ARX OS Insight

Version- 20.22.01
October-2023

ARX
Advanced Realtime Xecutables Management

Technology- Safe, Secure & Reliable

ARX is an advanced, feature-rich real-time operating system (RTOS) designed to address the scheduling needs of time-critical applications, such as those found in embedded systems, real-time firmware, robotics, and other real-time applications. ARX provides a real-time OS platform with firmware development acceleration capabilities, unleashing substantial productivity gains and time-to-market advantages for product engineering firms.

Use Intuitively- Customization, Configuration, Firmware/Application development, build in simple steps.

Services- System Control, IPC, Synchronization, Scalable HAL, Device drivers' development.

Portability- Highly portable, allowing application /Firmware migration from one hardware platform to other.

Debugging & Profiling- In-built debugging and profiling infrastructure allows developers to analyze system behavior, resources consumptions, and performance profiling.

Secure HW Access- System hardware peripheral resources are accesses through secure HAL.

Support- Active support, detailed self-explanatory documentation with software examples.

Deterministic Behavior- Predictable & Repeatable response time to events which help accurately to predict and control the timing of system behavior.

Low-latency- Real time services with constant and low latency ensure events are handled within designated time window.

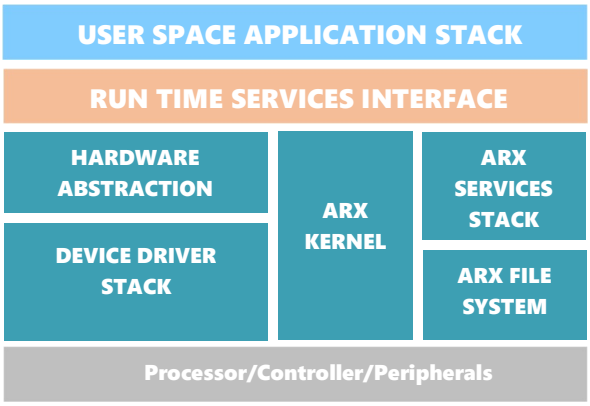
Safety & Reliability- Ensuring system safety and reliability through spatial and temporal isolation of user space applications against privilege firmware mode.

Safe Scheduling- Efficient scheduling for multiple classes of tasks/threads to ensures safe application integrity during execution by inbuilt priority inversion algorithm.

Scalability & Lean Footprint- Highly scalable kernel footprint which allow low-power devices to large-scale distributed systems,

Architecture

ARX architecture allow roubst interfaces among ARX componets which result in better performance, low latency communication, mainatanace, better error handling.

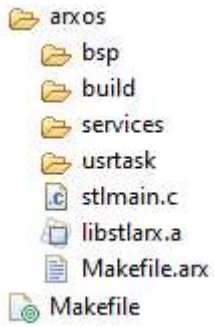


Released Modules

ARX components are organized in a simple directory structure.

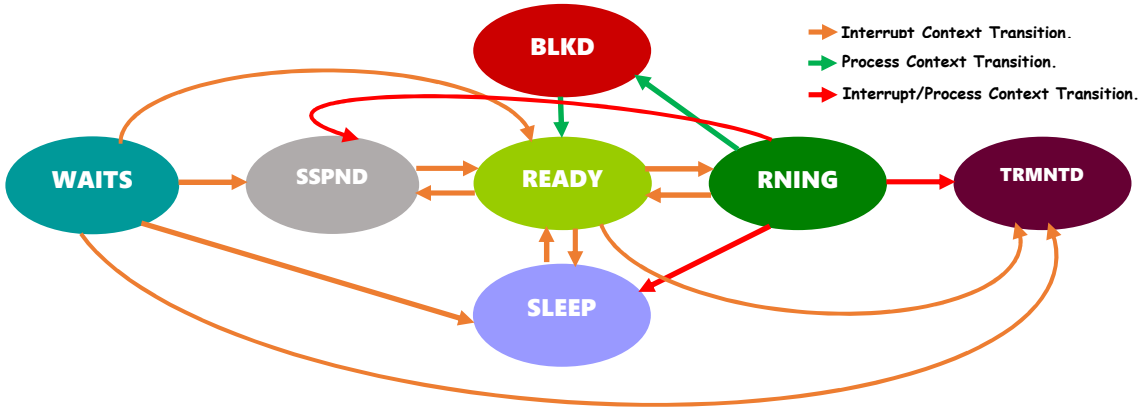
Please refer the following link to download ARX modules and necessary documents.

https://www.arxos.in/download



bsp	Contains board-specific peripherals initialization and drivers' software.
build	ARX provides the necessary (*.h) files to link the ARX kernel and user application services.
services	ARX provides services' interface (*.h) files.
usrtask	Organizes software for user applications(tasks).
stlmain.c	**main** function software example that can be called directly from the boot software to link the ARX kernel with the system boot software.
libstlarx.a	STL ARX static linkable library.
Makefile.arx	ARX components specific Makefile.
Makefile	Project master Makefile.

ARX State Transition



ARX supports 7 distinct states for user space processes, enabling efficient power management, control over task functions, and CPU utilization. A process can reside in one of the following states for a specific purpose and duration:

READY The process is eligible for CPU allocation by the scheduler, subject to CPU availability and its priority.

RUNNING Process is currently hosted by the CPU.

SLEEP A process can sleep for an indefinite period until a wakeup signal is trigger.

SUSPEND A state of limited/unlimited duration of suspension.

The process resumes either when a resume signal is triggered or its experience a time out.

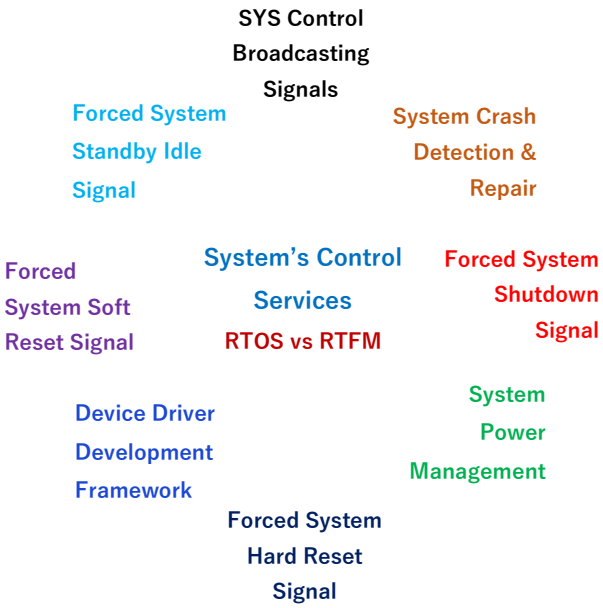
BLOCKED The process is in a state of being blocked over a resource.

WAIT This state is for event-driven or periodic class processes.

Event-driven processes expect an event, while periodic class processes wait until their period expires.

TERMINATED A state when a process is terminated, the resources of terminated process reclaimed by ARX to create new process.

Execution Control & BSP Development



ARX facilitates system-level control through various global services and signal broadcasts.

ARX operates in two different modes: User space process and privilege firmware mode.

It offers BSIR (Boot, Shutdown, Idle, and Reset) control through system broadcast signals.

Efficient power management is achieved without requiring software programming or configuration.

ARX provides a safe & secure framework for developing device drivers, BSP (Board Support Package), and SDK (Software Development Kit).

ARX Class

ARX simplifies the creation of new processes in alignment with the application's nature and functional requirements, without the need for additional configuration or software programming.

ARX classifies processes into four distinct classes, each with its own priority. These classes can accommodate multiple processes of the same or different priorities.

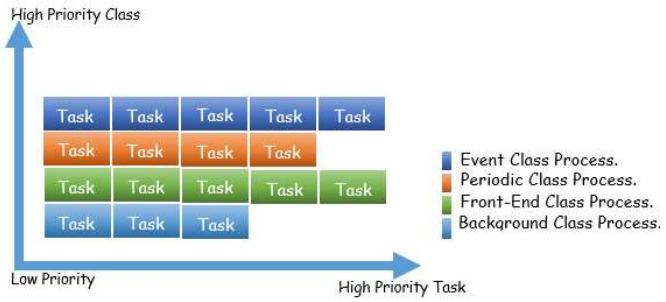
When selecting the most eligible process to assign the CPU, ARX applies 2-dimensional preemption rules, ensuring efficient process management.

Event-Driven Process Class

(Highest Priority Class)

In this class, a process becomes runnable only when an event occurs, otherwise, it waits for an event.

Each event creates a single execution session, as future executions require new events.



Periodic Process Class

(2ND Highest Priority Class)

Processes in this class are associated with their own periods and become executable when their specific periods expire.

The scheduler strives to meet recurring deadlines with an accuracy rate of up to 99.99%.

Front-End Process Class

(3RD Highest Priority Class)

Processes in this class are suitable for regular workload processing.

They are selected for CPU allocation by the scheduler in a round-robin algorithm, based on applicable priorities and preemption rules.

Background Process Class

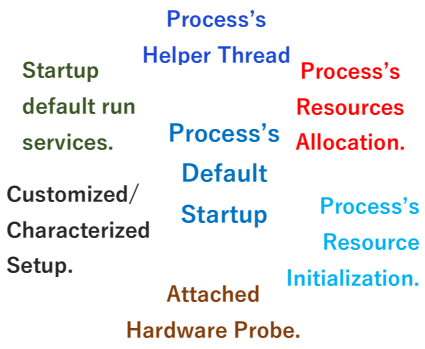
(Lowest Priority Class)

This class is designed for background processing of system loads.

Background processes are selected by the scheduler when no other class processes are available to run or when the system is least busy.

To run, processes in this class must be allocated time slices or may borrow bandwidth from the system's front-end processes those are not ready at that moment.

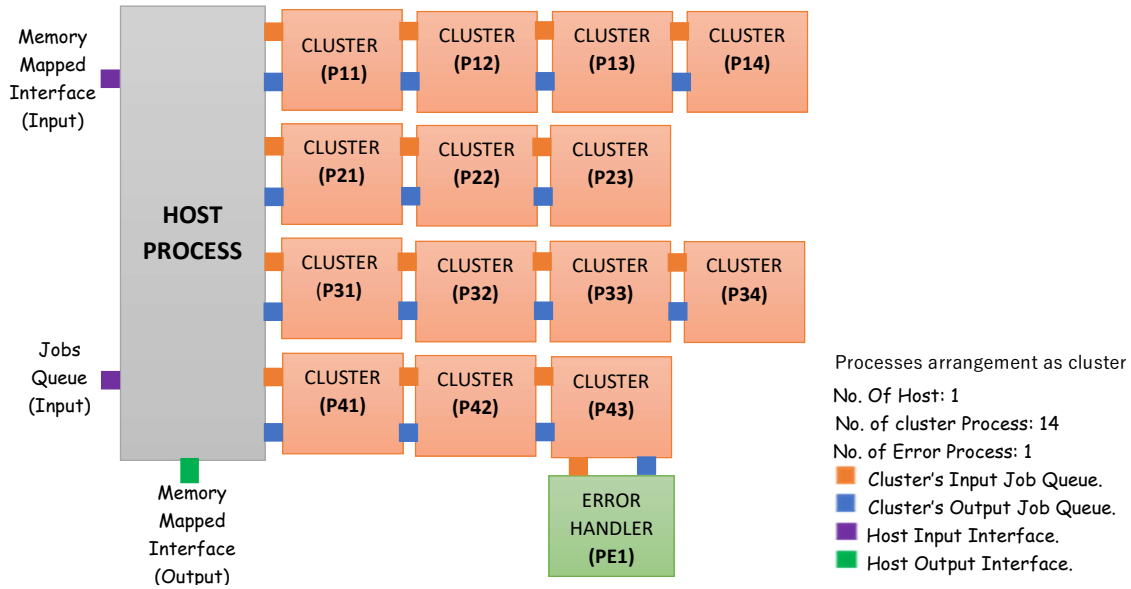
Process's Default Startup



ARX process management provides the capability to manage the default process startup. This includes setting up its helper threads, allocating resources, initializing resources, configuring process parameters with characterized or customized values, and executing default services before initiating process scheduling. The startup service is available if the respective APIs were registered.

ARX Clustering

ARX process management facilitates one or more processes to form an unique entity called cluster. Cluster address firmware development challenge when complex workloads are difficult to manage within a single process, and processing needs to be distributed across multiple processes. Within a cluster, each process is assigned only an fraction of workload. Cluster processes operate on commands from host process. The host process has four categories of commands, each of which can in-turn have various types of sub-commands. Failures during command execution can be handled by a fault handler (if attached and capable) or by a dedicated error handler task that is part of the cluster. Attached I/O job queues facilitate command transfer between processes within a cluster. An additional safe and secure memory-mapped input interface source is available to cluster through host process. The system can accommodate multiple clusters, and the host process can choose to include any cluster based on configuration.



Scalable Command Execution Infrastructure

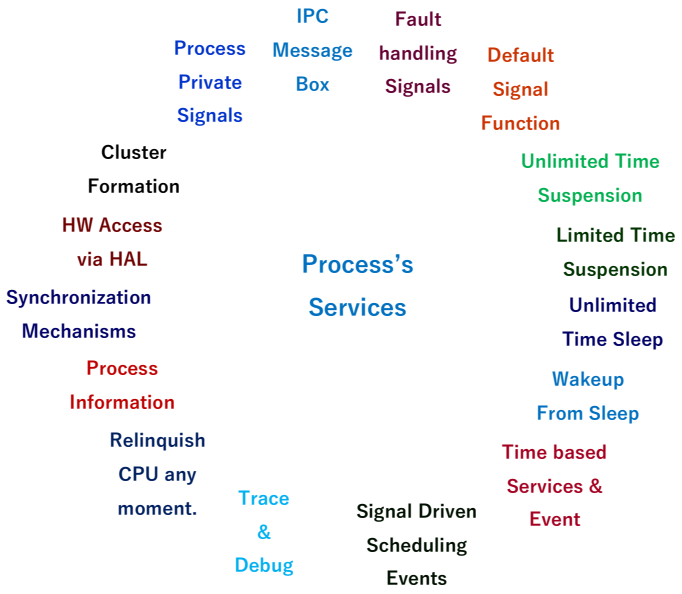
ARX features a built-in command processing infrastructure capable of executing various commands while handling and reporting failure cases for each command attempted. The typical command phases include decoding, building, and execution. Administrators ensure that each stage is completed according to the applied rules.

```
CMD Decoding Failed:34152
CMD Corrected(ERR Task):4082
CMD Corrected(Fault Handler):4004
ERR(RD Build):1410  Corrected(RD CMD):  975  ERR In Exec(RD CMD):  5233  RD CMD Executed: 7849010  RD CMD RCVD: 7864200
ERR(WR Build):1387  Corrected(WR CMD): 1015  ERR In Exec(WR CMD):  5127  WR CMD Executed: 7849137  WR CMD RCVD: 7864200
ERR(ER Build):1408  Corrected(ER CMD): 1004  ERR In Exec(ER CMD):  5206  ER CMD Executed: 7849053  ER CMD RCVD: 7864200
ERR(AD Build):1388  Corrected(AD CMD): 1010  ERR In Exec(AD CMD):  5106  AD CMD Executed: 7849183  AD CMD RCVD: 7864200
All RWEA CMDs are processed
```

At a clock speed of 166MHz, a single-core microcontroller processed over 24 commands within a 1-millisecond timeframe.

Scalable Firmware Designer & Process Control

ARX offers a self-managed interface for registering various services those are invoked when respective signal is asserted. It features APIs for designing safe and secure firmware and applications.



ARX processes are scheduling entities that enable developers to select the nature of tasks based on the system specific requirements. ARX offers services to enhance process functionality.

Private Signals- These are asserted by the system in interrupt context to individual tasks. Each process can register up to eight different signals. User can define the purpose of individual signal.

Fault Signals- These signals are asserted if to recover from a faults and errors that could occur during task functions. Faults are generated in the process context but can be handled in either the process context or interrupt context.
Each process can register up to eight different signals.

Default Signals- This interface enhances the functionality of processes. A process can optionally register up to 16 default service APIs. A service is invoked when a signal is asserted to a specific process or broadcast to a group of processes or the entire system.

State Transition- Processes transit from one state to another in response to scheduling needs and system requirements.

Information Exchange- Processes exchange information with other processes/systems via messages.

Debug & Trace- The inbuilt infrastructure provides real-time system insight information.

In-built safe system degradation signal mechanism

ARX offers a self-managed interface to control system activities, including force shutdown, forced idle, forced soft/hard-reset, crash detection and repair, as well as global or group-level message broadcasting.

Safe & Secure Memory Protection

ARX implements memory protection mechanisms to prevent tasks from accessing each other's memory space, ensuring system reliability, and preventing interference.

Advance Technology – Industry Leading Features Challenge

Comparative Analysis - ARX VS other Leading RTOS

SN	Features	Leading RTOS	ARX
1	Class - based Tasks & Scheduling.	🟠	🟢
2	Front-end process class for regular workload.	🔴	🟢
3	Event driven process class for urgent jobs.	🟢	🟢
4	Fixed period process class for regular periodic jobs.	🟠	🟢
5	Background process class for truly balancing housekeeping load.	🔴	🟢
6	Background process bandwidth management.	🔴	🟢
7	Process's assigned class Priority.	🟠	🟢
8	Process's own Priority within class itself.	🟠	🟢
9	Preemption among different scheduling Classes.	🔴	🟢
10	Preemption among processes of a class.	🟠	🟢
11	Event based Preemption.	🟢	🟢
12	Timeout based preemption.	🟠	🟢
13	Process functionality in cluster form.	🔴	🟢
14	Cluster's process input data management.	🔴	🟢
15	Cluster's process output data management.	🔴	🟢
16	Process's command/Jobs based execution model.	🔴	🟢
17	Process's responsiveness to various faults arises in each phase of CMD handling.	🔴	🟢
18	Three phase default CMD handling- Decoding, Building and Execution.	🔴	🟢
19	Process' own private signals handling.	🟠	🟢
20	Process's default functionality enhancement interface.	🔴	🟢
21	Process's Default & Private signal raised interface (Interrupt context only).	🔴	🟢
22	Watch Dog reports from the process context.	🟢	🟢
23	Device driver development framework.	🟠	🟢
24	HW Peripheral access through designed HAL interface.	🟢	🟢
25	User defined system calls registration.	🟢	🟢
26	Process Sleep (unlimited time sleep).	🟢	🟢
27	Process Suspension (unlimited time suspension).	🟢	🟢
28	Process Suspension (limited time suspension).	🟠	🟢
29	Unlimited time suspension cancelled through resume call.	🟢	🟢
30	Unlimited time sleep cancelled through wakeup call.	🟢	🟢
31	Process Termination (Remove from active scheduling).	🟢	🟢
32	Process Rescheduling request (Process & Interrupt context).	🟢	🟢
33	Process & Interrupt context Synchronization Mechanism.	🟢	🟢
34	Process allowed to Relinquish CPU at any time (Process Context only).	🟠	🟢

35	Inter-process communication (Point to point communication).	🟢	🟢
36	Process 's group-based message broadcasting.	🟠	🟢
37	Process's class-based message broadcasting.	🟠	🟢
38	Process's system-based message broadcasting.	🟠	🟢
39	Firmware mode (Privilege) VS User mode (Un-privilege) configuration.	🟠	🟢
40	Forced Idle signal call (Bring system down to Idle condition).	🔴	🟢
41	Forced Shutdown signal call (Bring system to shutdown condition).	🔴	🟢
42	Forced Reset signal call (Reset system- Soft OR Hard Reset).	🔴	🟢
43	Resume call to cancel Forced Idle, Shutdown and Reset condition.	🔴	🟢
44	User defined Idle task load function registration.	🟠	🟢
45	Previous power cycle self-crash detection and repair mechanism.	🔴	🟢
46	Stack overflow early warning.	🟠	🟢
47	SDK based organized components.	🟠	🟢
48	Process attributes information access.	🟠	🟢
49	Scheduling feature access & configuration.	🟠	🟢
50	Fixed duration Delay APIs for process's context.	🟢	🟢
51	ASSERT logics to stress TRUE/FALSE logics for process/interrupt context.	🟢	🟢
52	User defined IRQ registration interface.	🟢	🟢
53	Soft timer's registrations.	🟠	🟢
54	Auto enabled power management.	🟠	🟢
55	Up to 90% default power saving in Idle condition.	🟠	🟢
56	Software Examples, Documentation & Support.	🟢	🟢
57	🟢 Feature available without any software programming OR configuration. 🟠 Feature might be available OR need software programming OR configuration. 🔴 Feature might not be available to use and subject to further investigation.		

Lean Resource Footprints

ARX has one of the lowest RAM footprints.

ARX RAM/Flash Expenses	**Typical size in Bytes**
libstlarx.a (maximum available linkable size)	242.5 KB
libstlarx.a (minimum/default linkable size)	12.0 KB
ARX Data sections (Init/Non Init) (RAM)	2.5 KB
ARX Idle task expenses (RAM), **Inclusive 1.0KB Stack Size.**	1.2 KB
ARX Services	
New Task(RAM)	204B
New Task(Cluster)	264B
Messaging(IPC) Services (Per Task)	80B
FSIR Services	160B
Default Service API	100B

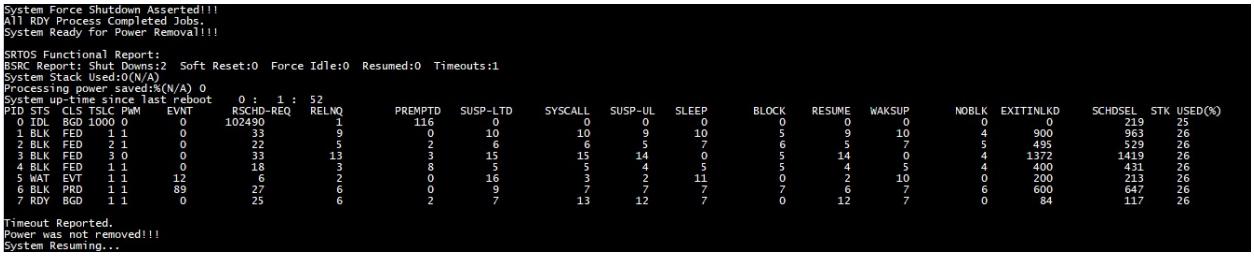
Lighting Execution Speed

ARX is the fastest among all commercial RTOS options, achieving sub-microsecond context switches across all popular platforms.

SN	ARX Services Overhead	Process Context (Typical Timing)	% Expenses	Firmware Context (Typical Timing)	% Expenses
1	Relinquish CPU (Process context only).	1.470	0.147	1.101	0.110
2	Rescheduling Request (Process/Interrupt context).	1.060	0.106	0.610	0.061
3	Suspend Task (Limited Period).	5.071	0.507	4.512	0.451
4	Suspend Task (Unlimited Period).	1.190	0.119	0.701	0.070
5	Sleep Request (Unlimited Period).	1.810	0.181	1.510	0.151
6	Wakeup Request.	1.404	0.140	1.420	0.142
7	Wakeup handling overhead at system.	3.160	0.316	3.160	0.316
8	Resume Request.	1.420	0.142	1.420	0.142
9	Resume Request overhead at system.	3.050	0.305	3.050	0.305
10	Event Request for Task.	1.740	0.174	1.730	0.173
11	Event overhead at system.	2.170	0.217	2.170	0.217
12	INBOX Message Check.	0.400	0.040	0.400	0.040
13	Message sending to another task.	1.170	0.117	1.170	0.117
14	Messages received from another task.	1.140	0.114	1.140	0.114
15	Get Mutex .	0.450	0.045	0.450	0.045
16	Release Mutex.	1.001	0.100	1.001	0.100
17	Blocked over Mutex overhead at system.	2.120	0.212	2.120	0.212
18	Context Switch.	2.760	0.276	2.760	0.276
19	Periodic Task overhead at system.	2.210	0.221	2.210	0.221
20	Timeout overhead at system.	3.840	0.384	3.840	0.384
21	System Call overhead.	1.530	0.153	0.010	0.001
22	Privilege access overhead.	0.931	0.093	0.010	0.001
23	Forced Idle request overhead.	0.587	0.058	0.587	0.058
24	Forced Shutdown request overhead.	0.587	0.058	0.587	0.058
25	Forced Reset request overhead.	0.587	0.058	0.587	0.058
26	Resume from forced condition overhead.	0.587	0.058	0.587	0.058
27	Private Signal raised request overhead.	0.545	0.054	0.545	0.054
28	Fault Signal raised request overhead.	0.479	0.047	0.479	0.047
29	Default Signal raised request overhead.	0.521	0.052	0.521	0.052
30	Broadcast default signal.	0.135	0.013	0.135	0.013

Note: All timing values are in microseconds, with a clock frequency of 166MHz and a tick value of 1 millisecond. These measurements were obtained in a standard testing environment and are subject to change in future ARX releases. ARX is committed to enhancing security and resolving potential software issues. Expenses are calculated for a single tick value that was assigned to the process.

In-built Debug View



```
System Force Shutdown Asserted!!!
All RDY Process Completed Jobs.
System Ready for Power Removal!!!

SRTOS Functional Report:
BSRC Report: Shut Downs:2  Soft Reset:0  Force Idle:0  Resumed:0  Timeouts:1
System Stack Used:0(N/A)
Processing power saved:%(N/A) 0
System up-time since last reboot     0 :   1 :  52
PID STS  CLS TSLC PWM     EVNT      RSCHD-REQ   RELNQ        PREMPTD    SUSP-LTD     SYSCALL    SUSP-UL    SLEEP      BLOCK     RESUME    WAKSUP       NOBLK  EXITINLKD     SCHDSEL   STK USED(%)
  0 IDL  BGD 1000 0        0         102490          1          116          0           0          0          0          0          0          0           0          0          219    25
  1 BLK  FED    1 1        0             33          9            0         10          10          9         10          5          9         10           4        900          963    26
  2 BLK  FED    2 1        0             22          5            2          6           6          5          7          6          5          7           5        495          529    26
  3 BLK  FED    3 0        0             33         13            3         15          15         14          0          5         14          0           4       1372         1419    26
  4 BLK  FED    1 1        0             18          3            8          5           5          4          5          5          4          5           4        400          431    26
  5 WAT  EVT    1 1       12              6          2            0         16           3          2         11          0          2         10           0        200          213    26
  6 BLK  PRD    1 1       89             27          6            0          9           7          7          7          7          6          7           6        600          647    26
  7 RDY  BGD    1 1        0             25          6            2          7          13         12          7          0         12          7           0         84          117    26

Timeout Reported.
Power was not removed!!!
System Resuming...
```

Build Tools

ARX does not require specific tool requirements. A GCC cross-compiler is sufficient for supported architectures to create executables.

Users have the freedom to choose any build system that suits their project requirements. Additionally, a sample Eclipse-based project and a makefile are provided with the release software to assist with the build system.

A makefile is suitable for building and linking various ARX components with other parts of the user's project.

MISRA And CERT Ready

The entire released software is in compliance with state of art coding and software security practices.

About STL ARX

STL ARX is a real-time operating system developed under the license of STL Pvt. Ltd. Bangalore. It is architected, designed, developed, and tested by an alumnus of the National Institute of Technology Karnataka. Among the many existing real-time operating systems, ARX offers advanced technology to address firmware development challenges on real-time platforms.

For more information, you can contact them at contact@arxos.in.

Copyright Notice and Trademark

Disclaimer

The contents of this document are believed to be correct, but we do not guarantee that they are completely free from errors or typos. The short descriptive manual for ARX explains the product's currently available features briefly. However, please note that all the features and parameters mentioned here are subject to change without notice. These changes may occur to improve functionality or performance while ensuring the compatibility of existing user applications. For detailed information please refer to the user guide.