is not used)		
Serial array unit (SAU)			Wakeup operation is enabled only for CSIp and UARTq (switching to the SNOOZE mode) Operation is disabled for anything other than CSIp and UARTq		
Serial interface (IICA)			Wakeup by address match operable		
DTC			Operation disabled		
ELC			Operable function blocks can be linked		
Power-on-reset function			Operable		
Voltage detection function					
External interrupt					
Key interrupt function					
CRC operation function	High-speed CRC		Operation stopped		
	General-purpose CRC				
Illegal-memory access detection function					
RAM parity check function					
RAM guard function					
SFR guard function					

Note Only for products with 96 KB or more code flash memory.

Table 23-3. Operating Statuses in SNOOZE Mode

Item \ STOP Mode Setting			During STOP mode, receiving data signal from CSIp and UARTq, inputting timer trigger signal to A/D converter, and generating DTC activation by interrupt
			When CPU Is Operating on High-speed On-chip Oscillator Clock (f_{IH})
System clock			Clock supply to the CPU is stopped
	Main system clock	f_{IH}	Operation started
		f_X	Stopped
		f_{EX}	
	Subsystem clock	f_{XT}	Use of the status while in the STOP mode continues
		f_{EXS}	
	f_{IL}		Set by bits 0 (WDSTBYON) and 4 (WDTON) of option byte (000C0H), and WUTMMCK0 bit of operation speed mode control register (OSMC) • WUTMMCK0 = 1: Oscillates • WUTMMCK0 = 0 and WDTON = 0: Stops • WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 1: Oscillates • WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 0: Stops
CPU			Operation stopped
Code flash memory			
Data flash memory			
RAM			Operation stopped (Operable while in the DTC is executed)
Port (latch)			Use of the status while in the STOP mode continues
Timer array unit			Operation disabled
Real-time clock (RTC)			Operable
12-bit interval timer			
Watchdog timer			See CHAPTER 13 WATCHDOG TIMER
Timer RJ			Operation disabled
Timer RD			
Timer RG			
Clock output/buzzer output			Operable only when subsystem clock is selected as the count clock
A/D converter			Operable
D/A converter Note			Operable (Status before SNOOZE mode was set is retained)
Comparator Note			Operable (when digital filter is not used)
Serial array unit (SAU)			Operable only CSIp and UARTq only. Operation disabled other than CSIp and UARTq.
Serial interface (IICA)			Operation disabled
DTC			Operable
ELC			Operable function blocks can be linked
Power-on-reset function			Operable
Voltage detection function			
External interrupt			
Key interrupt function			
CRC operation function	High-speed CRC		Operation stopped
	General-purpose CRC		
Illegal-memory access detection function			
RAM parity check function			
RAM guard function			
SFR guard function			

Note Only for products with 96 KB or more code flash memory.