



ASR1600 Engine Parameters

Application Note

Version 1.2

Contents

INTRODUCTION	1
AUTOMATIC GAIN CONTROL.....	2
SPEECH DETECTION	3
Start mode	3
Stop mode	4
SPEECH DECODER.....	5
REJECTION AND KEYWORD SPOTTING.....	7
CONFIDENCE VALUES	9

INTRODUCTION

The ASR1600 SPI (Service Provider Interface) is the low level interface to the ASR1600 phonetic speech recognition engine. It provides access to a limited set of engine parameters. They are made accessible to an application because their optimal values depend on the complexity of the task, the environment,...

The parameters can be functionally grouped together. Following the signal from acquisition to recognition result, there are parameter groups that influence:

- the microphone amplification
- the detection of begin and end of speech
- the search complexity of the recognition engine
- the rejection and keyword spotting
- the search complexity of the spelling postprocessor.

Each group is discussed in the following sections.

AUTOMATIC GAIN CONTROL

Parameter	range	default
AGCON	boolean	false
FARTALK	boolean	true

The engine provides the software to adjust the analog gain of the speech signal if the audio hardware configuration permits it. However, automatic gain control (AGC) generates instantaneous fluctuations of the signal levels. Therefore, it is advised to disable this feature when it is not strictly needed. A better approach is to look for a good value of the gain using a microphone check procedure at initialization of the application, and to keep the gain fixed during recognition.

The FARTALK flag determines the sensitivity of the automatic gain control. In an office a speaker might move from or towards a desktop microphone. In this situation enabling the FARTALK flag allows to capture a wide range of signal levels. In most other situations the distance between speaker and microphone is fixed. It is then safer to disable the FARTALK flag, such that the AGC does not react on weak background noises.

Advised values for the FARTALK flag:	
office desktop microphone	true
car	false

SPEECH DETECTION

Parameter	range	default
START_ENABLE	boolean	true
SENSITIVITY	0 – 4000 (dB/100)	1200 (12 dB)
MINSPEECH	10 – 400 (msec)	60
TS_ENABLE	boolean	true
TS	100 – 2000 (msec)	300
TIMEOUT	0 - 30 (sec)	0 (= no time out)

A signal may contain a lot of silence surrounding the useful speech part. The ASR1600 engine provides both manual and automatic mode for detecting the begin and end points of the speech.

Start mode

The engine is started with the function `casrStart`. Automatic begin of speech detection is enabled/disabled by the `START_ENABLE` flag. If enabled, the engine starts in sleep mode. In sleep mode only limited processing occurs in order to look for a speech event. A speech event is defined as an energy jump of `SENSITIVITY/100` dB during at least `MINSPEECH` msec. If such an event is detected, the engine switches to run mode and uses full CPU power. In run mode the engine searches the grammar for a word that best matches the incoming signal. If `START_ENABLE=0`, the engine immediately starts in run mode. Sleep mode is a useful feature when an unknown amount of silence can precede speech.

The optimal value of the `SENSITIVITY` parameter is proportional to the signal to noise ratio of the target application. If the value is high, the engine becomes a bit deaf, as a larger energy jump is needed before the engine returns a recognition result. An unrealistically high value may even cause the engine to hang in the sleep state. On the other hand, a low value introduces the risk that the engine starts full recognition on weak non-stationary noises, such as background speech.

Advised values for the SENSITIVITY parameter:	
Office quality recordings	2000
Car quality recordings	1200

The `MINSPEECH` parameter avoids that the engine reacts on short high energy events such as clicks or door slams. There is no need to change the default value of 60 msec, except maybe in very non-stationary noise environments.

Stop mode

The recognition is stopped if any of the following three conditions are met:

1. the function `casrStop` is called (manual stop)
2. the recognizer has detected at least TS msec of silence after a word has been found and the automatic stop mode is enabled (TS_ENABLE=1)
3. the recognizer is at least TIMEOUT sec in either sleep, run or record mode (record mode is used for user word training). Setting TIMEOUT=0 disables this feature.

The TIMEOUT parameter serves as a safety net to avoid that the engine hangs in some intermediate state.

In automatic stop mode, the TS parameter determines the minimum trailing silence duration. In an isolated word grammar a low value is advantageous, because the response time of the engine is decreased. For continuous speech, the value has to be longer than the duration of a pause between two words, in order to avoid that the engine already returns a recognition result while the speaker is still talking.

Advised values for the TS parameter:	
Isolated words	300 msec
Continuous speech	800 msec

SPEECH DECODER

Parameter	range	default
MAXNBEST	1-1000	10
ACCURACY	100-10000	300

The decoder module is the central part of the engine. It searches the grammar for the most probable MAXNBEST word strings. The ACCURACY parameter sets a target value for the number of hypotheses that are simultaneously searched at any time. If the number of hypotheses gets bigger than ACCURACY, the ones that score worst are removed from the search. A hypothesis is the combination of a word string and a time alignment. The decoder module typically contains multiple copies of the same word string with different time stamps for entering and leaving words.

In order to cut down on system resources, the ACCURACY parameter is ideally as low as possible while avoiding loss in recognition performance. Indeed, if the value is too low, the hypothesis that matches best on the complete signal may have a bad local match on a begin part and be removed from the search.

Unfortunately the optimal value is very much task dependent and difficult to predict. It is certainly related to the vocabulary size and the grammar. In the table below a rule of thumb is given that relates the ACCURACY parameter to the branching factor. That is the number of words that follow a word or silence at some point in the finite state grammar. As an example, an isolated word grammar has N words and a leading and trailing silence. The followers of the leading silence are the N words. All N words have trailing silence as the only follower. The maximum branching factor is N, the average branching factor is $2N/(N+2)$.

Quickly calculating a wrong recognition result makes no sense. Therefore it is strongly advised to set the accuracy as high as possible given the available resources and a reasonable response time towards the user. But for specific grammars, tuning of the ACCURACY parameter may be needed. This can be done by testing the recognition performance for several values of the ACCURACY parameter on a small database. Initially it is set at a high value, and then gradually decreased until an unacceptable performance degradation is observed.

Given an offset, the CPU load evolves roughly linearly as a function of the ACCURACY parameter. There is also a minor linear increase in memory resources.

Advised values for the ACCURACY parameter:	
isolated words	Max(300,6 x number of words)
connected digits	300
spelling (connected letters)	1500
keyword spotting	Max(300,6 x number of keywords)
continuous speech	Max(300, 6 x maximum branching factor, 50 x average branching factor)

REJECTION AND KEYWORD SPOTTING

Parameter	range	default
REJECTION	0-100	50
GARBAGE	0-100	50

The engine itself always returns a recognition result, even if all words in the grammar have a bad match on the signal. It is up to the application to decide whether it accepts the recognized word string or not. This feature is called rejection.

In order to provide the application with the necessary information, the engine returns a score and a confidence value for each recognition result. The score reflects the match of the signal on the word string model. The higher the score, the worse the match between the signal and the recognized word string. The score is proportional to the signal length and therefore not useful for rejection purposes.

The confidence value is independent of the signal length. It is calculated by comparing the match of the signal on the recognized word string model and on a model composed of "average speech" words. The "average speech" models correspond to broad classes of speech-like sounds.

Word strings with a value above / below a certain threshold (typically 50%) can be accepted / rejected. In order to give the application control over the rejection behavior, a REJECTION parameter is provided. The higher the REJECTION parameter, the better the match on the average speech models and the lower the confidence values.

The engine provides confidence values for sentences and for individual words. The longer and the more phonetically rich a speech fragment, the more reliable the confidence value. Therefore confidence values on the word level have to be treated with care. For very short words like 'a' and 'the' they are unreliable.

In order to tune the REJECTION parameter, two sources of errors have to be considered:

- false rejection: a correctly recognized word that is rejected.
- false acceptance: a wrongly recognized word that is accepted.

The default REJECTION value is chosen as such that the sum of both types of errors is minimized. In some applications the consequence of accepting a wrong word may be worse than rejecting a good word. If this is the case, it makes sense to re-tune the REJECTION parameter. This can be accomplished by collecting a considerable database preferably recorded in the target environment. The engine provides an any speech model <...> in order to absorb "don't care" speech. It is used if at some place in the grammar the application is only interested in a limited set of keywords. Similar to the REJECTION parameter, the GARBAGE parameter modulates the match on the <...> model. The higher is GARBAGE, the longer are the parts of the signal that are recognized as <...>. Tuning the default value may again be useful in applications

where the cost of missing a keyword is substantially different from the cost of detecting a wrong keyword. Keywords are returned with a confidence value, such that a rejection test can be applied.

CONFIDENCE VALUES

This is not an engine parameter, this is the confidence value returned by the recognition result callback.

The result callback returns a word confidence value (for every word in every sentence) and a sentence confidence value (for every sentence).

The word confidence value indicate how sure we are that that word was uttered. The sentence confidence value indicates how sure we are that that sentence was uttered. The sentences are ordered in the array according to this confidence value, meaning that the most probable sentence is at array element 0.

This confidence value can be used to reject recognition results if they have values that aren't high enough (i.e. using a threshold for acceptance or rejection).

Range	Minimum value	Maximum value
Confidence value	0	10000

Take caution though that this value is not a probability.

A confidence value of 10000 means that the recognizer is very sure that that sentence (or word) was uttered, not that the engine is 100% sure.

You also have to keep in mind that this value is (somewhat) context dependent. You cannot define a threshold which is ideal for all contexts. Some contexts will return higher values than other. The size of the context and the word confusability will typically contribute to different ranges of confidence values.

A good strategy for an application developer is to test a newly created context and see what results are returned.
