APPLICATION NOTE:
Optimizing Performance using the 2500 Series™ Processor

December 2008

2500 Series™ Processor Scan

Normal I/O

Write outputs, read inputs on local, remote, and Profibus I/O.

Main RLL

Run RLL Task 1 to completion.

Special Function I/O

Service all local and remote SF modules to max task codes / scan. Service RBC serial ports.

NOTE: all discrete scan tasks run to completion, even if it causes a scan overrun. You cannot set time slices for these tasks.

Normal I/O

Local I/O Operation

- Read/write all I/O modules in the local base
- Very fast backplane access
- Typical scan time 1ms

Normal I/O

Remote I/O Operation

- 1Mbit communications, up to 15 bases
- Overhead:
  - First remote base adds ~5ms per scan
  - Each additional remote base adds 1ms per scan
  - Note: each “enabled” add 1ms per scan even if unused
- Actual scan time for the base depends on I/O installed
- Each RBC Communication Port transaction adds ~2 ms

Normal I/O

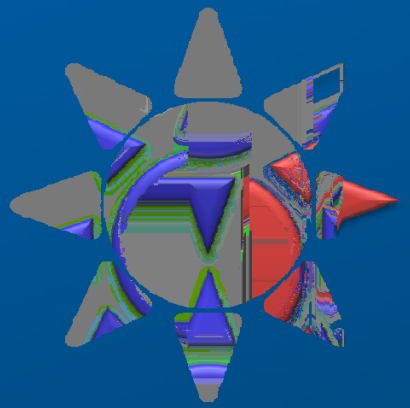
Remote I/O Optimization

- Use the minimum number of remote bases required for the application
- Disable all unused bases
- Minimize or eliminate the use of RBC serial ports

Normal I/O

Profibus I/O Operation

- Operates through a dual port RAM interface to a separate Profibus microprocessor



Dual Port RAM

PLC SCAN

PROFIBUS SCAN

Control Technology Inc.
Optimizing Process Performance Using the 2500 Series™ Processor

Normal I/O

Profibus I/O Optimization

- Important effects of separate Profibus scan
  - At the end of each Profibus cycle, the CPU receives an interrupt to get Profibus data
  - Transferring data from Dual-Port RAM to CPU memory requires 0.25-0.5 ms
  - Therefore, running a very fast Profibus cycle can degrade CPU performance.
  - For optimum performance, we recommend Profibus network operation at slowest baud rate that will achieve 2-3 cycles per PLC scan.
  - Profibus cycle speed is set in ComProfibus

Normal I/O

Profibus I/O Optimization

- Asynchronous Mode
  - PLC and Profibus scans run independently and fast as possible
- Synchronous Mode
  - Profibus is required to complete 2 scans during the normal I/O cycle
  - CPU will wait on Profibus scan before starting new RLL scan
- For best performance we recommend operation in asynchronous mode

Main RLL

Operation

- Runs once each scan from start until it encounters an END instruction.
- Most instructions execute < 20µs
- Some exceptions
  - TSET, DSET, and RSD
  - MOVE (can copy 32767 elements / scan)
  - Immediate I/O: Contacts, Coils, IORW
  - RLL PID (Fast Loop)
  - In-Line SF Programs and SF Subroutines

Main RLL

Optimization

- SKP (Skip) and LBL (Label)
  - Can be used to bypass sections of logic and improve scan time
  - Take care because Outputs are not updated and Timers do not run if “skipped”
- Use X, Y or C contacts and coils when possible. Avoid using “bit of word” contacts.
- Monitor Main RLL (Task 1) Peak Execution Time using TPET1 variable to see effects of optimizations

Special Function I/O

Operation

- SF I/O transactions can be large (up to hundreds of bytes)
- Each SF module in the system can do up to 8 transactions per scan
- A high activity SF module (NIM or Ethernet) adds up to 40 ms per scan
- Moderate activity: ~15-20 ms per scan

Special Function I/O

Optimization

- Move SF modules to Local Base when possible, especially communications modules with high activity.
- If it is not possible to put all SF modules in the Local Base, distribute SF modules evenly among Remote Bases.

2500 Series™ Processor Scan

PID Loops

Analog Alarms

Cyclic SF

Priority SF Programs

Normal SF Programs

RLL SFSUB

RLL SFSUB0

Normal Comm

Priority Comm

Network Comm

Diag-nostics

Analog Scan

Analog Tasks
Operation

- Each task is guaranteed execution once per scan
- Each task has a user-defined time slice which sets the maximum execution time for that task
- The task runs until all work is completed, or until the time slice expires
- Analog tasks are also executed during “wait time” that occurs in Normal I/O and SF I/O

2500 Series™ Processor Scan
Analog Tasks

PID Loops

- PID loops set for cyclic operation, executed in order of priority. SF programs called by loops are also executed here.

Analog Alarms

- Alarms set for cyclic operation, executed in order of priority. SF Programs called by alarms are also executed here.

2500 Series™ Processor Scan
Analog Tasks

Cyclic SF

- SF Programs set for cyclic operation, executed in order of priority

Priority SF Programs

- Priority SF Programs queued from RLL SFPGM box

Normal SF Programs

- Normal SF programs queued from RLL SFPGM box

2500 Series™ Processor Scan
Analog Tasks

RLL SFSUB

- SFSUBs queued from RLL SFSUB box

RLL SFSUB0

- SFSUBs queued from RLL SFSUB0 box

2500 Series™ Processor Scan
Analog Tasks

Normal Comm

- Commands from serial and USB ports which execute over several scans (ie SEARCH commands)

Priority Comm

- Commands from serial and USB ports which READ or WRITE data (0.5ms per command)

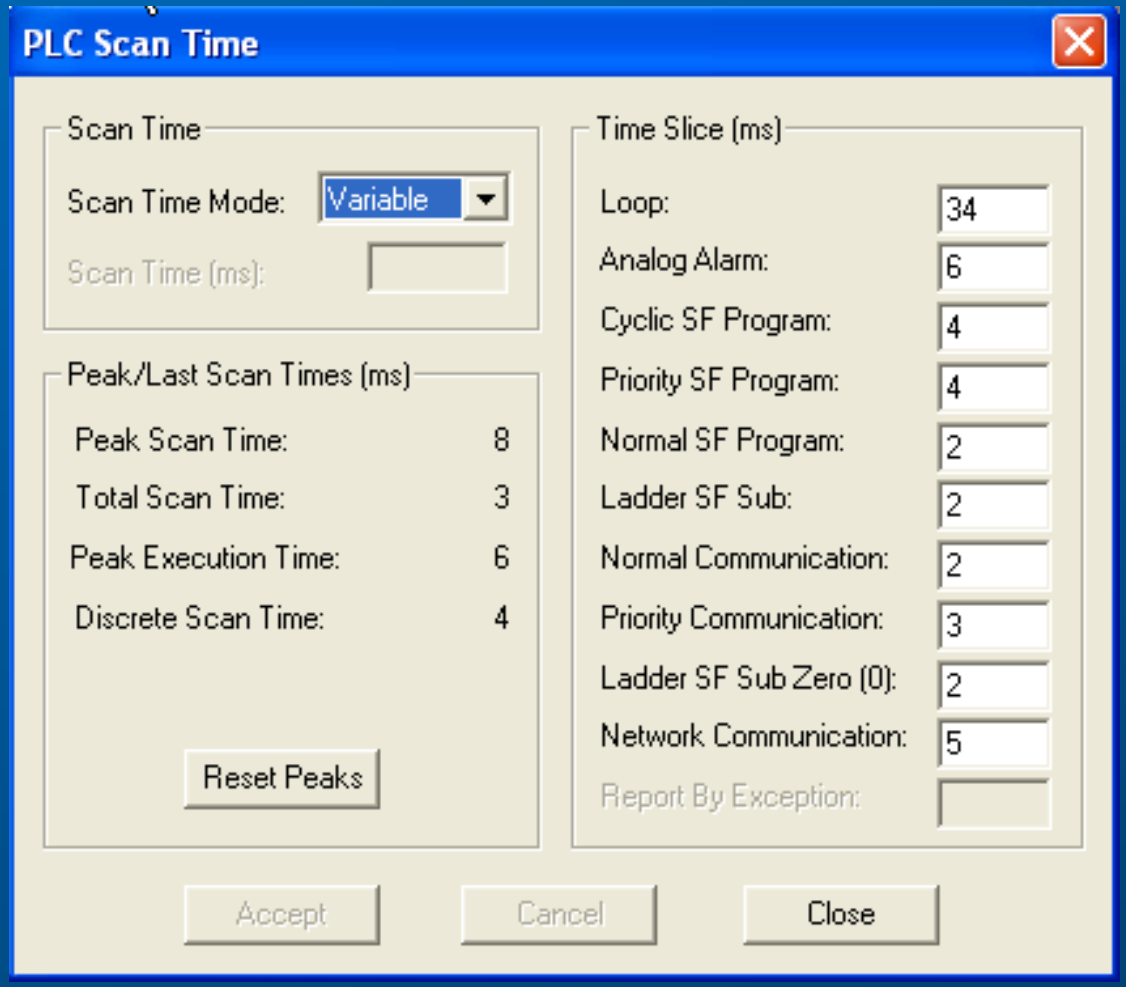
Network Comm

- Commands from CPU ethernet port (1-2ms per command)

Diag-nostics

- Memory test, program checksum, front panel, other hardware checks

Analog Tasks
Setting Time Slice



Analog Tasks
Time Slice Optimization

- Programming Reference Guide Ch7
- Basic strategy
  - Reduce analog time slices as much as possible
    - Be sure loops and alarms don't overrun
    - Be sure SF program execution is satisfactory
- Use STW162 to detect overrunning loops, alarms, SF programs

Analog Tasks
Time Slice Optimization

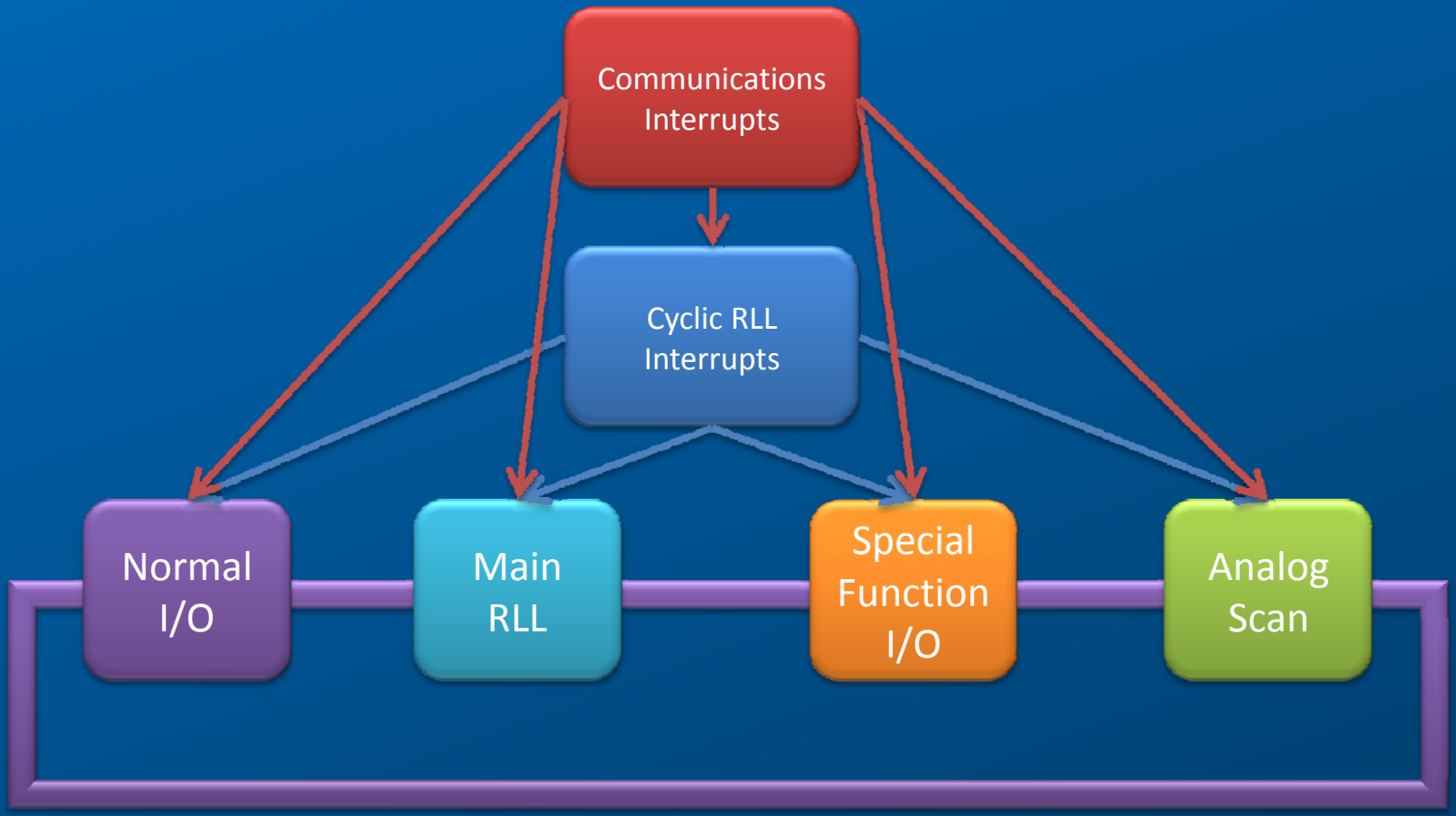
- Use “PET” variables to view individual peak execution times
  - TPET1, TPET2: RLL execution times
  - APETn, LPETn: Loop / Alarm time from scheduling until execution completes
  - PPETn: SF Programs (queued from RLL) from scheduling until execution completes
  - SPETn: SF Subroutine (queued from RLL) from scheduling until execution completes

PLC Scan Mode Selection

- Variable
  - Fastest PLC Scan
  - Executes each Analog Task once / scan
- Variable with Limit
  - Repeats Analog Task processing (if needed) until Scan Time Limit is reached
- Fixed
  - Use only if fixed I/O update required

2500 Series™ Processor

High Priority Interrupts

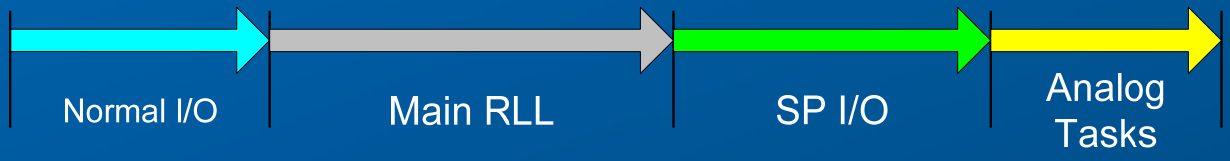


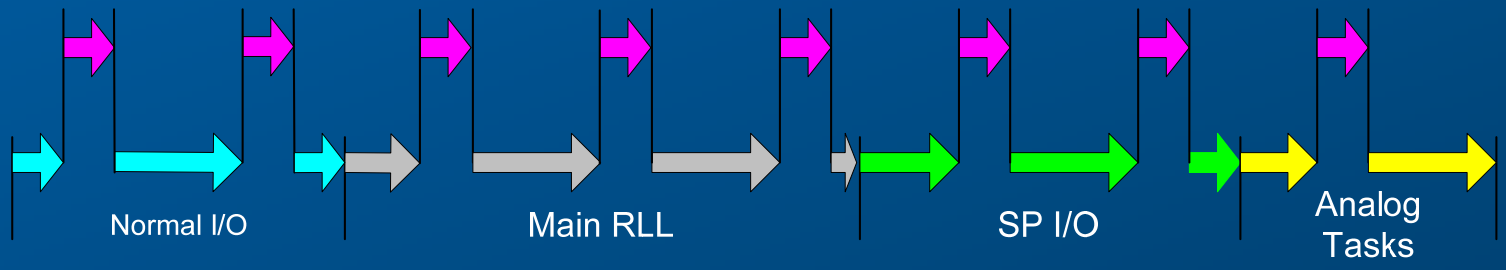
2500 Series™ Processor
High Priority Interrupts

- Cyclic RLL
  - Separate RLL program (TASK2) that runs periodically based on specified cycle time
  - Extends PLC Scan by the total time used to execute TASK2 logic during each scan.
  - Ideal for performing fixed time updates to critical I/O using Immediate I/O instructions.
  - It is also possible to “starve” processing time for other tasks if run too frequently.

Cyclic RLL

Effect on PLC Scan:



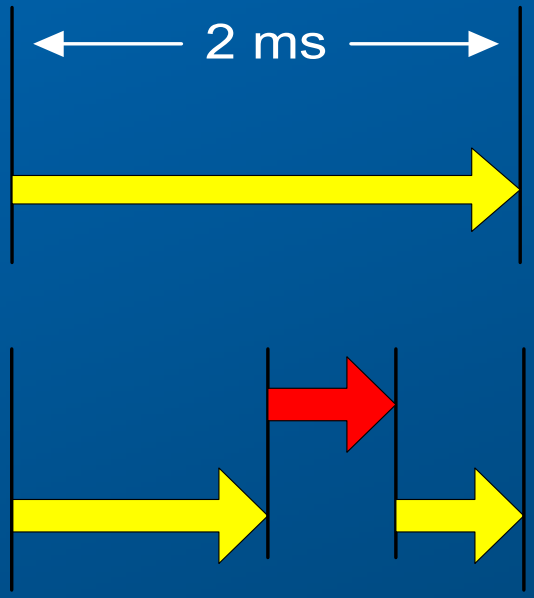


2500 Series™ Processor
High Priority Interrupts

- Communications Interrupts
  - Serial port (RS-232/USB) interrupts use minimal processing time (10-20μs)
  - TCP/IP message processing takes ~ 500μs to "run the stack" (validate and extract data).
  - Extends scan time if it occurs during Discrete Scan
  - Reduces time available for Analog Tasks if it occurs during Analog Scan

Communication Interrupts

Effect on Analog Tasks:



Other Optimization Guidelines

- Use “Variable” or “Variable with Limit” scan type.
- When fixed-interval I/O update is required, use Immediate I/O instructions in Cyclic RLL task.
- Avoid running Cyclic RLL task more frequently than required. Check execution time (TPET2).
- If using Ethernet connection:
  - Set ‘Network Comm’ time slice at 5ms minimum