Yale Real Living

ZigBee

Home Automation Profile

User and Integrator Reference Guide

February 2017

Rev C

Contents

Home Automation Profile

This document covers the ZigBee home automation implementation of the Yale Real Living family of deadbolt and lever lock products. The Yale ZigBee locks were built based on version 1.2 of the Home Automation Profile.

This document is designed to be used as quick reference during development for all of the commands and attributes supported by the Yale Real Living lock. Along with some information about the general behavior of the lock. This document assumes that the reader is already familiar with ZigBee interoperable mesh networking technology. Please refer to the ZigBee specifications and the Home Automation profile specification if more detail is needed.

Within every cluster of the Home Automation Profile there are multiple optional attributes and commands that may or may not be supported by a device implementing that cluster. In order to limit the length of this document only supported attributes and commands are outlined. Any attribute or command appearing in the Home Automation Profile Specification but not explicitly mentioned within this document can be assumed to be unsupported by the Yale lock.

Referenced Specifications

The Yale lock was designed to meet the following versions of ZigBee specifications. All of these documents can be obtained through the ZigBee Alliance by referencing the given document number.

Document Number	Document	Revision
3520	docs-05-3520-29-00ha-home-automation-profile-specification	29
5264	docs-09-5264-21-00zi-zigbee-ota-upgrade-cluster-specification	23
3474	docs-05-3474-21-0csg-zigbee-specification	21
5123	docs-07-5123-05-0afg-zigbee-cluster-library-specification	5

Current ZigBee Module Versions

Table 1: Yale Real Living HA1.2 Module Versions

Date Released	Version	Changes
April 2013	79	• Initial Beta Release
September 2013	86	• HA 1.2 Certified Firmware
December 2013	122	• Full System OTA
September 2016	171	• Second Generation Hardware Released (Improved RF performance)

Endpoints

All of the supported clusters of the device reside in a single ZigBee endpoint; endpoint 1.

ZigBee Security

The Yale Door lock only supports the default ZigBee network security at this time.

Yale Lock Integration Best Practices

As a battery powered end device in a ZigBee network it is important to keep communication with the Yale lock as limited as possible in order to conserve battery life. With proper binding and a robust mesh network the lock will report all state changes immediately after they happen. It should also be noted that since Yale locks are multi-processor devices some ZigBee commands

Supported Clusters

Table 2: Supported Clusters

Servers	Clients
Basic Cluster	Time Cluster
Door Lock Cluster	OTA Cluster
Power Configuration Cluster	
Alarm Cluster	
Poll Control Cluster	
Identify Cluster	
Diagnostic Cluster	

Basic Cluster

The Yale lock supports the Basic Cluster as a server per the ZigBee specification. This cluster is mainly used to retrieve general information about the lock.

Commands

Yale locks support the Reset to Factory Defaults command in the Basic Cluster. It is important to note that this command will wipe all user codes from the lock including the master code, which leaves the lock completely unsecure. The next user to activate the keypad will be prompted to pick a new Master Code.

ID	Description	Supported
0x00	Reset To Factory Defaults	Y

Attributes

Table 3: Basic Cluster Attributes

ID	Description	Type	Writable	Value	Supported
0x0001	Application Version	INT8U	F	(ZigBee Module Firmware Version)	Y
0x0003	Hardware Version	INT8U	F	(Overall Device Hardware Version)	Y
0x0004	Manufacturer Name	CHAR_STRING	F	**"Yale"**	Y
0x0005	Model Identifier	CHAR_STRING	F	See Model Identifier Table	Y
0x0007	Power Source		F		Y

The Yale lock employs a module design where the ZigBee radio communications is handled by a separate RF module that gets plugged into the back of the lock. The Application Version attribute from the Basic Cluster returns the firmware version currently running on the ZigBee chip. It is recommended to use the Current File Version attribute from the OTA cluster as this attribute contains the firmware version for all available processors in the system. See OTA cluster section for more details.

Door Lock Cluster

The Door Lock Cluster server is responsible for providing the majority of the locks functionality. The door lock cluster is a very large cluster and for convenience this section is structured to group together certain attributes and commands into groups of related functionality.

Basic Set

The basic set contains attributes and commands that are used for basic operation of the lock, including checking the current status of the lock and sending lock and unlock commands.

Table 4: Door Lock Cluster Basic Set Attributes

ID	Description	Type	Writable	Value (Decimal)	Supported
0x0000	Lock State	ENUM8	F	01 = Locked | 02 = Unlocked	Y
0x0001	Lock Type	ENUM8	F	00 = Dead Bolt **|** 07 = Tubular = Lever Lock	Y
0x0002	Actuator Enabled	BOOLEAN	F	**TRUE**	Y
0x0003	Door State*	ENUM8	F		Y*

*Supported by certain locks with optional DPS hardware installed

Table 5: Door Lock Cluster Basic Set Commands

ID	Description	Supported
0x00	Lock Door	Y
0x01	Unlock Door	Y

A note on communication timing. The lock will not lock or unlock instantly after receiving a lock or unlock command. It could take up to 2.5 seconds for the motor to complete command and fully extend or retract the bolt or thumb turn. The lock will still respond to ZigBee communication during this time but it should be considered best practice to wait until the lock completes the lock/unlock command completely before continuing communication with the lock. This can be accomplished by either implementing a short delay after a lock/unlock command (not recommended) or by utilizing binding to receive either event notifications or an attribute report once the command is completed. Please see the Binding and Attribute Reporting sections for more details.

User Settings

The User Settings group of commands and attributes contains everything required to add and manipulate users in the lock. Including setting a schedule for a user and also retrieving log information. These commands and attributes are further divided into several sub sections below.

PIN

Yale locks are capable of supporting up to 500 (see Model Identifiers table for model specific user code database sizes) users with a variable pin length of 4 to 8 digits. The coordinator should query the lock to determine which size user code database it supports. One of the user slots is reserved for the master code. The master code resides in user slot 0. The lock supports several read only attributes that can be used to query this information from the device. These are outlined in the following table:

Table 6: Door Lock Cluster PIN Attributes

ID	Description	Type	Writable	Value (Decimal)	Supported
0x0011	Num Total Users Supported	INT16U	F	**250 | 500 ****	Y
0x0012	Num Pin Users Supported	INT16U	F	**250 | 500 ****	Y
0x0017	Max Pin Length	INT8U	F	**8**	Y
0x0018	Min Pin Length	INT8U	F	**4**	Y

** Depends on which lock model is being used. See Model Identifiers table for details.

The following commands are supported by the lock and can be used to manipulate the users in the lock. Please refer to the latest version of the Home Automation v1.2 specification for details about the payload format for each command.

Table 7: Door Lock Cluster PIN Commands

ID	Description	Supported
0x05	Set PIN Code	Y
0x06	Get PIN Code	Y
0x07	Clear PIN Code	Y
0x08	Clear All PIN Codes	Y
0x09	Set User Status	Y
0x0A	Get User Status	Y
0x15	Get User Type	Y

Scheduling

The Yale door lock supports 7 Week Day Schedule per user. The Yale lock supports 1 Year Day Schedule per user. Yale locks do not support Holiday Schedules. The following table contains the attributes that can be used to query this information from the lock.

Table 8: Door Lock Cluster Scheduling Attributes

ID	Description	Type	Writable	Value (Decimal)	Supported
0x0014	Num weekday schedules supported per user	INT8U	F	**7**	Y
0x0015	Num Year day Schedules Supported per user	INT8U	F	**1**	Y

The following commands can be used to manipulate the schedule data for a single user.

Table 9: Door Lock Cluster Scheduling Commands

ID	Description	Supported
0x0B	Set Weekday Schedule	Y
0x0C	Get Weekday Schedule	Y
0x0D	Clear Weekday Schedule	Y
0x0E	Set Year Day Schedule	Y
0x0F	Set Year Day Schedule	Y
0x10	Clear Year Day Schedule	Y

There are 2 requirements that need to be satisfied before a schedule can be set for a user:

- The user slot that the controller is trying to set a schedule for needs to have a valid pin code set before the week day schedule is set.
- The network that the lock is included on needs to include a Time Server. The Time Client in the lock needs to have the correct time before scheduling will work. (Please see the Time Cluster section for more details.)

Logging

The Yale lock supports event logging and will record up to the last 255 events. The following attribute can used to query this information from the lock.

Table 10: Door Lock Cluster Logging Attribute

ID	Description	Type	Writable	Value (Decimal)	Supported
0x0010	Num Lock Records Supported	INT16U	F	**255**	Y

The following command can be used to query log records from the lock.

Table 11: Door Lock Cluster Logging Commands

ID	Description	Supported
0x04	Get Log Record	Y

The Yale lock does not support the Enable Logging attribute. If the logging feature of the lock will be used then it is important that the network contains a Time Server so that the lock can obtain the correct time. If the lock does not have the correct time then the time stamp of the log record will be incorrect.

Operational Settings

Table 11 contains all of the operational settings attributes that the Yale lock supports.

Table 12: Door Lock Cluster Operation Settings Attributes

ID	Description	Type	Writable	Value (Decimal) **bold** = default	Supported
0x0021	Language	CHAR_STRING	T	"fr"|**"en"**|"es"	Y*
0x0023	Auto Relock Time	INT32U	T	**0 = Disabled** | Max = 180	Y
0x0024	Sound Volume**	INT8U	T	0 = Silent | **01 = Low** | 02 = High	Y
0x0025	Operating Mode	ENUM8	T	**00 = Normal** | 01 = Vacation | 02 = Privacy	Y
0x0029	Enable One Touch Locking	BOOLEAN	T	**TRUE**	Y
0x002A	Enable Inside Status LED	BOOLEAN	T	**FALSE**	Y
0x002B	Enable Privacy Mode Button***	BOOLEAN	T	**FALSE**	Y

* Push Button Models do not support the language attribute

** For the push button models of the deadbolt and lever lock there are only 2 settings for the Sound Volume attribute: 0 = silent and **2 = High.**

*** This attribute is only supported by models that have a Privacy button. This is not a feature in all Yale Locks. See Model Identifier Table for more details.

There are no commands associated with the operation settings functionality.

Security Settings

The following table contains the attributes of the security settings group which the Yale lock supports.

ID	Description	Type	Writable	Value (Decimal)	Supported
0x0030	Wrong Code Entry Limit	INT8U	T	1 - 7 , **default = 5**	Y
0x0031	User Code Temporary Disable Time	INT8U	T	1 to 255, **default = 60**	Y
0x0032	Send Pin Over Air	BOOLEAN	T	default = **TRUE**	Y
0x0034	ZigBee Security Level	INT8U	F	**00 = "Network Security"**	Y

There are no commands associated with the security settings.

Alarms, Reports and Events

The Yale lock supports both Event notifications and Attribute reporting. These can be used to monitor the door lock for any changes in state. Please refer to the sections on Binding and Attribute Reporting for more details.

Alarm Cluster

The Yale lock implements the Alarm Cluster to allow the generation of alarms from the door lock cluster and the power configuration cluster. The alarm conditions and their respective alarm codes are detailed in the Bindings section of this document. Both the Door Lock cluster and the Power Configuration cluster also support an alarm mask field. When not masked, alarm notifications are reported to targets that have subscribed via binding.

Since the Door Lock cluster implements logging the Yale lock's implementation of the alarm cluster does not support an alarm table.

Please reference the Binding section for more information.

Commands

Table 13: Alarm Cluster Supported Commands

ID	Description	Supported
0x00	Reset Alarm	Y
0x01	Reset All Alarms	Y

Both the Reset Alarm command and the Reset All Alarms commands are supported by the Yale lock since they are mandatory commands of the Alarm cluster. However neither of these commands perform any function on the lock because none of the alarms generated by the lock require resetting.

Attributes

There are no attributes associated with the Yale lock's implementation of the Alarm cluster.

Power Configuration Cluster

The Yale lock's implementation of the Power Configuration cluster is used for battery level reporting.

Commands

There are no commands associated with the Power Configuration Cluster.

Attributes

The Yale lock only supports the following attributes from the Power Configuration cluster.

Table 14: Power Configuration Cluster Attributes

ID	Description	Type	Writable	Value	Supported
0x0021	Battery Percentage	INT8U	N	0x00 to 0x64	Y
0x0035	Battery Alarm Mask	BITMAP8	Y	See Table	Y
0x003e	Battery Alarm State	BITMAP32	N	See Table	Y

Low Battery Warnings from Lock

The lock uses the ZigBee binding mechanism to determine which node on the network reports should be sent to. Binding should be set by the network coordinator. Once the correct binding is set then the lock will generate low battery alarms automatically once the battery level is below the threshold. The lock also supports an automatic battery percentage report that will get generated every 8 hours. In order for the lock to automatically generate these reports the attribute reporting for the Battery Percentage needs to be set. It is recommended that the coordinator configures the Battery Alarm State Attribute for attribute reporting on state change. This will notify the coordinator when the lock enters a low battery state and also when that state is cleared by installing new batteries.

Table 15: Battery Messages from Lock

Message	Type of Message	Frequency
Battery Percentage Report	Report	Approx. every 8 hours
Low Battery Alarm	APS Alarm*	When Below Threshold

*Please see Binding section for more information about low battery alarms.

Poll Control Cluster

The Poll Control cluster provides a mechanism for the management of an end device's MAC data request rate. For the Yale Lock the Poll Control cluster can be used by the coordinator to help prolong the battery life of the lock. The MAC Data Request rate (poll rate) is the frequency with which the lock sends a MAC Data Request to its parent. By default the long poll interval of the Yale lock is set to 5 seconds to ensure that it will always receive commands intended for it. If there are periods of time where constant communication with the lock is not required, such as the middle of the night, then the coordinator can modify the long poll interval to allow the lock to sleep for longer periods of time and help prolong the battery life of the lock.

In order to ensure that the coordinator receives the Check-In from the lock the correct binding must be set for the Poll Control cluster. An appropriate value will need to be set for the Check In interval also in order to make sure that the coordinator will be able to adjust the Long Poll interval once the lock needs to be returned to normal operation.

Commands

Table 16: Poll Control Cluster Supported Commands

ID	Description	Supported
0x00	Check-In Response	Y
0x01	Fast Poll Stop	Y
0x02	Set Long Poll Interval	Y
0x03	Set Short Poll Interval	Y

Attributes

The Yale lock supports the following attributes from the Poll Control cluster.

Table 17: Poll Control Supported Attributes

ID	Description	Type	Writable	Value	Supported
0x00	Check-in Interval	INT32U	T	Default = 0x3840 (1hr.)	Y
0x01	Long Poll Interval	INT32U	F	Default = 0x14 (5sec)	Y
0x02	Short Poll Interval	INT16U	F	Default = 0x02(2 qs)	Y
0x03	Fast Poll Timeout	INT16U	T	Default = 0x28 (10 sec.)	Y
0x04	Check-in Interval Min	INT32U	F	Default = 0	Y
0x05	Long Poll Interval Min	INT32U	F	Default = 0	Y
0x06	Fast Poll Timeout Max	INT16U	F	Default = 0	Y

Diagnostics Cluster

The Diagnostics Cluster can be used by the coordinator to determine the link quality between the lock and its parent node.

Commands

The Diagnostic Cluster does not contain any commands.

Attributes

The Yale locks support the following attributes from the Diagnostics Cluster.

ID	Description	Type	Writable	Value	Supported
0x011B	Average Mac Retry per APS msg sent		F	Avg retries required to communicate with parent	Y
0x011C	Last Message LQI		F	LQI of last message sent	Y
0x011D	Last Message RSSI		F	RSSI of last message sent	Y

Time Cluster

The Yale lock implements the Time Cluster as a client. Upon powering up (if already enrolled in a network) or when joining a network for the first time the Yale lock will use a match descriptor request to try and find a Time Cluster server on the network. If no time server can be found on the network then the lock will continue to function normally except the coordinator will not be able to set user schedules and all log records will have invalid time stamps.

If the lock finds a Time Server on the network then next it will read the Time Status attribute of the time server to determine if the server has a valid time for the lock to retrieve. There is also the possibility of having multiple time servers on the network in which case the lock will use the following precedence rules to determine which server to get the time from.

1. A server with the Master bit set shall be chosen over a server without the bit set.
2. The server with the lower short address shall be chosen (note that this means that a coordinator with the Superseding and Master bits set will always be chosen as the network time server.)
3. A time server with neither the Master nor Synchronized bits set should not be chosen as the network time server.

Commands

There are no commands associated with the time cluster.

Attributes

ID	Description	Type	Writable	Value	Supported
0x0000	Time	UTC_TIME	T	(UTC Time)	Y
0x0001	Time Status	BITMAP8	T	See Table	Y
0x0002	Time Zone	INT32S	T	(Local Time Zone)	Y
0x0003	DST Start	INT32U	T	(in Seconds)	Y
0x0004	DST End	INT32U	T	(in Seconds)	Y
0x0005	DST Shift	INT32S	T	(in Seconds)	Y
0x0007	Local Time	INT32U	F	*	Y

* The Local time attribute is used if the Time Server does not implement DST. The Local time is already adjusted for DST and Time Zone.

OTA Cluster

The OTA boot loading cluster is implemented as a client using the Ember ZigBee stack.

The Yale lock will use a match descriptor command to determine if there is an OTA cluster server available on the network. The lock performs this check immediately after enrolling in a network or after powering up if the lock is already in a network. If the match descriptor is successful in finding an OTA server then the lock will query the OTA server to determine if a new firmware image is available for download. If a correct new image with a higher version number is found then the lock will begin the OTA process.

Once the lock is enrolled in the network it then becomes the OTA server's responsibility to notify the lock if there is a new firmware image available for download using an OTA Image Notify command. Once in the network the lock will check for a new image when power cycled or once a day.

Dual Processor OTA updating

The Yale lock utilizes the existing ZigBee OTA cluster to upgrade all processors in the system. This does not require any special handling on the OTA server side, the standard ZigBee OTA server can be used to update both processors in the Yale lock system. To accomplish this all Yale OTA images are a single file that contains an image for each processor in the system. The images are separated in the file by a header describing the image version and size. When the OTA server responds to the lock that there is an image available then the lock will proceed to download the OTA file headers to ensure that the image is valid for the model being updated and decide which image (if any) to download.

If there is an update available for more than one processor the lock will update one image at a time, always starting with the ZigBee image if available. The lock will download part of the OTA file and apply the update. Upon applying the update the lock will reboot at which point it will query the OTA server for any available image so that it can start the process over and download the image for the second processor. This process is repeated until all processors have been updated.

Current File Version

The Current File Version Attribute is used to report multiple versions in order to accommodate the use of a single merged OTA image file. For Yale locks the Current File Version should be interpreted as follows: 0x00ZZ00LL. The byte 'ZZ' refers to the ZigBee processor firmware version and the byte 'LL' refers to the lock processor firmware version.

Binding

The Yale lock supports end device binding. The coordinator must set the appropriate bindings for each cluster it wants to receive notifications from. With the lock being a sleepy end device it is not recommended to have the coordinator poll the lock in order to determine the status of the lock. If the binding is set correctly and the lock is in a healthy ZigBee network it will generate an event notification or alarm up to the coordinator whenever the lock is operated in any way.

Clusters Supporting Binding

Table 18: Clusters Supporting Binding

Clusters Requiring Binding
Door Lock Cluster (0x0101)
Alarm Cluster (0x0009)
Power Configuration Cluster (0x0001)
Poll Control Cluster (0x0020)

Attribute Reporting

The Yale door lock supports attribute reporting for up 5 attributes. If the binding is correctly set then any of the door locks attributes can be configured to generate a Report Attributes Command when the value changes (or periodically). Although any operation that occurs at the lock will result in an event notification, attribute reporting is another option available to the coordinator to ensure that it stays in sync with the Yale door lock. Please refer to the ZigBee Cluster documentation for more details on attribute reporting.

Masking

The door lock cluster contains several writable attributes that allow the coordinator to turn event notifications and alarms on or off through masking. The Tables in the Notifications and Alarms section contain a column titled Bitmap Number which corresponds to the position in the attribute bitmask that controls that particular event or alarm.

By default all Alarms and Events are enabled. Setting a bitmap position to 0 will disable the corresponding event or alarm.

Notification Masking

Table 19: Door Lock Cluster Event Notification Masking Attributes

ID	Description	Type	Writable	Value
0x0041	Keypad operation event mask	BITMAP16	T	see table
0x0042	RF Operation Event Mask	BITMAP16	T	see table
0x0043	Manual Operation Event Mask	BITMAP16	T	see table
0x0044	RFID Operation Event Mask	BITMAP16	T	see table
0x0045	Keypad Programming Event Mask	BITMAP16	T	see table
0x0046	RF Programming Event Mask	BITMAP16	T	see table
0x0047	RFID Programming Event Mask	BITMAP16	T	see table

Alarm Masking

Table 20: Door Lock Cluster Alarm Masking Attribute

ID	Description	Type	Writable	Value
0x0040	Door Lock Alarm Mask	BITMAP16	T	see table

Table 21: Power Configuration Alarm Masking Attribute

ID	Description	Type	Writable	Value
0x0035	Battery Alarm Mask	BITMAP8	Y	See Table

Notifications

The following subsections contain tables listing the event notifications that the Yale door lock supports.

Programmatic Notifications

Table 22: Keypad Programming Event (Event Source = 0x00)

Program Event Code	Event Description	Bitmap Number	Supported
0x01	Master Code Changed, Source: Keypad	1	Y
0x02	PIN Added, Source: Keypad	2	Y
0x03	PIN Deleted, Source: Keypad	3	Y
0x04	PIN Changed, Source: Keypad	4	Y

Table 23: RF Programming Event (Event Source = 0x01)

Program Event Code	Event Description	Bitmap Number	Supported
0x00	MfgSpecific – User Schedule updated (Notification generated once any action on user schedules is completed)	0	Y
0x02	PIN Added, Source: RF	2	Y
0x03	PIN Deleted, Source: RF	3	Y
0x04	PIN Changed, Source: RF	4	Y

Table 24: RFID Programming Event (Event Source = 0x03)

Program Event Code	Event Description	Bitmap Number	Supported
0x00	MfgSpecific- Configuration Parameters updated	0	Y

Operational Notifications

Table 25: Keypad Operation Event (Event Source = 0x00)

Operation Event Code	Event Description	Bitmap Number	Supported
0x01	Lock, source : keypad	1	Y
0x02	Unlock, source : keypad	2	Y

Table 26: RF Operation Event (Event Source = 0x01)

Operation Event Code	Event Description	Bitmap Number	Supported
0x01	Lock, source : RF	1	Y
0x02	Unlock, source : RF	2	Y

Table 27: Manual Operation Event (Event Source = 0x02)

Operation Event Code	Event Description	Bitmap Number	Supported
0x07	One Touch Lock	3	Y
0x0A	Auto Lock	6	Y
0x0D	Manual Lock, Source : Thumbturn/Key	9	Y
0x0E	Manual Unlock, Source : Thumbturn/Key	10	Y

Table 28: RFID Operation Event (Event Source = 0x03)

Operation Event Code	Event Description	Bitmap Number	Supported
0x02	Unlock, source: RFID (Used to report BLE mobile key access. Not supported by all models)	3	Y

APS Alarms

APS Alarms are used to communicate when there is a critical state on the door lock.

Table 29: Door Lock Cluster APS Alarms

Alarm Code	Alarm Condition	Bitmap Number	Supported
0x00	Deadbolt Jammed	0	Y
0x01	Lock Reset To Factory Defaults	1	Y
0x03	RF Module Power Cycled	3	Y
0x04	Tamper - wrong code entry limit	4	Y
0x05	Tamper - Front Escutcheon Removed	5	Y

Table 30: Power Configuration Cluster APS Alarms

Alarm Code	Alarm Condition	Bitmap Number	Supported
0x10	Battery Min Threshold		Y
0x11	Battery Threshhold 1		Y
0x12	Battery Threshhold 2		Y

Basic Operation

Installing the Network Module

IMPORTANT: The batteries must be removed prior to removing or inserting the network module:

- Remove batteries
- Insert network module
- Reinstall batteries

Network Operations

Enroll/Add device to network (Inclusion Mode)

- Enter the 4-8 digit Master PIN code followed by the ⚙ key.
- Press the 7 key followed by the ⚙ key.
- Press the 1 key followed by the ⚙ key.

Unenroll/Remove device from network (Exclusion Mode)

- Enter the 4-8 digit Master PIN code followed by the ⚙ key.
- Press the 7 key followed by the ⚙ key.
- Press the 3 key followed by the ⚙ key.

When the Yale lock is unenrolled/excluded from the network through the device menu mode, the user code database will not be cleared and the configuration settings will not be reset to the defaults.

Appendix

Model Identifiers

Lock Platform	Models	Image Type ID	Model ID String	# User Slots	DPS Support	BLE Support (RFID)
WGA6	YRD216	0x8004	"YRD216 PBDB"	250	NA	NA
	YRD226	0x8002	"YRD226/246 TSDB"	250	NA	NA
	YRD246		"YRD226/246 TSDB"	250	NA	NA
WGA11	NTB610	0x8001	"NTB PB"	500	A	NA
	NTB630			500	A	NA
	NTB620	0x8003	"NTB TS"	500	A	NA
	NTB640			500	A	NA
WGA16	YRD446	0x8002	"YRD446 BLE TSDB"	250	NA	S

NA = Not Available | A = Available with optional hardware | S = Supported